

## **Teacher Access to Student Information**

Teacher access to Student Information is a web based application that allows viewing of student data maintained on the VAX Student Information System (SIS). The system consists of three main parts;

- The first part is a VAX application used to extract teacher viewable information from the SIS and exports this data as ASCII text files to a network file share.
- The second part is a SQL 2000 database, that imports the ASCII text files from the network file share and condition that data for viewing via the Intranet.
- The final portion of this system is a Java based application that controls user access, querying and display of data in the SQL 2000 database. This Java application conditions data from the SQL 2000 database into XML and uses an XSL style sheet to display this data in the HTML Web Interface, viewable by the IE browser on teacher work stations.

# Web Interface

The Web Interface is designed specifically for Microsoft Internet Explorer (5.0 or greater). This dynamic web interface uses the HTML generated by the Java Application and JavaScript to provide a dynamic, easy to use and understand interface.

When a Teacher first accesses this application, she/he is prompted to enter the Network User ID and Password for that Teacher. Once the Teacher is authenticated, the following interface is displayed:

Enter you username and password.

Username:

Password:

After logging in, the Teacher, (in this example: John W. Smith of Adams Elementary), the following screen is displayed.

The screenshot shows a web interface with a light blue header and a white main content area. In the top left corner, it displays 'Teacher: John W. Smith' and 'Emp. ID: 123456789'. The main title is 'Teacher Access to Student Information'. On the right, there is a 'Log Off' button. In the center, there is a logo with a star and the text 'DELIVERING ON THE DREAM'. On the right side, there is a search section with a 'School' dropdown menu set to 'ADAMS ELEMENTARY'. Below this are four input fields: 'Home Room', 'Student Last Name', 'Student ID', and 'Student First Name'. To the right of these fields are three buttons: 'Find', 'Clear', and 'Reset'.

Teacher: John W. Smith  
Emp. ID: 123456789

Teacher Access to Student Information

VER. 1.0.0 Ab  
**Log Off**

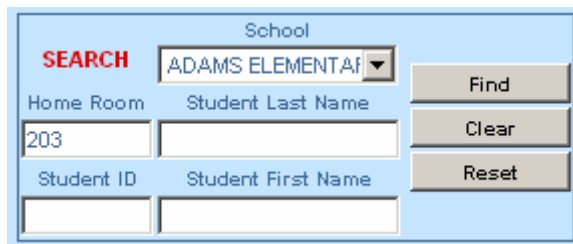
School  
SEARCH ADAMS ELEMENTARY Find

Home Room Student Last Name Clear

Student ID Student First Name Reset

John W. Smith, Welcome to Teacher access to Student Information

From here the Teacher can obtain a list of students according to desired School, Home Room, Student Last and/or First Name, and Student ID.



The image shows a search form with a light blue background. At the top left, the word "SEARCH" is written in red. To its right is a dropdown menu labeled "School" with "ADAMS ELEMENTARY" selected. Below this are three rows of input fields. The first row has "Home Room" and "Student Last Name" labels, with "203" entered in the Home Room field. The second row has "Student ID" and "Student First Name" labels, with empty fields. To the right of these fields are three buttons: "Find", "Clear", and "Reset".

**Example of a search for all students in Adams Elementary who are assigned to Home Room 203**

The **Find** button will initiate the search.

The **Clear** button will clear all search criteria.


The **Reset** button will restore the search criteria from the previous search.

**In the example below, the Teacher has asked for a listing of all student at Adams Elementary who are assigned to Home Room 203**

By clicking on a particular Student ID from the list, details for that student are displayed.

Teacher: John W. Smith  
Emp. ID: 123456789

## Teacher Access to Student Information



VER. 1.0.0 Ab  
**Log Off**

Student id	Name	Home Room
011111111	AABERT, ABLE ABLE	203

Record Last Modified on 01/03/2002

**+ HEALTH CONCERNS +**  
MILD CP/NO PERTUSSIS IMMU

SCHOOL

SEARCH

Home Room  Student Last Name

Student ID  Student First Name

STUDENT

PARENT/GUARDIAN

EMERGENCY CONTACTS

DAY CARE

Grade	Birthdate	Gender	Ethnicity	Option	Grad Year	Admit Date	Status	Current School	Previous School
10	06/28/1902	MALE	WHITE	REGULAR PROGRAM	2004	09/06/2000	Active	ADAMS ELEMENTARY	ADAMS ELEMENTARY

Home Address	Mailing Address	Student Phone	Living With	Joint Custody
11111 GIGGLE PL NW SEATTLE, WA. 98177	11113 GIGGLE PL NW SEATTLE, WA. 98177	<input type="text"/>	MOTHER ONLY	N

Privacy	504	Homeless	Case Worker	Case Worker Phone
N	C	N		

**BILINGUAL INFO**

Bilg Elig

Country

Primary

Home

**SPECIAL ED INFO**

Level

Disability

Type Svc

Unit

In the upper left corner of the display are the Name and Employee ID for the teacher that has logged onto the system. If details for a specific student are being viewed, the Student ID, Name and Home Room of that student are displayed. As well as the date when this student's information was last updated.

Teacher: John W. Smith  
Emp. ID: 123456789

Student id	Name	Home Room
011111111	AABERT, ABLE ABLE	203

Record Last Modified on 01/03/2002

If a student has any Health Concerns, No Contact Information and/or is a "Confidential" Student, boxes are displayed above the tab bar to indicate details on these situations.

**CONTACT OFFICE FOR ADDITIONAL  
INFORMATION  
REGARDING THIS STUDENT**

**CONFIDENTIAL STUDENT**

**!!! NO CONTACT !!!  
W/MOTHER!!!**

**NO CONTACT**

**+ HEALTH CONCERNS +  
ASTHMA/INHALER PRN**

**HEALTH CONCERN**

## Tab Contents

Each tab in the display will bring up specific details regarding the student. These tabs include **STUDENT**, **PARENT/GUARDIAN**, **EMERGENCY CONTACTS**, and **DAY CARE** (if that data is available for this student).



*Note:* If a student is a “Confidential” Student, only the STUDENT tab is displayed for that student. This tab will contain only minimal information regarding the “Confidential” Student.



Below are screen snapshots of the data displayed in each tab.

### Student Tab

STUDENT										PARENT/GUARDIAN		EMERGENCY CONTACTS			DAY CARE	
Grade	Birthdate	Gender	Ethnicity	Option	Grad Year	Admit Date	Status	Current School	Previous School							
10	06/28/1902	MALE	WHITE	REGULAR PROGRAM	2004	09/06/2000	Active	ADAMS ELEMENTARY	ADAMS ELEMENTARY							
Home Address			Mailing Address			Student Phone	Living With	Joint Custody								
11111 GIGGLE PL NW SEATTLE, WA. 98177			11113 GIGGLE PL NW SEATTLE, WA. 98177				MOTHER ONLY	N								
		Privacy	504	Homeless	Case Worker	Case Worker Phone										
		N	C	N												
<b>BILINGUAL INFO</b>				<b>SPECIAL ED INFO</b>												
Bilg Elig	<input type="checkbox"/>	Level	DROPPED													
Country	<input type="checkbox"/>	Disability														
Primary	<input type="checkbox"/>	Type Svc														
Home	<input type="checkbox"/>	Unit														

### Bilingual and Special Ed Information

BILINGUAL INFO		SPECIAL ED INFO	
Bilg Elig	ELIGIBLE: LOW ITBS/5 OR MORE YEARS IN PROGRAM	Level	SPECIALIZED SELF-CONTAINED
Country	CHINA	Disability	338 - MULTIPLE DISABILITIES 335 - ORTHOPEDICALLY IMPAIRED 348 - MILD MENTAL RETARDATION
Primary	CANTONESE	Type Svc	SPECIALIZED SELF-CONTAINED
Home	CANTONESE	Unit	LOW INCIDENCE 6:3

## PARENT/GUARDIAN

STUDENT	<b>PARENT/GUARDIAN</b>	EMERGENCY CONTACTS	DAY CARE
---------	------------------------	--------------------	----------

**Parent/Guardian 1** Relationship Home Phone Unlisted Email Employer Work Phone Ext. Pager

AABERT, CARLA	MOTHER	(206) 5555555		AAAAAZ@HOTMAIL.COM	BOB'S	(206) 5555555		
---------------	--------	---------------	--	--------------------	-------	---------------	--	--

Home Address                      Mailing Address

11111 GIGGLE PL NW SEATTLE, WA. 98177	11113 GIGGLE PL NW SEATTLE, WA. 98177
--	--

**Parent/Guardian 2** Relationship Home Phone Email Employer Work Phone Ext. Pager

AABERT, GARY	FATHER	(206) 5555555						
--------------	--------	---------------	--	--	--	--	--	--

Address

0000 QUEEN ANNE AV N SEATTLE, WA. 98109	Mail to? <input type="checkbox"/> Y
--	-------------------------------------

**Parent/Guardian 3** Relationship Home Phone Email Employer Work Phone Ext. Pager

--	--	--	--	--	--	--	--	--

Address

	Mail to? <input type="checkbox"/> N
--	-------------------------------------

**Cell Phone Numbers**

Name	Cell Phone	Name	Cell Phone
AAAAAA, BBBBBB	(206) 5555555		

## EMERGENCY CONTACTS

STUDENT	<b>PARENT/GUARDIAN</b>	<b>EMERGENCY CONTACTS</b>	DAY CARE
---------	------------------------	---------------------------	----------

**Emergency Contact 1** Phone Relationship

GARBO, GRETA	(206) 5555555	SISTER
--------------	---------------	--------

**Emergency Contact 2** Phone

	(206) 5555555
--	---------------

**Emergency Contact 3** Phone

--	--

**Emergency Contact 4** Phone

--	--

## DAY CARE

STUDENT	<b>PARENT/GUARDIAN</b>	EMERGENCY CONTACTS	<b>DAY CARE</b>
---------	------------------------	--------------------	-----------------

Daycare Center      Contact      Title      Phone      Ext

BARNY'S BABY BRAINWASHING	MR. PURPLE	DIRECTOR	(206) 5555555	22222
---------------------------	------------	----------	---------------	-------

Address

1111 A AVE NNW SEATTLE 99999-9999
--------------------------------------


When the Teacher has finished using the application, he/she can log off the system by clicking on the Log Off button at the upper right side of the display and will be returned to the Teachers Tools area of [HTTP://inside.SeattleSchools.org](http://inside.SeattleSchools.org) .

**Log Off**

Teacher: John W. Smith  
Emp. ID: 123456789

Teacher Access to Student Information

VER: 1.0.0 Ab



**SEARCH** School: ADAMS ELEMENTARY

Home Room	Student Last Name	Clear
<input type="text"/>	<input type="text"/>	
Student ID	Student First Name	Reset
<input type="text"/>	<input type="text"/>	

**Goodbye John W. Smith, You have been Logged Off, and are being returned to Teacher Tools!!!**

# Technical Details

## Web Interface

This interface consists of the resulting HTML, generated by the Java Application (**StudentInfo.java**) and its associated XSL style sheet, (**studentinfo-html.xsl**) and a JavaScript file (**studentinfo.js**) used for dynamic display of information in the HTML document.

The contents of **StudentInfo.java** can be found in *Appendix I*.

The XML generated by **StudentInfo.java** varies depending on the desired display. The XSL style sheet (**studentinfo-html.xsl**) will display information according to what XML dataset sent. All tab controls and dynamic data display is handled by JavaScript contained in **studentinfo.js**. There are four basic XML data sets passed to XSL. These include **initial log in**, **search results**, **individual student data** and **user log out**.

### *Initial Log In*

The typical dataset sent to **studentinfo-html.xsl** on user log in is as follows:

```
<content><title>Teacher Access to Student Information</title>
<teachersis>
<teachername>John W. Smith</teachername>
<teacherid>123456789</teacherid>
<version>VER. 1.0.1</version>
<teacherschoollist>
<school value="0201" id="0201">ADAMS ELEMENTARY</school>
<school value="0011" id="0011">BALLARD HIGH SCHOOL</school>
</teacherschoollist>
<currentschool>0</currentschool>
<statusmsg>John W. Smith, Welcome to Teacher access to Student Information</statusmsg>
<arrTabLabelcontent>"          "</arrTabLabelcontent>
<arrDisplayDivcontent>"blankfill"</arrDisplayDivcontent>
<logout></logout>
</teachersis>
</content>
```

<teacherschoollist> contains the list of schools that this individual is authorized to see.

<arrTabLabelcontent> contains the labels for the tabs to be displayed.

<arrDisplayDivcontent> contains the div to display when the tab labeled by the same index in

<arrTabLabelcontent> is clicked.



## Search Results

The typical dataset sent to **studentinfo-html.xsl** for search results is as follows, (search done for students in Adams Elementary who have the last name of “aabert” in homeroom “203”):

```
<content><title>Teacher Access to Student Information</title>
<teachersis>
<teachername>John W. Smith</teachername>
<teacherid>123456789</teacherid>
<version>VER. 1.0.1</version>
<teacherschoollist>
<school value="0201" id="0201">ADAMS ELEMENTARY</school>
<school value="0011" id="0011">BALLARD HIGH SCHOOL</school>
</teacherschoollist>
<currentschool>0201</currentschool>
<searchfirstname></searchfirstname>
<searchlastname>aabert</searchlastname>
<searchid></searchid>
<searchhr>203</searchhr>
<searchresults>
<student>
<id>011111111</id>
<last>AABERT</last>
<first>ABLE</first>
<middle>ABLE</middle>
<birth>06&#47;28&#47;1902</birth>
<gender>MALE</gender>
<hr>203</hr>
<grade>10</grade>
<status r="">Active</status>
<curschool r="">ADAMS ELEMENTARY</curschool>
</student>
<student>
<id>011111112</id>
<last>AABERT</last>
<first>BAKER</first>
<middle>ABLE</middle>
<birth>06&#47;28&#47;1902</birth>
<gender>MALE</gender>
<hr>203</hr>
<grade>10</grade>
<status r="">Active</status>
<curschool r="">ADAMS ELEMENTARY</curschool>
</student>
<student>
<id>011111113</id>
<last>AABERT</last>
<first>CHARLEY</first>
<middle>ABLE</middle>
<birth>06&#47;28&#47;1902</birth>
<gender>MALE</gender>
<hr>203</hr>
<grade>10</grade>
<status r="">Active</status>
<curschool r="">ADAMS ELEMENTARY</curschool>
</student>
<recordcount>3</recordcount>
</searchresults>
<statusmsg></statusmsg>
<arrTabLabelcontent>"SEARCH RESULTS"</arrTabLabelcontent>
<arrDisplayDivcontent>"StudentsFoundList"</arrDisplayDivcontent>
</logout></logout>
</teachersis>
</content>
```

## Individual Student Information

The typical dataset sent to **studentinfo-html.xsl** for displaying individual student information is as follows:

```
<content><title>Teacher Access to Student Information</title>
<teachersis>
<teachername>John W. Smith</teachername>
<teacherid>123456789</teacherid>
<version>VER. 1.0.1</version>
<teacherschoollist>
<school value="0201" id="0201">ADAMS ELEMENTARY</school>
<school value="0011" id="0011">BALLARD HIGH SCHOOL</school>
</teacherschoollist>
<currentschool>0201</currentschool>
<searchfirstname></searchfirstname>
<searchlastname>aabert</searchlastname>
<searchid></searchid>
<searchhr>203</searchhr>
<currentstudent>
<studentid>011111112</studentid>
<studentname>AABERT, BAKER ABLE</studentname>
<studenthr>203</studenthr>
<moddate>01&#47;03&#47;2002</moddate>
</currentstudent>
<specialinfo>
<nocontact>ADAM WEST</nocontact>
</specialinfo>
<studentinfo>
<grade>10</grade>
<birth>06&#47;28&#47;1902</birth>
<gender>MALE</gender>
<ethnicity>CHINESE</ethnicity>
<option>REGULAR PROGRAM</option>
<gradyear>2004</gradyear>
<displaydate type="Admit Date">09&#47;06&#47;2000</displaydate>
<status>Active</status>
<curschool>ADAMS ELEMENTARY</curschool>
<prevschool>ADAMS ELEMENTARY</prevschool>
<pladdress>
<street>11111 GIGGLE PL NW</street>
<city>SEATTLE</city>
<state>WA</state>
<zip>98177</zip>
</pladdress>
<plmailaddress>
<street>11113 GIGGLE PL NW</street>
<city>SEATTLE</city>
<state>WA</state>
<zip>98177</zip>
</plmailaddress>
<stuphone><area></area><phone></phone></stuphone>
<livingwith>MOTHER ONLY</livingwith>
<joint>N</joint>
<privacy>Y</privacy>
<homeless>N</homeless>
<caseworker></caseworker>
<cwphone><area></area><phone></phone></cwphone>
<five04>CONFIRMED 504 STUDENT, NEEDS ACCOMMODATION</five04>
<bielig>ELIGIBLE: LOW ITBS&#47;5 OR MORE YEARS IN PROGRAM</bielig>
<country>CHINA</country>
<pri>CANTONESE</pri>
<home>CANTONESE</home>
<level>SPECIALIZED SELF-CONTAINED</level>
<disabilities>
<db><code>338</code><desc>MULTIPLE DISABILITIES</desc></db>
<db><code>335</code><desc>ORTHOPEDICALLY IMPAIRED</desc></db>
<db><code>348</code><desc>MILD MENTAL RETARDATION</desc></db>
</disabilities>
<svctype>SPECIALIZED SELF-CONTAINED</svctype>
<unit>LOW INCIDENCE 6:3</unit>
</studentinfo>
<parentinfo>
<p1name>AABERT, CARLA</p1name>
```

<plrelation>MOTHER</plrelation>  
<plphone><area>206</area><phone>5555555</phone></plphone>  
<plunlisted></plunlisted>  
<plemail>AAAAAAZ@HOTMAIL.COM</plemail>  
<plemployer>BOB&apos;S </plemployer>  
<plworkphone><area>206</area><phone>5555555</phone></plworkphone>  
<plext></plext>  
<plpager><area></area><phone></phone></plpager>  
<pladdress>  
<street>11111 GIGGLE PL NW</street>  
<city>SEATTLE</city>  
<state>WA</state>  
<zip>98177</zip>  
</pladdress>  
<plmailaddress>  
<street>11113 GIGGLE PL NW</street>  
<city>SEATTLE</city>  
<state>WA</state>  
<zip>98177</zip>  
</plmailaddress>  
<p2name>AABERT, GARY</p2name>  
<p2relation>FATHER</p2relation>  
<p2phone><area>206</area><phone>5555555</phone></p2phone>  
<p2email></p2email>  
<p2employer></p2employer>  
<p2workphone><area></area><phone></phone></p2workphone>  
<p2ext></p2ext>  
<p2pager><area></area><phone></phone></p2pager>  
<p2address>  
<street>0000 QUEEN ANNE AV N</street>  
<city>SEATTLE</city>  
<state>WA</state>  
<zip>98109</zip>  
</p2address>  
<p2mailhere>Y</p2mailhere>  
<p3name></p3name>  
<p3relation></p3relation>  
<p3phone><area></area><phone></phone></p3phone>  
<p3email></p3email>  
<p3employer></p3employer>  
<p3workphone><area></area><phone></phone></p3workphone>  
<p3ext></p3ext>  
<p3pager><area></area><phone></phone></p3pager>  
<p3address>  
<street></street>  
<city></city>  
<state></state>  
<zip></zip>  
</p3address>  
<p3mailhere>N</p3mailhere>  
<cpname>AAAAAA, BBBB</cpname>  
<cpphone><area>206</area><phone>5555555</phone></cpphone>  
<cqname></cqname>  
<cqphone><area></area><phone></phone></cqphone>  
</parentinfo>  
<emergencyinfo>  
<contactname1>GARBO, GRETA</contactname1>  
<contactphone1><area>206</area><phone>5555555</phone></contactphone1>  
<relationship1>SISTER</relationship1>  
<contactname2>MARY WORTH</contactname2>  
<contactphone2><area>206</area><phone>5555555</phone></contactphone2>  
<contactname3></contactname3>  
<contactphone3><area></area><phone></phone></contactphone3>  
<contactname4></contactname4>  
<contactphone4><area></area><phone></phone></contactphone4>  
</emergencyinfo>  
<daycareinfo>  
<center>BARNY&apos;S BABY BRAINWASHING</center>  
<contact>MR. PURPLE</contact>  
<title>DIRECTOR</title>  
<phonenumber><area>206</area><phone>5555555</phone></phonenumber>  
<ext>22222</ext>  
<address>  
<street>1111 A AVE NNW</street>  
<state>SEATTLE</state>

```

<zip>99999-9999</zip>
</address>
</daycareinfo>
<arrTabLabelcontent>"STUDENT", "PARENT/GUARDIAN", "EMERGENCY CONTACTS"</arrTabLabelcontent>
<arrDisplayDivcontent>"StuInfo","ParInfo","EmergInfo"</arrDisplayDivcontent>
</logout>
</teachersis>
</content>

```

## *User Log Out*

The typical dataset sent to **studentinfo-html.xsl** on user log out is as follows:

```

<content><title>Teacher Access to Student Information</title>
<teachersis>
<teachername>John W. Smith</teachername>
<teacherid>123456789</teacherid>
<version>VER. 1.0.1</version>
<teacherschoollist>
<school value="0201" id="0201">ADAMS ELEMENTARY</school>
<school value="0011" id="0011">BALLARD HIGH SCHOOL</school>
</teacherschoollist>
<currentschool>0201</currentschool>
<errmsg>Goodbye John W. Smith, You have been Logged Off, and are being returned to Teacher
Tools!!!</errmsg>
<arrTabLabelcontent>" " </arrTabLabelcontent>
<arrDisplayDivcontent>"blankfill"</arrDisplayDivcontent>
</logout>http://sequoia.inside.seattleschools.org/area/teachertools/index.dxml</logout>
</teachersis>
</content>

```

The URL contained in the <logout> tag is used in a Meta Tag to forward the user to that location after 5 seconds.

## **studentinfo-html.xsl**

**studentinfo-html.xsl** will take the XML data delivered by **StudentInfo.java** and create the appropriate HTML.

The contents of **studentinfo-html.xsl** can be found in *Appendix II*.

## **studentinfo.js**

The dynamic HTML, tab control and display, and the conditioning of HTTP Request to the web server is controlled by **studentinfo.js**. Because this application is written specifically for MS Internet Explorer, no cross browser compatibility issues are addressed in **studentinfo.js**. To reduce the time necessary to paint gif images to represent parts of the “tabs”, **studentinfo.js** uses tables with the appropriate background color set to emulate a corresponding gif images. This allows more rapid repaints of the screen as tab states change, (as opposed to reloading gif images from cache). This approach does not work always in other browsers.

The contents of **studentinfo.js** can be found in *Appendix III*.

## SQL 2000 Database Teacher2SIS

### Data Imports

This database receives data from 8 text files generated by the VAX Student Information System.

These files are ASCII text files which have fields delimited by the '|' character. They are as follows:

**SCHOOL-EXTRACT.TXT**  
**STBL-EXTRACT.TXT**  
**STUD-EXTRACT.TXT**  
**STUD-SPED.TXT**  
**STUD-BILG.TXT**  
**TEAC-DIVI.TXT**  
**TEAC-EMPID.TXT**  
**CENT-EXTRACT.TXT (not used as of 3/01/02 but still exist in dBase for future ref.)**

The ASCII text files are placed at [\\snocap02\ftp\\$\teacher](\\snocap02\ftp$\teacher) at the end of each VAX process run.

The import of the ASCII text files is controlled from the NT command file, **import\_Teacher2sis\_files.cmd**, which is run from a scheduled NT batch command line as follows:

```
import_Teacher2sis_files.cmd <Server name of SQL database> <Admin Login> <Admin Password> <Path to import files>
```

**import\_Teacher2sis\_files.cmd** is located in [\\snocap02\ftp\\$\teacher\batch](\\snocap02\ftp$\teacher\batch) and is run from a scheduled event on SNOCSQ03 at 3:00 a.m. daily.

As each file is processed, that file name is changed from **XXX-XXX.txt** to **done-XXX-XXX\_MM-DD-YYYY.txt** (where **MM** = month, **DD** = day and **YY**= year).

A log file is created documenting the process of the import process and is emailed to individuals listed in **import\_Teacher2sis\_files.cmd**. This log file is called **Import\_Teacher-SIS\_MM-DD-YYYY.log** (where **MM** = month, **DD** = day and **YY**= year).

The import process consist of importing the ASCII text files to a corresponding IMPORT table and then updating the tables used to actually display the student data. These import tables are as follows:

**SCHOOL-EXTRACT-IMPORT**  
**STBL-EXTRACT-IMPORT**  
**STUD-EXTRACT-IMPORT**  
**STUD-SPED-IMPORT**  
**STUD-BILG-IMPORT**  
**TEAC-DIVI-IMPORT**  
**TEAC-EMPID-IMPORT**  
**CENT-EXTRACT-IMPORT (not used as of 3/01/02 but still exist in dBase for future ref.)**

Once the data has been imported into the IMPORT tables, conditioning is performed on the display tables (depending on the table involved) by the stored procedure *sp\_updateAll*. Then the contents of the appropriate IMPORT tables are inserted into the DISPLAY tables. The DISPLAY tables are as follows:

**SCHOOL-EXTRACT**  
**STBL-EXTRACT**  
**STUD-EXTRACT**  
**STUD-SEARCH**  
**STUD-SPED**  
**STUD-BILG**  
**TEAC-DIVI**  
**TEAC-EMPID**  
**CENT-EXTRACT** (not used as of 3/01/02 but still exist in dBase for future ref.)

The logic used to display the content of these tables into a readable format is conducted by SQL queries from within the Java application (**StudentInfo.java**).

A second database was created for the purpose of creating reports without impacting the performance of the Teacher2SIS database. This report database is called *Teacher2SISReports* and contains copies of the following tables:

**SCHOOL-EXTRACT**  
**STBL-EXTRACT**  
**STUD-EXTRACT**  
**STUD-SPED**  
**STUD-BILG**  
**TEAC-DIVI**  
**TEAC-EMPID**

The copied tables are updated nightly by the import process conducted by **import\_Teacher2sis\_files.cmd**.

The contents of **import\_Teacher2sis\_files.cmd** can be found in *Appendix IV*.

**Stored procedures** used by Teacher2SIS are contained in *Appendix V*.

## Table Structures used in Teacher2SIS

The following is the structure of each table used in Teacher2SIS

### CENT-EXTRACT

#### Columns

Name	Type	Size
SCHOOL_ID	Nvarchar	5
STUDENT_ID	Nvarchar	20
BIRTHDATE	Nvarchar	8
LASTNAME	Nvarchar	30
FIRSTNAME	Nvarchar	30

### CENT-EXTRACT-IMPORT

#### Columns

Name	Type	Size
SCHOOL_ID	Nvarchar	255
STUDENT_ID	Nvarchar	255
BIRTHDATE	Nvarchar	255
LASTNAME	Nvarchar	255
FIRSTNAME	Nvarchar	255

### SCHOOL-EXTRACT

#### Columns

Name	Type	Size
SCHOOL_NAME	Nvarchar	100
PRINCIPAL	Nvarchar	100
SCHOOL_ID	Nvarchar	10

### SCHOOL-EXTRACT-IMPORT

#### Columns

Name	Type	Size
SCHOOL_NAME	Nvarchar	255
PRINCIPAL	Nvarchar	255
SCHOOL_ID	Nvarchar	10

### STBL-EXTRACT

#### Columns

Name	Type	Size
TYPE	Nvarchar	10
CODE	Nvarchar	10
DATA	Nvarchar	255

### STBL-EXTRACT-IMPORT

#### Columns

Name	Type	Size
TYPE	Nvarchar	255
CODE	Nvarchar	255
DATA	Nvarchar	255

## STUD-BILG

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	5
STUDENT_ID	Nvarchar	20
HOME	Nvarchar	3
PRIM	Nvarchar	3
COUNTRY	Nvarchar	3
BILG_CD	Nvarchar	3

## STUD-BILG-IMPORT

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	255
STUDENT_ID	Nvarchar	255
HOME	Nvarchar	255
PRIM	Nvarchar	255
COUNTRY	Nvarchar	255
BILG_CD	Nvarchar	255

## STUD-EXTRACT

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	5
STUDENT_ID	Nvarchar	20
STUD_STATUS	Nvarchar	2
LASTNAME	Nvarchar	30
FIRSTNAME	Nvarchar	30
MIDDLENAME	Nvarchar	30
LIVE_W_CD	Nvarchar	2
BILG_ELIG	Nvarchar	2
SPED_SVC	Nvarchar	2
UNLIST_PH	Nvarchar	2
AREACODE	Nvarchar	3
PHONE	Nvarchar	7
GRADE	Nvarchar	2
STUD_OPTION	Nvarchar	4
SEX	Nvarchar	2
BIRTH	Nvarchar	8
ETHNIC_CD	Nvarchar	2
CONF_CD	Nvarchar	2
PRIV_FG	Nvarchar	2
NO_CONTACT_NAME	Nvarchar	40
HOMELESS	Nvarchar	2
JOINT_CUSTODY	Nvarchar	2
HANDICAP_FG	Nvarchar	3
HR	Nvarchar	6
COUNSELOR	Nvarchar	30
SCHL_YR	Nvarchar	4
LOCKER_NBR	Nvarchar	10
LOCKER_COMBO	Nvarchar	10
ADMIT_DATE	Nvarchar	8
SCH_HRS	Nvarchar	2
YEAR_OF_GRAD	Nvarchar	4
SCHED_CHG_DATE	Nvarchar	8
MOD_DATE	Nvarchar	8
WITHDRAW_DATE	Nvarchar	8
WITHDRAW_CD	Nvarchar	3
TRANSFER_DATE	Nvarchar	8
XFER_SCH	Nvarchar	5
PREV_SCHOOL	Nvarchar	5
ACCESS_SCHOOL	Nvarchar	5



NUMBER_PADR	Nvarchar	2
P1_NAME	Nvarchar	30
P1_RELATIONSHIP	Nvarchar	2
P1_ADDRESS1	Nvarchar	80
P1_CITY	Nvarchar	30
P1_STATE	Nvarchar	4
P1_ZIP	Nvarchar	10
P1_M_ADDRESS1	Nvarchar	80
P1_M_CITY	Nvarchar	30
P1_M_STATE	Nvarchar	4
P1_M_ZIP	Nvarchar	10
P1_EMPLOYER	Nvarchar	40
P1_WK_AREACODE	Nvarchar	3
P1_WK_PHONE	Nvarchar	7
P1_WK_EXT	Nvarchar	10
P1_EMAIL	Nvarchar	40
P1_PAGER_AREACODE	Nvarchar	3
P1_PAGER_PHONE	Nvarchar	7
P2_NAME	Nvarchar	30
P2_RELATIONSHIP	Nvarchar	2
P2_ADDRESS1	Nvarchar	80
P2_CITY	Nvarchar	30
P2_STATE	Nvarchar	4
P2_ZIP	Nvarchar	10
P2_AREACODE	Nvarchar	3
P2_PHONE	Nvarchar	7
P2_EMPLOYER	Nvarchar	40
P2_WK_AREACODE	Nvarchar	3
P2_WK_PHONE	Nvarchar	7
P2_WK_EXT	Nvarchar	10
P2_EMAIL	Nvarchar	40
P2_PAGER_AREACODE	Nvarchar	3
P2_PAGER_PHONE	Nvarchar	7
P2_MAILING_CD	Nvarchar	2
P3_NAME	Nvarchar	30
P3_RELATIONSHIP	Nvarchar	2
P3_ADDRESS1	Nvarchar	80
P3_CITY	Nvarchar	30
P3_STATE	Nvarchar	4
P3_ZIP	Nvarchar	10
P3_AREACODE	Nvarchar	3
P3_PHONE	Nvarchar	7
P3_EMPLOYER	Nvarchar	40
P3_WK_AREACODE	Nvarchar	3
P3_WK_PHONE	Nvarchar	7
P3_WK_EXT	Nvarchar	10
P3_EMAIL	Nvarchar	40
P3_PAGER_AREACODE	Nvarchar	3
P3_PAGER_PHONE	Nvarchar	7
P3_MAILING_CD	Nvarchar	2
PHYS_NAME	Nvarchar	40
PHYS_AREACODE	Nvarchar	3
PHYS_PHONE	Nvarchar	7
HLTH_NAME	Nvarchar	40
HLTH_AREACODE	Nvarchar	3
HLTH_PHONE	Nvarchar	7
HLTH_CONCERN	Nvarchar	255
EC1_NAME	Nvarchar	50
EC1_AREACODE	Nvarchar	3
EC1_PHONE	Nvarchar	7
EC1_REL	Nvarchar	50
EC2_NAME	Nvarchar	50
EC2_AREACODE	Nvarchar	3
EC2_PHONE	Nvarchar	7
EC3_NAME	Nvarchar	50
EC3_AREACODE	Nvarchar	3
EC3_PHONE	Nvarchar	7
EC4_NAME	Nvarchar	50
EC4_AREACODE	Nvarchar	3
EC4_PHONE	Nvarchar	7

CP_NAME	Nvarchar	50
CP_AREACODE	Nvarchar	3
CP_PHONE	Nvarchar	7
CQ_NAME	Nvarchar	50
CQ_AREACODE	Nvarchar	3
CQ_PHONE	Nvarchar	7
STUD_AREACODE	Nvarchar	3
STUD_PHONE	Nvarchar	7
CASEWKR_NAME	Nvarchar	50
CASEWKR_AREACODE	Nvarchar	3
CASEWKR_PHONE	Nvarchar	7
DC_NAME	Nvarchar	50
DC_CONTACT	Nvarchar	40
DC_TITLE	Nvarchar	30
DC_AREACODE	Nvarchar	3
DC_PHONE	Nvarchar	7
DC_EXT	Nvarchar	10
DC_ADDRESS1	Nvarchar	40
DC_CITY_STATE	Nvarchar	40
DC_ZIP	Nvarchar	10
DC_FG	Nvarchar	3

STUD-EXTRACT-IMPORT

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	255
STUDENT_ID	Nvarchar	255
STUD_STATUS	Nvarchar	255
LASTNAME	Nvarchar	255
FIRSTNAME	Nvarchar	255
MIDDLENAME	Nvarchar	255
LIVE_W_CD	Nvarchar	255
BILG_ELIG	Nvarchar	255
SPED_SVC	Nvarchar	255
UNLIST_PH	Nvarchar	255
AREACODE	Nvarchar	255
PHONE	Nvarchar	255
GRADE	Nvarchar	255
STUD_OPTION	Nvarchar	255
SEX	Nvarchar	255
BIRTH	Nvarchar	255
ETHNIC_CD	Nvarchar	255
CONF_CD	Nvarchar	255
PRIV_FG	Nvarchar	255
NO_CONTACT_NAME	Nvarchar	255
HOMELESS	Nvarchar	255
JOINT_CUSTODY	Nvarchar	255
HANDICAP_FG	Nvarchar	255
HR	Nvarchar	255
COUNSELOR	Nvarchar	255
SCHL_YR	Nvarchar	255
LOCKER_NBR	Nvarchar	255
LOCKER_COMBO	Nvarchar	255
ADMIT_DATE	Nvarchar	255
SCH_HRS	Nvarchar	255
YEAR_OF_GRAD	Nvarchar	255
SCHED_CHG_DATE	Nvarchar	255
MOD_DATE	Nvarchar	255
WITHDRAW_DATE	Nvarchar	255
WITHDRAW_CD	Nvarchar	255
TRANSFER_DATE	Nvarchar	255
XFER_SCH	Nvarchar	255
PREV_SCHOOL	Nvarchar	255
ACCESS_SCHOOL	Nvarchar	255
NUMBER_PADR	Nvarchar	255
P1_NAME	Nvarchar	255
P1_RELATIONSHIP	Nvarchar	255

P1_ADDRESS1	Nvarchar	255
P1_CITY	Nvarchar	255
P1_STATE	Nvarchar	255
P1_ZIP	Nvarchar	255
P1_M_ADDRESS1	Nvarchar	255
P1_M_CITY	Nvarchar	255
P1_M_STATE	Nvarchar	255
P1_M_ZIP	Nvarchar	255
P1_EMPLOYER	Nvarchar	255
P1_WK_AREACODE	Nvarchar	255
P1_WK_PHONE	Nvarchar	255
P1_WK_EXT	Nvarchar	255
P1_EMAIL	Nvarchar	255
P1_PAGER_AREACODE	Nvarchar	255
P1_PAGER_PHONE	Nvarchar	255
P2_NAME	Nvarchar	255
P2_RELATIONSHIP	Nvarchar	255
P2_ADDRESS1	Nvarchar	255
P2_CITY	Nvarchar	255
P2_STATE	Nvarchar	255
P2_ZIP	Nvarchar	255
P2_AREACODE	Nvarchar	255
P2_PHONE	Nvarchar	255
P2_EMPLOYER	Nvarchar	255
P2_WK_AREACODE	Nvarchar	255
P2_WK_PHONE	Nvarchar	255
P2_WK_EXT	Nvarchar	255
P2_EMAIL	Nvarchar	255
P2_PAGER_AREACODE	Nvarchar	255
P2_PAGER_PHONE	Nvarchar	255
P2_MAILING_CD	Nvarchar	255
P3_NAME	Nvarchar	255
P3_RELATIONSHIP	Nvarchar	255
P3_ADDRESS1	Nvarchar	255
P3_CITY	Nvarchar	255
P3_STATE	Nvarchar	255
P3_ZIP	Nvarchar	255
P3_AREACODE	Nvarchar	255
P3_PHONE	Nvarchar	255
P3_EMPLOYER	Nvarchar	255
P3_WK_AREACODE	Nvarchar	255
P3_WK_PHONE	Nvarchar	255
P3_WK_EXT	Nvarchar	255
P3_EMAIL	Nvarchar	255
P3_PAGER_AREACODE	Nvarchar	255
P3_PAGER_PHONE	Nvarchar	255
P3_MAILING_CD	Nvarchar	255
PHYS_NAME	Nvarchar	255
PHYS_AREACODE	Nvarchar	255
PHYS_PHONE	Nvarchar	255
HLTH_NAME	Nvarchar	255
HLTH_AREACODE	Nvarchar	255
HLTH_PHONE	Nvarchar	255
HLTH_CONCERN	Nvarchar	255
EC1_NAME	Nvarchar	255
EC1_AREACODE	Nvarchar	255
EC1_PHONE	Nvarchar	255
EC1_REL	Nvarchar	255
EC2_NAME	Nvarchar	255
EC2_AREACODE	Nvarchar	255
EC2_PHONE	Nvarchar	255
EC3_NAME	Nvarchar	255
EC3_AREACODE	Nvarchar	255
EC3_PHONE	Nvarchar	255
EC4_NAME	Nvarchar	255
EC4_AREACODE	Nvarchar	255
EC4_PHONE	Nvarchar	255
CP_NAME	Nvarchar	255
CP_AREACODE	Nvarchar	255
CP_PHONE	Nvarchar	255

CQ_NAME	Nvarchar	255
CQ_AREACODE	Nvarchar	255
CQ_PHONE	Nvarchar	255
STUD_AREACODE	Nvarchar	255
STUD_PHONE	Nvarchar	255
CASEWKR_NAME	Nvarchar	255
CASEWKR_AREACODE	Nvarchar	255
CASEWKR_PHONE	Nvarchar	255
DC_NAME	Nvarchar	255
DC_CONTACT	Nvarchar	255
DC_TITLE	Nvarchar	255
DC_AREACODE	Nvarchar	255
DC_PHONE	Nvarchar	255
DC_EXT	Nvarchar	255
DC_ADDRESS1	Nvarchar	255
DC_CITY_STATE	Nvarchar	255
DC_ZIP	Nvarchar	255
DC_FG	Nvarchar	255

STUD-SEARCH

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	5
STUDENT_ID	Nvarchar	20
LASTNAME	Nvarchar	30
FIRSTNAME	Nvarchar	30
MIDDLENAME	Nvarchar	30
BIRTH	Nvarchar	8
SEX	Nvarchar	2
HR	Nvarchar	6
GRADE	Nvarchar	2
STUD_STATUS	Nvarchar	2
ACCESS_SCHOOL	Nvarchar	5

STUD-SPED

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	5
STUDENT_ID	Nvarchar	20
DIS_1	Nvarchar	5
DIS_2	Nvarchar	5
DIS_3	Nvarchar	5
DIS_4	Nvarchar	5
UNIT_CD	Nvarchar	3
SERVICE_CD	Nvarchar	3

STUD-SPED-IMPORT

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	255
STUDENT_ID	Nvarchar	255
DIS_1	Nvarchar	255
DIS_2	Nvarchar	255
DIS_3	Nvarchar	255
DIS_4	Nvarchar	255
UNIT_CD	Nvarchar	255
SERVICE_CD	Nvarchar	255

TEAC-DIVI

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	5
TEAC_ID	Nvarchar	20
TEAC_HR	Nvarchar	255

TEAC-DIVI-IMPORT

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	5
TEAC_ID	Nvarchar	20
TEAC_HR	Nvarchar	255

TEAC-EMPID

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	5
TEAC_ID	Nvarchar	5
EMP_ID	Nvarchar	255

TEAC-EMPID-IMPORT

**Columns**

Name	Type	Size
SCHOOL_ID	Nvarchar	5
TEAC_ID	Nvarchar	5
EMP_ID	Nvarchar	255

<<END OF DOCUMENT>>

```
1 package org.seattleschools.web.apps.tools;
2
3
4 import java.net.*;
5 import java.io.*;
6 import java.util.*;
7 import java.sql.*;
8 import java.lang.reflect.*;
9 import javax.servlet.*;
10 import javax.servlet.http.*;
11 import org.xml.sax.SAXException;
12 import org.xml.sax.InputSource;
13 import org.apache.xerces.parsers.SAXParser;
14 import org.seattleschools.web.apps.Page;
15 import org.seattleschools.web.apps.PageWrapper;
16 import org.seattleschools.web.apps.components.*;
17 import org.seattleschools.web.utils.WebSecurity;
18 import org.seattleschools.web.utils.DataValidator;
19 import org.apache.regexp.*;
20
21
22
23 /** This application is used to display, (only), Student Information to Teachers and School District Staff.
24 *
25 * The data originates in the current Student Information System and is exported to ASCII text files located on a network "share".
26 *
27 * These files are:
28 *
29 * CENT-EXTRACT.TXT - List of SchoolID, Student ID, Birthdate, First Name, Last Name (not used as of 3/2002)
30 * SCHOOL-EXTRACT.TXT - Relates School ID, School Name, Principal Name
31 * STBL-EXTRACT.TXT - Lookup tables for Student Info.
32 * STUD-BILG.TXT - Bi Lingual information for each student
33 * STUD-EXTRACT.TXT - Details for this student
34 * STUD-SPED.TXT - Special Ed details for each student
35 * TEAC-DIVI.TXT - Teacher to Home Room and School ID
36 * TEAC-EMPID.TXT - Relates Teacher Employee ID, School Teacher ID, and School ID
37 *
38 * These text files are then imported into a SQL database and conditioned for display through this application.
39 *
40 * The SQL 2000 database is called 'Teacher2SIS'
41 *
42 * The tables that this application access directly are:
43 *
44 *
45 *
46 * -----
47 * Created 2/2002 Daniel J. Moore
48 *
49 * REVISION HISTORY-----
50 *
51 *
52 *
53 *****/
54 public class StudentInfo extends Page {
55
56     private String dbDriver;
57     private String dbSQL2000Driver;
58     private String sql2000ServerDBsis;
59     private String sql2000AdminIdSis;
60     private String sql2000AdminPwSis;
61     private String strLogout;
62
63     //used to test if this individual can see all the schools...
64     private String strListAllSchoolID = "0799";
65
66
67     /** Initialize StudentInfo
68     * @param config ServletConfig
69     * @throws ServletException Exception Thrown
70     */
71     public void init(ServletConfig config) throws ServletException {
72         super.init(config);
73
74         System.out.println("Initializing StudentInfo");
75
76         dbDriver = (String)context.getAttribute("dbDriver");
77         //System.out.println(dbDriver);
78
79         dbSQL2000Driver = (String)context.getAttribute("dbSQL2000Driver");
80         //System.out.println(dbSQL2000Driver);
81         sql2000ServerDBsis = (String)context.getAttribute("sql2000ServerDBsis");
82         //System.out.println(sql2000ServerDBsis);
83         sql2000AdminIdSis = (String)context.getAttribute("sql2000AdminIdSis");
84         //System.out.println(sql2000AdminIdSis);
85         sql2000AdminPwSis = (String)context.getAttribute("sql2000AdminPwSis");
86         //System.out.println(sql2000AdminPwSis);
87         strLogout = (String)context.getAttribute("sislogout");
88
89
90     }
91
92     /** Send contents of this tool to Page Wrapper...
93     * @param request HttpServletRequest
94     * @param response HttpServletResponse
95     * @param showWrappers Array used to control Wrapper display, by setting the 0 (zero) index value to 0 (zero), the site wrapper will not be displayed.
96     * @return Sting containing tool contents...
97     */
98     protected String getContent(HttpServletRequest request, HttpServletResponse response, boolean[] showWrappers) {
```

```

99     showWrappers[0] = false;
100     String TeachName = "";
101     String TeachID = "";
102     StringBuffer content = new StringBuffer();
103     String method = request.getParameter("method");
104
105
106     content.append("<content>");
107     content.append("<title>Teacher Access to Student Information</title>\n");
108
109     if(validateUser("use", "StudentInfo", request, content)){
110
111         //-----This Teacher Info here from active directory -----
112         HttpSession session = request.getSession();
113
114         String ThisUser = (String)session.getAttribute("userId");
115
116         if(ThisUser.equalsIgnoreCase("SISTester")){
117             TeachName = "John W. Smith";
118             TeachID = "123456789";
119         }else{
120             TeachName = (String)session.getAttribute("givenName") + " " + (String)session.getAttribute("surname");
121             TeachID = (String)session.getAttribute("empID");
122         }
123         //-----
124
125         content.append("<teachersis>\n");
126         content.append("<teachername>"+TeachName+"</teachername>\n");
127         content.append("<teacherid>"+TeachID+"</teacherid>\n");
128         content.append("<version>VER. 1.0.1</version>\n");
129         content.append("<teacherschoollist>\n");
130
131         //GET THE SCHOOL LIST FOR THIS TEACHER...
132         content.append(getTeacherSchoolList(TeachID));
133
134
135         content.append("</teacherschoollist>\n");
136
137         if(request.getParameter("schoollist") != null){
138             content.append("<currentschool>"+request.getParameter("schoollist")+</currentschool>\n");
139         }else{
140             content.append("<currentschool>0</currentschool>\n");
141         }
142
143         if(method != null && method.length() > 0) {
144
145             if(method.equals("logout")) {
146                 request.getSession().removeAttribute("userId");
147                 request.getSession().removeAttribute("toolNav");
148                 content.append("<errmsg>Goodbye "+TeachName+", You have been Logged Off, and are being returned to Teacher Tools!!!</errmsg>\n");
149                 content.append("<arrTabLabelcontent>\n" + "\n"</arrTabLabelcontent>\n");
150                 content.append("<arrDisplayDivcontent>\nblankfill\n"</arrDisplayDivcontent>\n");
151                 content.append("<logout>"+strLogout+"</logout>\n");
152             }else{
153
154                 if(method.equals("studentsearch")) { //get serach results
155                     content.append(getSearchResults(request));
156                     content.append("<logout></logout>\n");
157                 }else{
158                     if(method.equals("getstudent")) { //get info for individual student
159                         content.append(getThisStudent(request));
160                         content.append("<logout></logout>\n");
161                     }else{//unknown method entered...
162                         content.append("<statusmsg>"+TeachName+", Welcome to Teacher Access to Student Information</statusmsg>\n");
163                         content.append("<arrTabLabelcontent>\n" + "\n"</arrTabLabelcontent>\n");
164                         content.append("<arrDisplayDivcontent>\nblankfill\n"</arrDisplayDivcontent>\n");
165                         content.append("<logout></logout>\n");
166                     }
167                 }
168             }
169         }
170     }
171     else { //initial screen
172         content.append("<statusmsg>"+TeachName+", Welcome to Teacher access to Student Information</statusmsg>\n");
173         content.append("<arrTabLabelcontent>\n" + "\n"</arrTabLabelcontent>\n");
174         content.append("<arrDisplayDivcontent>\nblankfill\n"</arrDisplayDivcontent>\n");
175         content.append("<logout></logout>\n");
176     }
177 }
178     content.append("</teachersis>\n");
179 }
180
181     content.append("</content>");
182
183     //System.out.println(content.toString());
184     return content.toString();
185 }
186
187
188 /** This Method is used return the XML results of the student info query entered by the teacher.
189 * @param request HttpServletRequest
190 * @return String
191 */
192 public String getSearchResults(HttpServletRequest request) {
193     DataValidator dv = new DataValidator();
194     StringBuffer content = new StringBuffer();
195     String strCurSchool = request.getParameter("schoollist");
196     String strFirst = request.getParameter("searchstufirst");

```

```

197 String strLast = request.getParameter("searchstulast");
198 String strHR = request.getParameter("searchstuhomerroom");
199 String strID = request.getParameter("searchstuid");
200 long lreccount = 0;
201
202 String sql = "SELECT * FROM [STUD-SEARCH] WHERE (([SCHOOL_ID]='"+request.getParameter("schoolist")+"' OR [ACCESS_SCHOOL]='"+request.getParameter("schoolist")+"')";
203
204 if(strFirst.length() > 0){ sql += " AND [FIRSTNAME]='"+strFirst+"'";}
205 if(strLast.length() > 0){ sql += " AND [LASTNAME]='"+strLast+"'";}
206 if(strHR.length() > 0){ sql += " AND [HR]='"+strHR+"'";}
207 if(strID.length() > 0){ sql += " AND [STUDENT_ID]='"+strID+"'";}
208
209 sql += " ORDER BY [LASTNAME]";
210
211 content.append("<searchfirstname>"+strFirst+"</searchfirstname>\n");
212 content.append("<searchlastname>"+strLast+"</searchlastname>\n");
213 content.append("<searchid>"+strID+"</searchid>\n");
214 content.append("<searchhr>"+strHR+"</searchhr>\n");
215 content.append("<searchresults>\n");
216
217 //System.out.println("StudentInfo.getSearchResults() -- sql = "+sql);
218
219 try {
220     Class.forName(dbSQL2000Driver);
221
222     Connection dbConSIS = DriverManager.getConnection(sql2000ServerDBsis,sql2000AdminIdSis,sql2000AdminPwSis);
223
224     Statement stSIS = dbConSIS.createStatement();
225     ResultSet rsSIS = stSIS.executeQuery(sql);
226
227     while(rsSIS.next()) {
228
229         content.append("<student>\n");
230         content.append("<id>"+ prepText(rsSIS.getString("STUDENT_ID")) + "</id>\n");//
231         content.append("<last>"+ prepText(rsSIS.getString("LASTNAME")) + "</last>\n");
232         content.append("<first>"+ prepText(rsSIS.getString("FIRSTNAME")) + "</first>\n");
233         content.append("<middle>"+ prepText(rsSIS.getString("MIDDLENAME")) + "</middle>\n");
234         content.append("<birth>"+ prepDate(rsSIS.getString("BIRTH")) + "</birth>\n");
235         if(rsSIS.getString("SEX").equalsIgnoreCase("m")){
236             content.append("<gender>MALE</gender>\n");
237         }
238         else{
239
240             if(rsSIS.getString("SEX").equalsIgnoreCase("f")){
241                 content.append("<gender>FEMALE</gender>\n");
242             }
243             else{
244                 content.append("<gender>"+ prepText(rsSIS.getString("SEX")) + "</gender>\n");
245             }
246         }
247         content.append("<hr>"+ prepText(rsSIS.getString("HR")) + "</hr>\n");
248         content.append("<grade>"+ prepText(rsSIS.getString("GRADE")) + "</grade>\n");
249
250         if(strCurSchool.equalsIgnoreCase(rsSIS.getString("SCHOOL_ID"))){
251             content.append("<status r=\"1\">"+getItemDetails("SIS", rsSIS.getString("STUD_STATUS"))+"</status>\n");//
252             content.append("<cursorchool r=\"1\">"+ getSchoolName(rsSIS.getString("SCHOOL_ID")) + "</cursorchool>\n");
253         }
254         else{
255             content.append("<status r=\"1\">"+getItemDetails("SIS", rsSIS.getString("STUD_STATUS"))+"</status>\n");//
256             content.append("<cursorchool r=\"1\">"+ getSchoolName(rsSIS.getString("SCHOOL_ID")) + "</cursorchool>\n");
257         }
258         //
259         content.append("</student>\n");
260
261         lreccount++;
262     }
263
264     stSIS.close();
265     dbConSIS.close();
266     Class.forName(dbDriver);
267 }
268 catch(ClassNotFoundException cnfe) {
269     System.out.println("ClassNotFoundException thrown in StudentInfo.getSearchResults(): " + cnfe.getMessage());
270     content.append("</searchresults>\n");
271     content.append("<errmsg>ClassNotFoundException thrown in StudentInfo.getSearchResults(): "+cnfe.getMessage()+"</errmsg>\n");
272     content.append("<arrTabLabelcontent>\n" + "</arrTabLabelcontent>\n");
273     content.append("<arrDisplayDivcontent>\n\"blankfill\"</arrDisplayDivcontent>\n");
274     return content.toString();
275 }
276 catch(SQLException sqle) {
277     System.out.println("SQLException thrown in StudentInfo.getSearchResults(): " + sqle.getMessage());
278     content.append("</searchresults>\n");
279     content.append("<errmsg>SQLException thrown in StudentInfo.getSearchResults(): "+sqle.getMessage()+"</errmsg>\n");
280     content.append("<arrTabLabelcontent>\n" + "</arrTabLabelcontent>\n");
281     content.append("<arrDisplayDivcontent>\n\"blankfill\"</arrDisplayDivcontent>\n");
282     try{
283         Class.forName(dbDriver);// make sure driver reset to mySql
284     }
285     catch(ClassNotFoundException cnfe) {
286         System.out.println("ClassNotFoundException thrown in StudentInfo.getSearchResults(): " + cnfe.getMessage());
287     }
288     return content.toString();
289 }
290
291 content.append("<recordcount>"+lreccount+"</recordcount>\n");
292 content.append("</searchresults>\n");
293
294 content.append("<statusmsg></statusmsg>\n");

```



```

295     content.append("<arrTabLabelcontent>\SEARCH_RESULTS\</arrTabLabelcontent>\n");
296     content.append("<arrDisplayDivcontent>\StudentsFoundList\</arrDisplayDivcontent>\n");//statusmsg
297
298
299     return content.toString();
300 }
301
302 /** This method is used to return the XML used to display the desired student information.
303  * @param request HttpServletRequest
304  * @return String
305  */
306 public String getThisStudent(HttpServletRequest request) {
307     DataValidator dv = new DataValidator();
308     StringBuffer content = new StringBuffer();
309     String strGetStudID = request.getParameter("studentfind");
310     String strFirst = request.getParameter("searchstufirst");
311     String strLast = request.getParameter("searchstulast");
312     String strHR = request.getParameter("searchstuhomerom");
313     String strID = request.getParameter("searchstuid");
314     String DCTest = "";
315     int iConf = 0;
316
317     String sql = "SELECT * FROM [STUD-EXTRACT] WHERE [STUDENT_ID]='"+strGetStudID+"'";
318
319     content.append("<searchfirstname>"+strFirst+"</searchfirstname>\n");
320     content.append("<searchlastname>"+strLast+"</searchlastname>\n");
321     content.append("<searchid>"+strID+"</searchid>\n");
322     content.append("<searchhr>"+strHR+"</searchhr>\n");
323
324     //System.out.println("StudentInfo.getThisStudent() -- sql = "+sql);
325
326     try {
327         Class.forName(dbSQL2000Driver);
328
329
330         Connection dbConSIS = DriverManager.getConnection(sql2000ServerDBsis,sql2000AdminIdSis,sql2000AdminPwSis);
331
332         Statement stSIS = dbConSIS.createStatement();
333         ResultSet rsSIS = stSIS.executeQuery(sql);
334
335         while(rsSIS.next()) {
336
337             content.append("<currentstudent>\n");
338             content.append("<studentid>"+ prepText(rsSIS.getString("STUDENT_ID")) + "</studentid>\n");
339             content.append("<studentname>"+ prepText(rsSIS.getString("LASTNAME")) + ", " + prepText(rsSIS.getString("FIRSTNAME")) + " " + prepText(rsSIS.getString("MIDDLENAME")) + "</studentname>\n");
340             content.append("<studentthr>"+ prepText(rsSIS.getString("HR")) + "</studentthr>\n");
341             content.append("<moddate>"+ prepDate(rsSIS.getString("MOD_DATE")) + "</moddate>\n");
342             content.append("</currentstudent>\n");
343
344             content.append("<specialinfo>\n");
345             if(rsSIS.getString("NO_CONTACT_NAME") !=null){
346                 if(rsSIS.getString("NO_CONTACT_NAME").length()>0 && !rsSIS.getString("NO_CONTACT_NAME").equalsIgnoreCase("null")){
347                     content.append("<nocontact>"+ prepText(rsSIS.getString("NO_CONTACT_NAME").trim()) + "</nocontact>\n");
348                 }
349             }
350             if(rsSIS.getString("HLTH_CONCERN") !=null){
351                 if(rsSIS.getString("HLTH_CONCERN").length()>0 && !rsSIS.getString("HLTH_CONCERN").equalsIgnoreCase("null")){
352                     content.append("<health>"+ prepText(rsSIS.getString("HLTH_CONCERN").trim()) + "</health>\n");
353                 }
354             }
355
356             if(rsSIS.getString("CONF_CD") !=null){
357                 if(rsSIS.getString("CONF_CD").length()>0 && rsSIS.getString("CONF_CD").equalsIgnoreCase("y")){
358                     content.append("<confidential/>\n");
359                     iConf = 1;
360                 }
361             }
362
363             content.append("</specialinfo>\n");
364
365             content.append("<studentinfo>\n");
366
367             content.append("<grade>"+ prepText(rsSIS.getString("GRADE")) + "</grade>\n");
368             content.append("<birth>"+ prepDate(rsSIS.getString("BIRTH")) + "</birth>\n");
369
370             if(rsSIS.getString("SEX").equalsIgnoreCase("m")){
371                 content.append("<gender>MALE</gender>\n");
372             }
373             else{
374                 if(rsSIS.getString("SEX").equalsIgnoreCase("f")){
375                     content.append("<gender>FEMALE</gender>\n");
376                 }
377             }
378             content.append("<gender>"+ prepText(rsSIS.getString("SEX")) + "</gender>\n");
379
380         }
381     }
382
383     content.append("<ethnicity>"+getItemDetails("ETH", rsSIS.getString("ETHNIC_CD"))+"</ethnicity>\n");//
384
385     content.append("<option>"+getItemDetails("I20", rsSIS.getString("STUD_OPTION"))+"</option>\n");//
386     content.append("<gradyear>"+ prepText(rsSIS.getString("YEAR_OF_GRAD")) + "</gradyear>\n");
387
388     /*" + prepText(rsSIS.getString("WITHDRAW_CD")) + "
389     if(rsSIS.getString("STUD_STATUS").equalsIgnoreCase("w")){
390         content.append("<displaydate type='Withdraw Date'\>"+ prepDate(rsSIS.getString("WITHDRAW_DATE")) + "</displaydate>\n");
391     }
392     else{

```

```

393     content.append("<displaydate type=\"Admit Date\">\" + prepDate(rsSIS.getString(\"ADMIT_DATE\")) + "</displaydate>\n");
394 }
395
396 content.append("<status>"+getItemDetails("SIS", rsSIS.getString("STUD_STATUS"))+"</status>\n");//
397 content.append("<cursschool>\" + getSchoolName(rsSIS.getString(\"SCHOOL_ID\")) + "</cursschool>\n");//
398 content.append("<prevschool>\" + getSchoolName(rsSIS.getString(\"PREV_SCHOOL\")) + "</prevschool>\n");//
399
400 if(iConf == 1){//confidential student...
401
402     content.append("<paddress>\n");
403     content.append("<street></street>\n");
404     content.append("<city></city>\n");
405     content.append("<state></state>\n");
406     content.append("<zip></zip>\n");
407     content.append("</paddress>\n");
408
409     content.append("<pmailaddress>\n");
410     content.append("<street></street>\n");
411     content.append("<city></city>\n");
412     content.append("<state></state>\n");
413     content.append("<zip></zip>\n");
414     content.append("</pmailaddress>\n");
415
416     content.append("<stuphone><area></area><phone></phone></stuphone>\n");
417     content.append("<livingwith></livingwith>\n");
418     content.append("<joint></joint>\n");
419
420     content.append("<privacy>y</privacy>\n");
421
422     content.append("<homeless></homeless>\n");
423     content.append("<caseworker>\" + prepText(rsSIS.getString(\"CASEWKR_NAME\")) + "</caseworker>\n");
424     content.append("<cwphone><area>\" + prepText(rsSIS.getString(\"CASEWKR_AREACODE\")) + "</area><phone>\" + prepText(rsSIS.getString(\"CASEWKR_PHONE\")) + "</phone></cwphone>\n");
425
426 }else{
427
428     content.append("<paddress>\n");
429     content.append("<street>\" + prepText(rsSIS.getString(\"P1_ADDRESS1\")) + "</street>\n");
430     content.append("<city>\" + prepText(rsSIS.getString(\"P1_CITY\")) + "</city>\n");
431     content.append("<state>\" + prepText(rsSIS.getString(\"P1_STATE\")) + "</state>\n");
432     content.append("<zip>\" + prepText(rsSIS.getString(\"P1_ZIP\")) + "</zip>\n");
433     content.append("</paddress>\n");
434
435     content.append("<pmailaddress>\n");
436     content.append("<street>\" + prepText(rsSIS.getString(\"P1_M_ADDRESS1\")) + "</street>\n");
437     content.append("<city>\" + prepText(rsSIS.getString(\"P1_M_CITY\")) + "</city>\n");
438     content.append("<state>\" + prepText(rsSIS.getString(\"P1_M_STATE\")) + "</state>\n");
439     content.append("<zip>\" + prepText(rsSIS.getString(\"P1_M_ZIP\")) + "</zip>\n");
440     content.append("</pmailaddress>\n");
441
442     content.append("<stuphone><area>\" + prepText(rsSIS.getString(\"STUD_AREACODE\")) + "</area><phone>\" + prepText(rsSIS.getString(\"STUD_PHONE\")) + "</phone></stuphone>\n");
443     content.append("<livingwith>"+getItemDetails("LWC", rsSIS.getString("LIVE_W_CD"))+"</livingwith>\n");//
444     content.append("<joint>\" + prepText(rsSIS.getString(\"JOINT_CUSTODY\")) + "</joint>\n");
445
446     content.append("<privacy>\" + prepText(rsSIS.getString(\"PRIV_FG\")) + "</privacy>\n");
447
448     content.append("<homeless>\" + prepText(rsSIS.getString(\"HOMELESS\")) + "</homeless>\n");
449     content.append("<caseworker>\" + prepText(rsSIS.getString(\"CASEWKR_NAME\")) + "</caseworker>\n");
450     content.append("<cwphone><area>\" + prepText(rsSIS.getString(\"CASEWKR_AREACODE\")) + "</area><phone>\" + prepText(rsSIS.getString(\"CASEWKR_PHONE\")) + "</phone></cwphone>\n");
451 }
452
453 content.append("<five04>"+getItemDetails("504", rsSIS.getString("HANDICAP_FG"))+"</five04>\n");//
454
455 content.append(getBILG(strGetStudID));
456
457 content.append("<level>"+getItemDetails("SPE", rsSIS.getString("SPED_SVC"))+"</level>\n");//
458 content.append(getSPED(strGetStudID));
459
460 content.append("</studentinfo>\n");
461
462 if(iConf == 0){
463
464     content.append("<parentinfo>\n");
465
466     content.append("<p1name>\" + prepText(rsSIS.getString(\"P1_NAME\")) + "</p1name>\n");
467     content.append("<p1relation>"+getItemDetails("PRL", rsSIS.getString("P1_RELATIONSHIP"))+"</p1relation>\n");//
468     content.append("<p1phone><area>\" + prepText(rsSIS.getString(\"AREACODE\")) + "</area><phone>\" + prepText(rsSIS.getString(\"PHONE\")) + "</phone></p1phone>\n");
469     content.append("<p1unlisted>\" + prepText(rsSIS.getString(\"UNLIST_PH\")) + "</p1unlisted>\n");
470     content.append("<p1email>\" + dv.stripSpaces(prepareText(rsSIS.getString(\"P1_EMAIL\"))) + "</p1email>\n");
471     content.append("<p1employer>\" + prepText(rsSIS.getString(\"P1_EMPLOYER\")) + "</p1employer>\n");
472     content.append("<p1workphone><area>\" + prepText(rsSIS.getString(\"P1_WK_AREACODE\")) + "</area><phone>\" + prepText(rsSIS.getString(\"P1_WK_PHONE\")) + "</phone></p1workphone>\n");
473     content.append("<p1ext>\" + prepText(rsSIS.getString(\"P1_WK_EXT\")) + "</p1ext>\n");
474     content.append("<p1pager><area>\" + prepText(rsSIS.getString(\"P1_PAGER_AREACODE\")) + "</area><phone>\" + prepText(rsSIS.getString(\"P1_PAGER_PHONE\")) + "</phone></p1pager>\n");
475     content.append("<paddress>\n");
476     content.append("<street>\" + prepText(rsSIS.getString(\"P1_ADDRESS1\")) + "</street>\n");
477     content.append("<city>\" + prepText(rsSIS.getString(\"P1_CITY\")) + "</city>\n");
478     content.append("<state>\" + prepText(rsSIS.getString(\"P1_STATE\")) + "</state>\n");
479     content.append("<zip>\" + prepText(rsSIS.getString(\"P1_ZIP\")) + "</zip>\n");
480     content.append("</paddress>\n");
481     content.append("<pmailaddress>\n");
482     content.append("<street>\" + prepText(rsSIS.getString(\"P1_M_ADDRESS1\")) + "</street>\n");
483     content.append("<city>\" + prepText(rsSIS.getString(\"P1_M_CITY\")) + "</city>\n");
484     content.append("<state>\" + prepText(rsSIS.getString(\"P1_M_STATE\")) + "</state>\n");
485     content.append("<zip>\" + prepText(rsSIS.getString(\"P1_M_ZIP\")) + "</zip>\n");
486     content.append("</pmailaddress>\n");
487
488     content.append("<p2name>\" + prepText(rsSIS.getString(\"P2_NAME\")) + "</p2name>\n");
489     content.append("<p2relation>"+getItemDetails("PRL", rsSIS.getString("P2_RELATIONSHIP"))+"</p2relation>\n");
490

```

```

491 content.append("<p2phone><area>" + prepText(rsSIS.getString("P2_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("P2_PHONE")) + "</phone></p2phone>\n");
492 content.append("<p2email>" + dv.stripSpaces(prepareText(rsSIS.getString("P2_EMAIL"))) + "</p2email>\n");
493 content.append("<p2employer>" + prepText(rsSIS.getString("P2_EMPLOYER")) + "</p2employer>\n");
494 content.append("<p2workphone><area>" + prepText(rsSIS.getString("P2_WK_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("P2_WK_PHONE")) + "</phone></p2workphone>\n");
495 content.append("<p2ext>" + prepText(rsSIS.getString("P2_WK_EXT")) + "</p2ext>\n");
496 content.append("<p2pager><area>" + prepText(rsSIS.getString("P2_PAGER_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("P2_PAGER_PHONE")) + "</phone></p2pager>\n");
497 content.append("<p2address>\n");
498 content.append("<street>" + prepText(rsSIS.getString("P2_ADDRESS1")) + "</street>\n");
499 content.append("<city>" + prepText(rsSIS.getString("P2_CITY")) + "</city>\n");
500 content.append("<state>" + prepText(rsSIS.getString("P2_STATE")) + "</state>\n");
501 content.append("<zip>" + prepText(rsSIS.getString("P2_ZIP")) + "</zip>\n");
502 content.append("</p2address>\n");
503 content.append("<p2mailhere>" + prepText(rsSIS.getString("P2_MAILING_CD")) + "</p2mailhere>\n");
504
505 content.append("<p3name>" + prepText(rsSIS.getString("P3_NAME")) + "</p3name>\n");
506 content.append("<p3relation>" + getItemDetails("PRL", rsSIS.getString("P3_RELATIONSHIP")) + "</p3relation>\n");
507 content.append("<p3phone><area>" + prepText(rsSIS.getString("P3_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("P3_PHONE")) + "</phone></p3phone>\n");
508 content.append("<p3email>" + dv.stripSpaces(prepareText(rsSIS.getString("P3_EMAIL"))) + "</p3email>\n");
509 content.append("<p3employer>" + prepText(rsSIS.getString("P3_EMPLOYER")) + "</p3employer>\n");
510 content.append("<p3workphone><area>" + prepText(rsSIS.getString("P3_WK_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("P3_WK_PHONE")) + "</phone></p3workphone>\n");
511 content.append("<p3ext>" + prepText(rsSIS.getString("P3_WK_EXT")) + "</p3ext>\n");
512 content.append("<p3pager><area>" + prepText(rsSIS.getString("P3_PAGER_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("P3_PAGER_PHONE")) + "</phone></p3pager>\n");
513 content.append("<p3address>\n");
514 content.append("<street>" + prepText(rsSIS.getString("P3_ADDRESS1")) + "</street>\n");
515 content.append("<city>" + prepText(rsSIS.getString("P3_CITY")) + "</city>\n");
516 content.append("<state>" + prepText(rsSIS.getString("P3_STATE")) + "</state>\n");
517 content.append("<zip>" + prepText(rsSIS.getString("P3_ZIP")) + "</zip>\n");
518 content.append("</p3address>\n");
519 content.append("<p3mailhere>" + prepText(rsSIS.getString("P3_MAILING_CD")) + "</p3mailhere>\n");
520
521 content.append("<cpname>" + prepText(rsSIS.getString("CP_NAME")) + "</cpname>\n");
522 content.append("<cpphone><area>" + prepText(rsSIS.getString("CP_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("CP_PHONE")) + "</phone></cpphone>\n");
523 content.append("<cqname>" + prepText(rsSIS.getString("CQ_NAME")) + "</cqname>\n");
524 content.append("<cqphone><area>" + prepText(rsSIS.getString("CQ_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("CQ_PHONE")) + "</phone></cqphone>\n");
525
526 content.append("</parentinfo>\n");
527
528 content.append("<emergencyinfo>\n");
529 content.append("<contactname1>" + prepText(rsSIS.getString("EC1_NAME")) + "</contactname1>\n");
530 content.append("<contactphone1><area>" + prepText(rsSIS.getString("EC1_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("EC1_PHONE")) + "</phone></contactphone1>\n");
531 content.append("<relationship1>" + prepText(rsSIS.getString("EC1_REL")) + "</relationship1>\n");
532 content.append("<contactname2>" + prepText(rsSIS.getString("EC2_NAME")) + "</contactname2>\n");
533 content.append("<contactphone2><area>" + prepText(rsSIS.getString("EC2_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("EC2_PHONE")) + "</phone></contactphone2>\n");
534 content.append("<contactname3>" + prepText(rsSIS.getString("EC3_NAME")) + "</contactname3>\n");
535 content.append("<contactphone3><area>" + prepText(rsSIS.getString("EC3_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("EC3_PHONE")) + "</phone></contactphone3>\n");
536 content.append("<contactname4>" + prepText(rsSIS.getString("EC4_NAME")) + "</contactname4>\n");
537 content.append("<contactphone4><area>" + prepText(rsSIS.getString("EC4_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("EC4_PHONE")) + "</phone></contactphone4>\n");
538 content.append("</emergencyinfo>\n");
539
540 content.append("<daycareinfo>\n");
541
542 content.append("<center>" + prepText(rsSIS.getString("DC_NAME")) + "</center>\n");
543 content.append("<contact>" + prepText(rsSIS.getString("DC_CONTACT")) + "</contact>\n");
544 content.append("<title>" + prepText(rsSIS.getString("DC_TITLE")) + "</title>\n");
545 content.append("<phonenumber><area>" + prepText(rsSIS.getString("DC_AREACODE")) + "</area><phone>" + prepText(rsSIS.getString("DC_PHONE")) + "</phone></phonenumber>\n");
546 content.append("<ext>" + prepText(rsSIS.getString("DC_EXT")) + "</ext>\n");
547 content.append("<address>\n");
548 content.append("<street>" + prepText(rsSIS.getString("DC_ADDRESS1")) + "</street>\n");
549 content.append("<state>" + prepText(rsSIS.getString("DC_CITY_STATE")) + "</state>\n");
550 content.append("<zip>" + prepText(rsSIS.getString("DC_ZIP")) + "</zip>\n");
551 content.append("</address>\n");
552
553
554 DCtest = rsSIS.getString("DC_FG").trim();
555
556 content.append("</daycareinfo>\n");
557
558 }
559
560 }
561
562 stSIS.close();
563 dbConSIS.close();
564 Class.forName(dbDriver);
565 }
566
567 catch(ClassNotFoundException cnfe) {
568 System.out.println("ClassNotFoundException thrown in StudentInfo.getThisStudent(): " + cnfe.getMessage());
569 content.append("<errmsg>ClassNotFoundException thrown in StudentInfo.getThisStudent(): "+cnfe.getMessage()+"</errmsg>\n");
570 content.append("<arrTabLabelcontent>\n" + "\</arrTabLabelcontent>\n");
571 content.append("<arrDisplayDivcontent>\"blankfill\"</arrDisplayDivcontent>\n");
572 return content.toString();
573 }
574
575 catch(SQLException sqle) {
576 System.out.println("SQLException thrown in StudentInfo.getThisStudent(): " + sqle.getMessage());
577 content.append("<errmsg>SQLException thrown in StudentInfo.getThisStudent(): "+sqle.getMessage()+"</errmsg>\n");
578 content.append("<arrTabLabelcontent>\n" + "\</arrTabLabelcontent>\n");
579 content.append("<arrDisplayDivcontent>\"blankfill\"</arrDisplayDivcontent>\n");
580 try{
581 Class.forName(dbDriver);// make sure driver reset to mySql
582 }
583 catch(ClassNotFoundException cnfe) {
584 System.out.println("ClassNotFoundException thrown in StudentInfo.getThisStudent(): " + cnfe.getMessage());
585 }
586 return content.toString();
587 }
588

```

```

589 //set xsl tab arrays...
590 if (DCTest.length() > 0 && DCTest.equalsIgnoreCase("y")){//daycare flag set
591     if(iConf == 1){//confidential student
592         content.append("<arrTabLabelcontent>\\"STUDENT\\"</arrTabLabelcontent>\n");
593         content.append("<arrDisplayDivcontent>\\"StuInfo\\"</arrDisplayDivcontent>\n");
594     }else{
595         content.append("<arrTabLabelcontent>\\"STUDENT\\", \\"PARENT/GUARDIAN\\", \\"EMERGENCY CONTACTS\\", \\"DAY CARE\\"</arrTabLabelcontent>\n");
596         content.append("<arrDisplayDivcontent>\\"StuInfo\\",\\"ParInfo\\",\\"EmergInfo\\", \\"DayCareInfo\\"</arrDisplayDivcontent>\n");
597     }
598 }
599 }else{
600     if(iConf == 1){//confidential student
601         content.append("<arrTabLabelcontent>\\"STUDENT\\"</arrTabLabelcontent>\n");
602         content.append("<arrDisplayDivcontent>\\"StuInfo\\"</arrDisplayDivcontent>\n");
603     }else{
604         content.append("<arrTabLabelcontent>\\"STUDENT\\", \\"PARENT/GUARDIAN\\", \\"EMERGENCY CONTACTS\\"</arrTabLabelcontent>\n");
605         content.append("<arrDisplayDivcontent>\\"StuInfo\\",\\"ParInfo\\",\\"EmergInfo\\"</arrDisplayDivcontent>\n");
606     }
607 }
608 }
609 //System.out.println(content.toString());
610 return content.toString();
611
612
613
614 /* -----available in STUD-EXTRACT but currently not used
615  * + rsSIS.getString("SCHOOL_ID") + "
616  * + rsSIS.getString("COUNSELOR") + "
617  * + rsSIS.getString("SCHL_YR") + "
618  * + rsSIS.getString("LOCKER_NBR") + "
619  * + rsSIS.getString("LOCKER_COMBO") + "
620  * + rsSIS.getString("SCH_HRS") + "
621  * + rsSIS.getString("SCHED_CHG_DATE") + "
622  * + rsSIS.getString("TRANSFER_DATE") + "
623  * + rsSIS.getString("XFER_SCH") + "
624  * + rsSIS.getString("PHYS_NAME") + "
625  * + rsSIS.getString("PHYS_AREACODE") + "
626  * + rsSIS.getString("PHYS_PHONE") + "
627  * + rsSIS.getString("HLTH_NAME") + "
628  * + rsSIS.getString("HLTH_AREACODE") + "
629  * + rsSIS.getString("HLTH_PHONE") + "
630  * ----- */
631 }
632
633
634
635 /** Used to retrieve XML data regarding Special Education Details
636  * @param strStudentID Student Id (String)
637  * @return String
638  */
639 public String getSPED(String strStudentID) {
640     StringBuffer content = new StringBuffer();
641     int iFound = 0;
642
643     String sql = "SELECT * FROM [STUD-SPED] WHERE [STUDENT_ID]='"+strStudentID+"'";
644
645
646     try {
647         Class.forName(dbSQL2000Driver);
648
649
650         Connection dbConSIS = DriverManager.getConnection(sql2000ServerDBsis,sql2000AdminIdSis,sql2000AdminPwSis);
651
652         Statement stSIS = dbConSIS.createStatement();
653         ResultSet rsSIS = stSIS.executeQuery(sql);
654
655         while(rsSIS.next()) {
656
657             content.append("<disabilities>\n");
658             if(rsSIS.getString("DIS_1") != null){
659                 content.append("<db><code>" + prepText(rsSIS.getString("DIS_1").trim()) + "</code><desc>" + getItemDetails("DIS", rsSIS.getString("DIS_1"))+"</desc></db>\n");
660             }
661             if(rsSIS.getString("DIS_2") != null){
662                 content.append("<db><code>" + prepText(rsSIS.getString("DIS_2").trim()) + "</code><desc>" + getItemDetails("DIS", rsSIS.getString("DIS_2"))+"</desc></db>\n");
663             }
664             if(rsSIS.getString("DIS_3") != null){
665                 content.append("<db><code>" + prepText(rsSIS.getString("DIS_3").trim()) + "</code><desc>" + getItemDetails("DIS", rsSIS.getString("DIS_3"))+"</desc></db>\n");
666             }
667             if(rsSIS.getString("DIS_4") != null){
668                 content.append("<db><code>" + prepText(rsSIS.getString("DIS_4").trim()) + "</code><desc>" + getItemDetails("DIS", rsSIS.getString("DIS_4"))+"</desc></db>\n");
669             }
670             content.append("</disabilities>\n");
671             content.append("<svctype>" + getItemDetails("SPE", rsSIS.getString("SERVICE_CD"))+"</svctype>\n");
672             content.append("<unit>" + getItemDetails("UNI", rsSIS.getString("UNIT_CD"))+"</unit>\n");
673             iFound++;
674         }
675
676         if(iFound == 0){//no records found..
677
678             content.setLength(0);
679
680             content.append("<disabilities>\n");
681             content.append("</disabilities>\n");
682             content.append("<svctype></svctype>\n");
683             content.append("<unit></unit>\n");
684
685         }
686     }

```

```

687         stSIS.close();
688         dbConSIS.close();
689         Class.forName(dbDriver);
690     }
691     catch(ClassNotFoundException cnfe) {
692         System.out.println("ClassNotFoundException thrown in StudentInfo.getSPED(): " + cnfe.getMessage());
693     }
694     content.setLength(0);
695
696     content.append("<disabilities>\n");
697     content.append("</disabilities>\n");
698     content.append("<svctype></svctype>\n");
699     content.append("<unit></unit>\n");
700
701     return content.toString();
702 }
703
704 catch(SQLException sqle) {
705     System.out.println("SQLException thrown in StudentInfo.getSPED(): " + sqle.getMessage());
706 }
707 content.setLength(0);
708
709 content.append("<disabilities>\n");
710 content.append("</disabilities>\n");
711 content.append("<svctype></svctype>\n");
712 content.append("<unit></unit>\n");
713
714 try{
715     Class.forName(dbDriver);// make sure driver reset to mySql
716 }
717 catch(ClassNotFoundException cnfe) {
718     System.out.println("ClassNotFoundException thrown in StudentInfo.getSPED(): " + cnfe.getMessage());
719 }
720
721 return content.toString();
722 }
723
724 return content.toString();
725 }
726
727
728 /** Used to retrieve XML Bilingual information for the student.
729  * @param strStudentID Student ID (String)
730  * @return String
731  */
732 public String getBILG(String strStudentID) {
733     StringBuffer content = new StringBuffer();
734     int iFound = 0;
735
736     String sql = "SELECT * FROM [STUD-BILG] WHERE [STUDENT_ID]='"+strStudentID+"'";
737
738
739     try {
740         Class.forName(dbSQL2000Driver);
741
742
743         Connection dbConSIS = DriverManager.getConnection(sql2000ServerDBsis,sql2000AdminIdSis,sql2000AdminPwSis);
744
745         Statement stSIS = dbConSIS.createStatement();
746         ResultSet rsSIS = stSIS.executeQuery(sql);
747
748         while(rsSIS.next()) {
749
750
751             content.append("<bielig>"+getItemDetails("ELG", rsSIS.getString("BILG_CD"))+"</bielig>\n");//
752             content.append("<country>"+getItemDetails("CTC", rsSIS.getString("COUNTRY"))+"</country>\n");//
753             content.append("<pri>"+getItemDetails("ING", rsSIS.getString("PRIM"))+"</pri>\n");//
754             content.append("<home>"+getItemDetails("LNG", rsSIS.getString("HOME"))+"</home>\n");//
755             iFound++;
756         }
757
758
759         if(iFound == 0){//no records found...
760
761             content.setLength(0);
762
763             content.append("<bielig></bielig>\n");
764             content.append("<country></country>\n");
765             content.append("<pri></pri>\n");
766             content.append("<home></home>\n");
767
768         }
769     }
770
771     stSIS.close();
772     dbConSIS.close();
773     Class.forName(dbDriver);
774 }
775
776 catch(ClassNotFoundException cnfe) {
777     System.out.println("ClassNotFoundException thrown in StudentInfo.getBILG(): " + cnfe.getMessage());
778 }
779 content.setLength(0);
780
781 content.append("<bielig></bielig>\n");
782 content.append("<country></country>\n");
783 content.append("<pri></pri>\n");
784 content.append("<home></home>\n");

```

```

785         return content.toString();
786     }
787 }
788 }
789 catch(SQLException sqle) {
790     System.out.println("SQLException thrown in StudentInfo.getBILG(): " + sqle.getMessage());
791 }
792 content.setLength(0);
793
794 content.append("<bielig></bielig>\n");
795 content.append("<country></country>\n");
796 content.append("<pri></pri>\n");
797 content.append("<home></home>\n");
798
799 try{
800     Class.forName(dbDriver);// make sure driver reset to mySql
801 }
802 catch(ClassNotFoundException cnfe) {
803     System.out.println("ClassNotFoundException thrown in StudentInfo.getBILG(): " + cnfe.getMessage());
804 }
805
806 return content.toString();
807 }
808
809 return content.toString();
810 }
811 }
812
813 /** Wrapper Control.
814  * @param request HttpServletRequest
815  * @return String
816  */
817 public String getWrapper(HttpServletRequest request) {
818     StringBuffer wrapper = new StringBuffer();
819     wrapper.append("<wrapper>\n");
820     wrapper.append(((Hashtable)context.getAttribute("wrappers")).get("shell"));
821     wrapper.append(((Hashtable)context.getAttribute("wrappers")).get("tools"));
822     //wrapper.append(getNav(request));
823     wrapper.append("</wrapper>\n");
824     return wrapper.toString();
825 }
826 }
827
828 /** Used to condition text data from SQL tables for display in XSL Stylesheet.
829  * @param strIn String In
830  * @return String
831  */
832 public String prepText(String strIn){
833     //DataValidator dv = new DataValidator();
834     String Instr = strIn;
835     String strOut = "";
836     if(strIn != null){
837         if(Instr.length() > 0 && !Instr.equalsIgnoreCase("null")){
838             //strOut = dv.validateText(Instr);
839             strOut = Instr;
840
841             try {
842
843                 //strOut = new RE("\\\\").subst(strOut, "\\");
844                 //strOut = new RE("'").subst(strOut, "");
845                 //strOut = new RE("&").subst(strOut, "&");
846                 strOut = new RE("&").subst(strOut, "%#38;");
847                 strOut = new RE(">").subst(strOut, "&gt;");
848                 strOut = new RE("<").subst(strOut, "&lt;");
849                 //strOut = new RE("\\'").subst(strOut, "&apos;");
850                 strOut = new RE("&").subst(strOut, "&quot;");
851                 strOut = new RE("/").subst(strOut, "%#47;"); //in addition to normal conditioning by Data
852                 strOut = new RE("&").subst(strOut, "&apos;");
853
854             }
855             catch(RESyntaxException rese) {
856                 System.out.println("Exception thrown in StudentInfo.prepText(): " + rese.getMessage());
857             }
858         }
859     }
860 }
861 }
862
863 return strOut;
864 }
865 }
866
867 /** Condition date data from SQL database for display in XSL Stylesheet.
868  * @param strIn String In
869  * @return String
870  */
871 public String prepDate(String strIn){
872     String Instr = strIn;
873     String strOut = "";
874     if(strIn != null){
875
876         strOut = strIn.substring(4,6) + "%#47;" + strIn.substring(6,8) + "%#47;" + strIn.substring(0,4);
877
878     }
879 }
880 //System.out.println("strIn = "+strIn);
881 //System.out.println("strOut = "+strOut);
882 return strOut;

```

```
883     }
884 }
885
886 /** Used to return the text from the lookup table for the given parameters.
887  * @param strType String (Type of Data)
888  * @param strCode String (Code to look up)
889  * @return String
890  */
891 public String getItemDetails(String strType, String strCode){
892     String strDetails = prepText(strCode); //-- if not found just return the original code...
893
894
895
896     String sql = "SELECT * FROM [STBL-EXTRACT] WHERE [TYPE]='"+strType+"' AND [CODE]='"+strCode+"'";
897
898
899     try {
900         Class.forName(dbSQL2000Driver);
901
902
903         Connection dbConSIS = DriverManager.getConnection(sql2000ServerDBsis,sql2000AdminIdSis,sql2000AdminPwSis);
904
905         Statement stSIS = dbConSIS.createStatement();
906         ResultSet rsSIS = stSIS.executeQuery(sql);
907
908         while(rsSIS.next()) {
909
910             strDetails = prepText(rsSIS.getString("DATA")).trim();
911
912         }
913
914
915
916         stSIS.close();
917         dbConSIS.close();
918         Class.forName(dbDriver);
919     }
920     catch(ClassNotFoundException cnfe) {
921         System.out.println("ClassNotFoundException thrown in StudentInfo.getItemDetails(): " + cnfe.getMessage());
922
923
924         return strDetails;
925     }
926
927     catch(SQLException sqle) {
928         System.out.println("SQLException thrown in StudentInfo.getItemDetails(): " + sqle.getMessage());
929
930         try{
931             Class.forName(dbDriver);// make sure driver reset to mySql
932         }
933         catch(ClassNotFoundException cnfe) {
934             System.out.println("ClassNotFoundException thrown in StudentInfo.getItemDetails(): " + cnfe.getMessage());
935         }
936
937         return strDetails;
938     }
939
940
941     return strDetails;
942 }
943
944 /** Returns the School Name for a given School_ID.
945  * @param strSchoolID String (School ID to look up)
946  * @return String
947  */
948 public String getSchoolName(String strSchoolID){
949     String strSchoolName = prepText(strSchoolID); //-- if not found just return the original code...
950
951
952
953     String sql = "SELECT * FROM [SCHOOL-EXTRACT] WHERE [SCHOOL_ID]='"+strSchoolID+"'";
954
955
956
957     try {
958         Class.forName(dbSQL2000Driver);
959
960
961         Connection dbConSIS = DriverManager.getConnection(sql2000ServerDBsis,sql2000AdminIdSis,sql2000AdminPwSis);
962
963         Statement stSIS = dbConSIS.createStatement();
964         ResultSet rsSIS = stSIS.executeQuery(sql);
965
966         while(rsSIS.next()) {
967
968             strSchoolName = prepText(rsSIS.getString("SCHOOL_NAME")).trim();
969
970         }
971
972
973
974         stSIS.close();
975         dbConSIS.close();
976         Class.forName(dbDriver);
977     }
978     catch(ClassNotFoundException cnfe) {
979         System.out.println("ClassNotFoundException thrown in StudentInfo.getSchoolName(): " + cnfe.getMessage());
980
981
982
983     }
```

```

981         return strSchoolName;
982     }
983 }
984 catch(SQLException sqle) {
985     System.out.println("SQLException thrown in StudentInfo.getSchoolName(): " + sqle.getMessage());
986 }
987 try{
988     Class.forName(dbDriver);// make sure driver reset to mySql
989 }
990 catch(ClassNotFoundException cnfe) {
991     System.out.println("ClassNotFoundException thrown in StudentInfo.getSchoolName(): " + cnfe.getMessage());
992 }
993     return strSchoolName;
994 }
995 }
996 }
997 }
998     return strSchoolName;
999 }
1000 }
1001 }
1002 /** Used to retrieve an XML list of Schools for XSL dropdown list that this teacher is allowed to see.
1003  * @param TeacherID String (Teacher_ID)
1004  * @return String
1005  */
1006 public String getTeacherSchoolList(String TeacherID){
1007     String strInTeacherID = TeacherID;
1008     int ischoolcount = 0;
1009     StringBuffer content = new StringBuffer();
1010 }
1011 int iListAll = 0;
1012 }
1013 //used to test for staff assigned to global access
1014 String sqlTeachTest = "SELECT * FROM [TEAC-EMPID] WHERE [EMP_ID]='"+strInTeacherID+"'";
1015 }
1016 }
1017 //used to display list of all schools
1018 String sqlall = "SELECT * FROM [SCHOOL-EXTRACT] ORDER BY [SCHOOL_NAME]";
1019 }
1020 //-- USE THIS SQL STRING FOR PRODUCTION
1021 String sql = "SELECT DISTINCT [TEAC-EMPID].[SCHOOL_ID], [SCHOOL-EXTRACT].[SCHOOL_NAME] FROM [TEAC-EMPID] INNER JOIN [SCHOOL-EXTRACT] ON [TEAC-EMPID].[SCHOOL_ID] = [SCHOOL-EXTRACT].[SCHOOL_ID] WHERE ((([TEAC-EMPID].[EMP_ID]='"+strInTeacherID+"'")) ORDER BY [SCHOOL_NAME]";
1022 }
1023 }
1024 }
1025 try {
1026     Class.forName(dbSQL2000Driver);
1027 }
1028 Connection dbConSIS = DriverManager.getConnection(sql2000ServerDBsis,sql2000AdminIdSis,sql2000AdminPwSis);
1029 }
1030 Statement stSIS = dbConSIS.createStatement();
1031 ResultSet rsSIS = stSIS.executeQuery(sqlTeachTest);
1032 }
1033 while(rsSIS.next()) { //TEST ALL SCHOOLS FOR THIS TEACHER AND SEE IF ONE IS = strListAllSchoolID
1034     if(strListAllSchoolID.equalsIgnoreCase(rsSIS.getString("SCHOOL_ID").trim())){
1035         iListAll = 1;
1036     }
1037 }
1038 }
1039 }
1040 }
1041 if(iListAll == 1){ // this individual can see all schools...
1042 }
1043 }
1044 ResultSet rsSIS2 = stSIS.executeQuery(sqlall);
1045 }
1046 }
1047 while(rsSIS2.next()) {
1048     content.append("<school value='"+rsSIS2.getString("SCHOOL_ID").trim()+"\ id='"+rsSIS2.getString("SCHOOL_ID").trim()+"\>" + prepText(rsSIS2.getString("SCHOOL_NAME").trim())+"</school>\n");
1049     ischoolcount++;
1050 }
1051 }
1052 }
1053 }
1054 }
1055 }
1056 }
1057 }
1058 }
1059 }
1060 }
1061 }
1062 }
1063 }
1064 }
1065 }
1066 }
1067 }
1068 }
1069 }
1070 }
1071 }
1072 }
1073 }
1074 }
1075 }
1076 }
1077 }

```



```
1078     }
1079 }
1080 catch(SQLException sqle) {
1081     System.out.println("SQLException thrown in StudentInfo.getTeacherSchoolList(): " + sqle.getMessage());
1082
1083     content.setLength(0);
1084     content.append("<school id=\"0000\" value=\"0000\">ERROR SQLException</school>\n");
1085     try{
1086         Class.forName(dbDriver);// make sure driver reset to mySql
1087     }
1088     catch(ClassNotFoundException cnfe) {
1089         System.out.println("ClassNotFoundException thrown in StudentInfo.getTeacherSchoolList(): " + cnfe.getMessage());
1090     }
1091     return content.toString();
1092 }
1093
1094 if(ischoolcount == 0){ //no schools available for this teacher
1095     content.setLength(0);
1096     content.append("<school id=\"0000\" value=\"0000\">NO SCHOOLS AVAILABLE</school>\n");
1097 }
1098
1099
1100 //System.out.println(content.toString());
1101 return content.toString();
1102 }
1103
1104 }
1105
1106 /** Style sheet used for display
1107  * @return string
1108  */
1109 public String getStylesheet() {
1110     return "/apps/tools/studentinfo-html.xml";
1111 }
1112
1113 /**
1114  * Returns a short description of the servlet.
1115  * @return string
1116  */
1117 public String getServletInfo() {
1118     return "Short description of StudentInfo";
1119 }
1120
1121 }
1122 }
1123
1124
1125
1126
```

```

1  <?xml version="1.0"?>
2
3  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4    <xsl:import href="../page-html.xml"/>
5
6  <!--
7    STUDENTINFO-HTML.XSL
8
9    This XSL is used with StudentInfo.java to display Student Information to Teachers and School District Employees.
10
11    Created 2/2002
12    Daniel J. Moore
13
14  -->
15
16
17  <xsl:template match="teachersis">
18    <META HTTP-EQUIV="Expires" CONTENT="0" />
19    <META HTTP-EQUIV="Pragma" CONTENT="no-cache" />
20    <xsl:choose>
21      <xsl:when test="logout=''">
22        </xsl:when>
23      <xsl:otherwise>
24        <META HTTP-EQUIV="refresh" content="5; url={logout}" />
25      </xsl:otherwise>
26    </xsl:choose>
27    <script language="Javascript" src="/lib/apps/tools/studentinfo.js"></script>
28    <table border="0" cellspacing="0" cellpadding="1" align="left" valign="top" width="100%" height="100%">
29      <tr><td width="100%" height="100%">
30        <table class="mainborder">
31          <tr>
32            <td align="center" width="100%" valign="top">
33              <table width="100%" align="center" valign="top" cellpadding="0" cellspacing="0">
34                <tr>
35                  <td align="left" width="30%">
36                    <table align="left" cellpadding="0" cellspacing="3">
37                      <tr>
38                        <td align="right" class="dialoginfohead">Teacher:</td>
39                        <td align="left" class="teacherinfo" nowrap="nowrap"><div id="teachername"><xsl:value-of select="teachername"/></div></td>
40                      </tr>
41                      <tr>
42                        <td align="right" class="dialoginfohead">Emp.&#160;ID:</td>
43                        <td align="left" class="teacherinfo" nowrap="nowrap"><div id="teacherempid"><xsl:value-of select="teacherid"/></div></td>
44                      </tr>
45                    </table>
46                  </td>
47                  <td align="center" class="title2">Teacher&#160;Access&#160;to&#160;Student&#160;Information</td>
48                <td valign="middle" align="center" class="dialoginfohead" nowrap="nowrap" width="30%"><xsl:value-of select="version"/><br/>
49                  <xsl:choose>
50                    <xsl:when test="logout=''">
51                      <input type="button" onclick="goLogout();return false;" class="logoffbutton" value="Log Off" />
52                    </xsl:when>
53                    <xsl:otherwise>
54                      <input type="button" onclick="return false;" class="logoffbutton" value=" " />
55                    </xsl:otherwise>
56                  </xsl:choose>
57                </td>
58              </tr>
59            </table>
60          </td>
61        </tr>
62      <tr>
63        <td align="center" valign="top">
64          <table width="100%" align="center" valign="top" cellpadding="4" cellspacing="0">
65            <tr>
66              <td align="left">
67                <table align="left" valign="middle" cellpadding="0" cellspacing="0" height="100%">
68                  <tr>
69                    <td align="left" valign="bottom" height="30%">
70                      <table align="left" valign="top" cellpadding="0" cellspacing="0" border="0">
71                        <tr>
72                          <td align="left" valign="top">
73                            <xsl:apply-templates select="currentstudent"/>
74                          </td>
75                        </tr>
76                      </table>
77                    </td>
78                    <td align="left" valign="bottom" height="70%"><xsl:apply-templates select="specialinfo"/></td>
79                  </tr>
80                </table>
81              </td>
82            </table>
83          </td>
84          <td align="right" height="100%">
85            <form name="StuSearch" id="StuSearch" action="StudentInfo" method="post">
86              <table border="0" cellspacing="0" cellpadding="0" align="right" valign="middle" height="100%" class="singleborder">
87                <tr>
88                  <td align="right" valign="top" height="100%">
89                    <table border="0" cellspacing="0" cellpadding="1">
90                      <tr>
91                        <td class="dialoginfohead" align="center" valign="bottom">&#160;</td>
92                        <td align="center" valign="bottom" class="dialoginfohead">School</td>
93                      </tr>
94                      <tr>
95                        <td class="dialoginfohead2" align="center" valign="top" style="color:#dd0000">SEARCH</td>
96                        <td align="center" valign="middle">
97                          <xsl:apply-templates select="teacherschoollist"/>
98                        </td>

```

```

99     </tr>
100    <tr>
101      <td class="dialoginfohead" align="center" valign="bottom">Home&#160;Room</td>
102      <td class="dialoginfohead" align="center" valign="bottom">Student&#160;Last&#160;Name</td>
103    </tr>
104    <tr>
105      <td align="center" valign="top"><input type="text" name="searchstuhomeroom" id="searchstuhomeroom" class="teacherinfo" size="7" maxlength="20">
106      <xsl:attribute name="value"><xsl:value-of select="searchhr"/></xsl:attribute></input>
107    </td>
108      <td align="center" valign="top"><input type="text" name="searchstulast" id="searchstulast" class="teacherinfo" size="18" maxlength="50">
109      <xsl:attribute name="value"><xsl:value-of select="searchlastname"/></xsl:attribute></input>
110    </td>
111  </tr>
112  <tr>
113    <td class="dialoginfohead" align="center" valign="bottom">Student&#160;ID</td>
114    <td class="dialoginfohead" align="center" valign="bottom">Student&#160;First&#160;Name</td>
115  </tr>
116  <tr>
117    <td align="center" valign="top"><input type="text" name="searchstuid" id="searchstuid" class="teacherinfo" size="7" maxlength="20">
118    <xsl:attribute name="value"><xsl:value-of select="searchid"/></xsl:attribute></input>
119  </td>
120    <td align="center" valign="top"><input type="text" name="searchstufirst" id="searchstufirst" class="teacherinfo" size="18" maxlength="50">
121    <xsl:attribute name="value"><xsl:value-of select="searchfirstname"/></xsl:attribute></input>
122  </td>
123 </tr>
124 </table>
125 </td>
126 <td class="dialoginfohead">&#160;</td>
127 <td align="right" valign="middle" height="100%">
128   <table cellspacing="0" cellpadding="1" valign="middle">
129     <tr>
130       <td align="center" valign="top">
131         <xsl:choose>
132           <xsl:when test="logout=''">
133             <input type="button" value="Find" onclick="displaySearchResults();return false;" class="buttonclass" style="width:80;"/>
134           </xsl:when>
135           <xsl:otherwise>
136             <input type="button" value=" " onclick="return false;" class="buttonclass" style="width:80;"/>
137           </xsl:otherwise>
138         </xsl:choose>
139       </td>
140     </tr>
141     <tr>
142       <td align="center" valign="middle"><input type="button" value="Clear" onclick="newSearch();return false;" class="buttonclass" style="width:80;"/></td>
143     </tr>
144     <tr>
145       <td align="center" valign="bottom"><input type="reset" class="buttonclass" style="width:80;" value="Reset"/></td>
146     </tr>
147   </table>
148 </td>
149 </tr>
150 </table>
151 <input type="hidden" name="studentfind" />
152 <input type="hidden" name="method" />
153 </form>
154 </td>
155 </tr>
156 </table>
157 </td>
158 </tr>
159 <tr>
160   <td align="center" valign="top" height="100%">
161     <table cellpadding="0" cellspacing="0" valign="top" align="center" height="100%" width="100%">
162       <tr>
163         <td valign="top">
164           <div id="tabbar"></div>
165         </td>
166       </tr>
167       <tr>
168         <td height="100%" width="100%" valign="top" align="center">
169           <table cellpadding="0" cellspacing="0" bgcolor="#ffffff" valign="top" align="left" height="100%" width="100%">
170             <tr>
171               <td width="100%" height="100%" valign="top" align="left">
172                 <div id="tabcontent"></div>
173               </td>
174             </tr>
175           </table>
176         </td>
177       </tr>
178     </table>
179   </td>
180 </tr>
181 </table>
182
183 </td></tr>
184 </table>
185
186 <!-- ***** Student Data ***** -->
187
188 <div id="stuinfo" style="visibility:hidden">
189 <xsl:apply-templates select="studentinfo"/>
190 </div>
191
192 <!-- ***** PARENT INFO ***** -->
193
194 <div id="parinfo" style="visibility:hidden">
195 <xsl:apply-templates select="parentinfo"/>
196

```

```

197 </div>
198
199 <!-- ***** EMERGENCY INFO ***** -->
200
201 <div id="emerginfo" style="visibility:hidden">
202 <xsl:apply-templates select="emergencyinfo"/>
203 </div>
204
205 <!-- ***** DAYCARE ***** -->
206
207 <div id="daycareinfo" style="visibility:hidden">
208 <xsl:apply-templates select="daycareinfo"/>
209 </div>
210
211 <!-- search results (if more than one student) -->
212
213 <div id="studentsfoundlist" style="visibility:hidden">
214 <xsl:apply-templates select="searchresults"/>
215 </div>
216
217 <!-- div id="thisstudent" style="visibility:hidden">
218 </div>
219
220
221 <div id="healthinfo" style="visibility:hidden">
222 </div>
223
224 <div id="confidentialinfo" style="visibility:hidden">
225 </div>
226
227 <div id="nocontactinfo" style="visibility:hidden">
228 </div -->
229
230 <div id="blankfill" style="visibility:hidden">
231 <table border="0" cellspacing="0" cellpadding="2" valign="bottom" align="left">
232 <tr>
233 <td >#160;</td>
234 </tr>
235 <tr>
236 <td class="title2" align="center"><xsl:value-of select="statusmsg"/></td>
237 </tr>
238 <tr>
239 <td class="title2" align="center" style="color:#dd0000;"><xsl:value-of select="errormsg"/></td>
240 </tr>
241 </table>
242 </div>
243
244
245
246 <script>
247 <xsl:comment>
248
249 document.StuSearch.schoollist.selectedIndex = getSelectedIndex( document.StuSearch.schoollist, '<xsl:apply-templates select="currentschool"/>');
250 var arrTabLabel = new Array(<xsl:apply-templates select="arrTabLabelcontent"/>);
251
252 var arrDisplayDiv = new Array(<xsl:apply-templates select="arrDisplayDivcontent"/>);
253
254 init();
255
256 // </xsl:comment>
257 </script>
258
259
260 </xsl:template>
261
262
263
264
265 <xsl:template match="specialinfo">
266 <table border="0" cellspacing="0" cellpadding="2" align="left">
267 <tr>
268 <td align="left" valign="middle"><xsl:apply-templates select="confidential"/></td>
269 <td align="left" valign="middle"><xsl:apply-templates select="nocontact"/></td>
270 <td align="left" valign="middle"><xsl:apply-templates select="health"/></td>
271 </tr>
272 </table>
273 </xsl:template>
274
275 <xsl:template match="nocontact">
276 <table border="0" cellspacing="0" cellpadding="2" bgcolor="#dd0000" valign="bottom" align="left" height="100%">
277 <tr>
278 <td class="ncwarningfill" align="center" style="font-size:12px;">!!!#160;NO#160;CONTACT#160;!!!</td>
279 </tr>
280 <tr>
281 <td class="ncwarningfill" align="center"><xsl:value-of select="."/></td>
282 </tr>
283 </table>
284 </xsl:template>
285
286 <xsl:template match="health">
287 <table border="0" cellspacing="0" cellpadding="2" class="healthbox" valign="bottom" height="100%">
288 <tr>
289 <td align="center" style="font-size:12px;">+##160;##160;HEALTH##160;CONCERNS##160;##160;+</td>
290 </tr>
291 <tr>
292 <td align="center"><xsl:value-of select="."/></td>
293 </tr>
294 </table>

```

```
295 </xsl:template>
296
297 <xsl:template match="confidential">
298 <table border="0" cellspacing="0" cellpadding="2" class="caution" height="100%">
299 <tr>
300 <td align="center" valign="middle" height="100%" style="font-size:12px;">CONTACT OFFICE FOR ADDITIONAL INFORMATION<BR/>REGARDING THIS STUDENT</td>
301 </tr>
302 </table>
303 </xsl:template>
304
305 <xsl:template match="teacherschoollist">
306 <select id="schoollist" name="schoollist" onclick="return false;" class="teacherinfo" style="height:9px;width:120px;">
307 <xsl:apply-templates select="school"/>
308 </select>
309 </xsl:template>
310
311 <xsl:template match="school">
312 <option value="{@id}" id="{@id}">
313 <xsl:value-of select="."/>
314 </option>
315 </xsl:template>
316
317 <xsl:template match="student">
318 <tr>
319 <td class="teacherinfo" align="center">
320 <xsl:choose>
321 <xsl:when test="id=''>
322 &#160;
323 </xsl:when>
324 <xsl:otherwise>
325 <a href="StudentInfo">
326 <xsl:attribute name="onClick">getThisStudent('{xsl:value-of select="id"/}')return false;</xsl:attribute><xsl:value-of select="id"/>
327 </a>
328 </xsl:otherwise>
329 </xsl:choose>
330 </td>
331 <td class="teacherinfo" align="center">
332 <xsl:choose>
333 <xsl:when test="last=''>
334 &#160;
335 </xsl:when>
336 <xsl:otherwise>
337 <xsl:value-of select="last"/>
338 </xsl:otherwise>
339 </xsl:choose>
340 </td>
341 <td class="teacherinfo" align="center">
342 <xsl:choose>
343 <xsl:when test="first=''>
344 &#160;
345 </xsl:when>
346 <xsl:otherwise>
347 <xsl:value-of select="first"/>
348 </xsl:otherwise>
349 </xsl:choose>
350 </td>
351 <td class="teacherinfo" align="center">
352 <xsl:choose>
353 <xsl:when test="middle=''>
354 &#160;
355 </xsl:when>
356 <xsl:otherwise>
357 <xsl:value-of select="middle"/>
358 </xsl:otherwise>
359 </xsl:choose>
360 </td>
361 <td class="teacherinfo" align="center">
362 <xsl:choose>
363 <xsl:when test="birth=''>
364 &#160;
365 </xsl:when>
366 <xsl:otherwise>
367 <xsl:value-of select="birth"/>
368 </xsl:otherwise>
369 </xsl:choose>
370 </td>
371 <td class="teacherinfo" align="center">
372 <xsl:choose>
373 <xsl:when test="gender=''>
374 &#160;
375 </xsl:when>
376 <xsl:otherwise>
377 <xsl:value-of select="gender"/>
378 </xsl:otherwise>
379 </xsl:choose>
380 </td>
381 <td class="teacherinfo" align="center">
382 <xsl:choose>
383 <xsl:when test="hr=''>
384 &#160;
385 </xsl:when>
386 <xsl:otherwise>
387 <xsl:value-of select="hr"/>
388 </xsl:otherwise>
389 </xsl:choose>
390 </td>
391 <td class="teacherinfo" align="center">
392 <xsl:choose>
```

```

393     <xsl:when test="grade=''>
394         &#160;
395     </xsl:when>
396     <xsl:otherwise>
397         <xsl:value-of select="grade"/>
398     </xsl:otherwise>
399     </xsl:choose>
400 </td>
401 <td class="teacherinfo" align="center">
402     <xsl:choose>
403         <xsl:when test="curschool=''>
404             &#160;
405         </xsl:when>
406         <xsl:otherwise>
407             <xsl:choose>
408                 <xsl:when test="curschool/@r=''>
409                     <xsl:value-of select="curschool"/>
410                 </xsl:when>
411                 <xsl:otherwise>
412                     <font color="#dd0000"><xsl:value-of select="curschool"/></font>
413                 </xsl:otherwise>
414             </xsl:choose>
415         </xsl:otherwise>
416     </xsl:choose>
417 </td>
418 <td class="teacherinfo" align="center">
419     <xsl:choose>
420         <xsl:when test="status=''>
421             &#160;
422         </xsl:when>
423         <xsl:otherwise>
424             <xsl:choose>
425                 <xsl:when test="status/@r=''>
426                     <xsl:value-of select="status"/>
427                 </xsl:when>
428                 <xsl:otherwise>
429                     <font color="#dd0000"><xsl:value-of select="status"/></font>
430                 </xsl:otherwise>
431             </xsl:choose>
432         </xsl:otherwise>
433     </xsl:choose>
434 </td>
435 </tr>
436 </xsl:template>
437
438 <xsl:template match="searchresults">
439     <font class="dialoginfohead">&#160;</font>
440     <table border="1" cellspacing="0" cellpadding="5" align="left" valign="top">
441         <tr>
442             <td colspan="10" class="dialoginfohead" align="center"><font color="#dd0000"><i>(Red Indicates Student is Enrolled in Another School)</i></font></td>
443         </tr>
444         <tr>
445             <td class="dialoginfohead" align="center">Student&#160;ID</td>
446             <td class="dialoginfohead" align="center">Last&#160;Name</td>
447             <td class="dialoginfohead" align="center">First&#160;Name</td>
448             <td class="dialoginfohead" align="center">Middle&#160;Name</td>
449             <td class="dialoginfohead" align="center">Birth&#160;Date</td>
450             <td class="dialoginfohead" align="center">Gender</td>
451             <td class="dialoginfohead" align="center">Home&#160;Room</td>
452             <td class="dialoginfohead" align="center">Grade</td>
453             <td class="dialoginfohead" align="center">Current&#160;School</td>
454             <td class="dialoginfohead" align="center">Status</td>
455         </tr>
456     <xsl:apply-templates/>
457 </table>
458 </xsl:template>
459
460 <xsl:template match="recordcount">
461     <font class="dialoginfohead">The Number of Records Returned = <xsl:value-of select="."/></font>
462 </xsl:template>
463
464 <xsl:template match="currentstudent">
465     <table border="0" cellspacing="2" cellpadding="2" align="left" valign="top">
466         <tr>
467             <td class="dialoginfohead" align="center" valign="top">Student&#160;id</td>
468             <td class="dialoginfohead" align="center" valign="top">Name</td>
469             <td class="dialoginfohead" align="center" valign="top">Home&#160;Room</td>
470         </tr>
471         <tr>
472             <td class="datacell_wt" align="center" valign="bottom">
473                 <xsl:choose>
474                     <xsl:when test="studentid=''>
475                         &#160;
476                     </xsl:when>
477                     <xsl:otherwise>
478                         <xsl:value-of select="studentid"/>
479                     </xsl:otherwise>
480                 </xsl:choose>
481             </td>
482             <td class="datacell_wt" align="center" valign="bottom">
483                 <xsl:choose>
484                     <xsl:when test="studentname=''>
485                         &#160;
486                     </xsl:when>
487                     <xsl:otherwise>
488                         <xsl:value-of select="studentname"/>
489                     </xsl:otherwise>
490                 </xsl:choose>

```

```

491     </td>
492     <td class="datacell_wt" align="center" valign="bottom">
493     <xsl:choose>
494     <xsl:when test="studentthr=''>
495     &#160;
496     </xsl:when>
497     <xsl:otherwise>
498     <xsl:value-of select="studentthr"/>
499     </xsl:otherwise>
500     </xsl:choose>
501 </td>
502 </tr>
503 <tr>
504 <td colspan="3" class="teacherinfo" style="font-size:9px;" align="left" valign="top">Records&#160;Last&#160;Modified&#160;on&#160;<b>
505 <font color="#880000">
506 <xsl:choose>
507 <xsl:when test="moddate=''>
508 &#160;
509 </xsl:when>
510 <xsl:otherwise>
511 <xsl:value-of select="moddate"/>
512 </xsl:otherwise>
513 </xsl:choose>
514 </font></b>
515 </td>
516 </tr>
517 </table>
518 </xsl:template>
519
520 <xsl:template match="daycareinfo">
521 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
522 <tr>
523 <td>
524 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
525 <tr>
526 <td class="dialoginfohead" align="center">Daycare Center</td>
527 <td class="dialoginfohead" align="center">Contact</td>
528 <td class="dialoginfohead" align="center">Title</td>
529 <td class="dialoginfohead" align="center">Phone</td>
530 <td class="dialoginfohead" align="center">Ext</td>
531 </tr>
532 <tr>
533 <td class="datacell" align="center">
534 <xsl:choose>
535 <xsl:when test="center=''>
536 &#160;
537 </xsl:when>
538 <xsl:otherwise>
539 <xsl:value-of select="center"/>
540 </xsl:otherwise>
541 </xsl:choose>
542 </td>
543 <td class="datacell" align="center">
544 <xsl:choose>
545 <xsl:when test="contact=''>
546 &#160;
547 </xsl:when>
548 <xsl:otherwise>
549 <xsl:value-of select="contact"/>
550 </xsl:otherwise>
551 </xsl:choose>
552 </td>
553 <td class="datacell" align="center">
554 <xsl:choose>
555 <xsl:when test="title=''>
556 &#160;
557 </xsl:when>
558 <xsl:otherwise>
559 <xsl:value-of select="title"/>
560 </xsl:otherwise>
561 </xsl:choose>
562 </td>
563 <td class="datacell" align="center"><xsl:apply-templates select="onenumber"/></td>
564 <td class="datacell" align="center">
565 <xsl:choose>
566 <xsl:when test="ext=''>
567 &#160;
568 </xsl:when>
569 <xsl:otherwise>
570 <xsl:value-of select="ext"/>
571 </xsl:otherwise>
572 </xsl:choose>
573 </td>
574 </tr>
575 </table>
576 </td>
577 </tr>
578 <tr>
579 <td>
580 <table border="0" cellspacing="0" cellpadding="2" valign="top" align="left">
581 <tr>
582 <td class="dialoginfohead" valign="bottom" align="center">Address</td>
583 </tr>
584 <tr>
585 <xsl:apply-templates select="address"/>
586 </tr>
587 </table>
588 </td>

```

```

589 </tr>
590 </table>
591 </xsl:template>
592
593
594 <xsl:template match="emergencyinfo">
595 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
596 <tr>
597 <td>
598 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
599 <tr>
600 <td class="dialoginfohead" align="center">Emergency&#160;Contact&#160;1</td>
601 <td class="dialoginfohead" align="center">Phone</td>
602 <td class="dialoginfohead" align="center">Relationship</td>
603 </tr>
604 <tr>
605 <td class="datacell" align="center">
606 <xsl:choose>
607 <xsl:when test="contactname1=''>
608 &#160;
609 </xsl:when>
610 <xsl:otherwise>
611 <xsl:value-of select="contactname1"/>
612 </xsl:otherwise>
613 </xsl:choose>
614 </td>
615 <td class="datacell" align="center"><xsl:apply-templates select="contactphone1"/></td>
616 <td class="datacell" align="center">
617 <xsl:choose>
618 <xsl:when test="relationship1=''>
619 &#160;
620 </xsl:when>
621 <xsl:otherwise>
622 <xsl:value-of select="relationship1"/>
623 </xsl:otherwise>
624 </xsl:choose>
625 </td>
626 </tr>
627 </table>
628 </td>
629 </tr>
630 <tr>
631 <td>
632 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
633 <tr>
634 <td class="dialoginfohead" align="center">Emergency&#160;Contact&#160;2</td>
635 <td class="dialoginfohead" align="center">Phone</td>
636 </tr>
637 <tr>
638 <td class="datacell" align="center">
639 <xsl:choose>
640 <xsl:when test="contactname2=''>
641 &#160;
642 </xsl:when>
643 <xsl:otherwise>
644 <xsl:value-of select="contactname2"/>
645 </xsl:otherwise>
646 </xsl:choose>
647 </td>
648 <td class="datacell" align="center"><xsl:apply-templates select="contactphone2"/></td>
649 </tr>
650 </table>
651 </td>
652 </tr>
653 <tr>
654 <td>
655 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
656 <tr>
657 <td class="dialoginfohead" align="center">Emergency&#160;Contact&#160;3</td>
658 <td class="dialoginfohead" align="center">Phone</td>
659 </tr>
660 <tr>
661 <td class="datacell" align="center">
662 <xsl:choose>
663 <xsl:when test="contactname3=''>
664 &#160;
665 </xsl:when>
666 <xsl:otherwise>
667 <xsl:value-of select="contactname3"/>
668 </xsl:otherwise>
669 </xsl:choose>
670 </td>
671 <td class="datacell" align="center"><xsl:apply-templates select="contactphone3"/></td>
672 </tr>
673 </table>
674 </td>
675 </tr>
676 <tr>
677 <td>
678 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
679 <tr>
680 <td class="dialoginfohead" align="center">Emergency&#160;Contact&#160;4</td>
681 <td class="dialoginfohead" align="center">Phone</td>
682 </tr>
683 <tr>
684 <td class="datacell" align="center">
685 <xsl:choose>
686 <xsl:when test="contactname4=''>

```



```

687         {$#160;
688         </xsl:when>
689         <xsl:otherwise>
690         <xsl:value-of select="contactname4"/>
691         </xsl:otherwise>
692         </xsl:choose>
693     </td>
694     <td class="datacell" align="center"><xsl:apply-templates select="contactphone4"/></td>
695 </tr>
696 </table>
697 </td>
698 </tr>
699 </table>
700 </xsl:template>
701
702 <xsl:template match="studentinfo">
703 <table border="0" cellpadding="0" cellspacing="0" valign="top" align="left">
704 <tr>
705 <td valign="top" align="left">
706 <table border="0" cellspacing="0" cellpadding="2">
707 <tr>
708 <td>
709 <table border="0" cellspacing="0" cellpadding="2">
710 <tr>
711 <td class="dialoginfohead" align="center">Grade</td>
712 <td class="dialoginfohead" align="center">Birthdate</td>
713 <td class="dialoginfohead" align="center">Gender</td>
714 <td class="dialoginfohead" align="center">Ethnicity</td>
715 <td class="dialoginfohead" align="center">Option</td>
716 <td class="dialoginfohead" align="center">Grad Year</td>
717 <td class="dialoginfohead" align="center"><xsl:value-of select="displaydate/@type"/></td>
718 <td class="dialoginfohead" align="center">Status</td>
719 <td class="dialoginfohead" align="center">Current School</td>
720 <td class="dialoginfohead" align="center">Previous School</td>
721 </tr>
722 <tr>
723 <td class="datacell" align="center">
724 <xsl:choose>
725 <xsl:when test="grade=''>
726 {$#160;
727 </xsl:when>
728 <xsl:otherwise>
729 <xsl:value-of select="grade"/>
730 </xsl:otherwise>
731 </xsl:choose>
732 </td>
733 <td class="datacell" align="center">
734 <xsl:choose>
735 <xsl:when test="birth=''>
736 {$#160;
737 </xsl:when>
738 <xsl:otherwise>
739 <xsl:value-of select="birth"/>
740 </xsl:otherwise>
741 </xsl:choose>
742 </td>
743 <td class="datacell" align="center">
744 <xsl:choose>
745 <xsl:when test="gender=''>
746 {$#160;
747 </xsl:when>
748 <xsl:otherwise>
749 <xsl:value-of select="gender"/>
750 </xsl:otherwise>
751 </xsl:choose>
752 </td>
753 <td class="datacell" align="center">
754 <xsl:choose>
755 <xsl:when test="ethnicity=''>
756 {$#160;
757 </xsl:when>
758 <xsl:otherwise>
759 <xsl:value-of select="ethnicity"/>
760 </xsl:otherwise>
761 </xsl:choose>
762 </td>
763 <td class="datacell" align="center">
764 <xsl:choose>
765 <xsl:when test="option=''>
766 {$#160;
767 </xsl:when>
768 <xsl:otherwise>
769 <xsl:value-of select="option"/>
770 </xsl:otherwise>
771 </xsl:choose>
772 </td>
773 <td class="datacell" align="center">
774 <xsl:choose>
775 <xsl:when test="gradyear=''>
776 {$#160;
777 </xsl:when>
778 <xsl:otherwise>
779 <xsl:value-of select="gradyear"/>
780 </xsl:otherwise>
781 </xsl:choose>
782 </td>
783 <td class="datacell" align="center">
784 <xsl:choose>

```

```

785         <xsl:when test="displaydate=''>
786             #160;
787         </xsl:when>
788         <xsl:otherwise>
789             <xsl:value-of select="displaydate"/>
790         </xsl:otherwise>
791     </xsl:choose>
792 </td>
793 <td class="datacell" align="center">
794     <xsl:choose>
795         <xsl:when test="status=''>
796             #160;
797         </xsl:when>
798         <xsl:otherwise>
799             <xsl:value-of select="status"/>
800         </xsl:otherwise>
801     </xsl:choose>
802 </td>
803 <td class="datacell" align="center">
804     <xsl:choose>
805         <xsl:when test="curschool=''>
806             #160;
807         </xsl:when>
808         <xsl:otherwise>
809             <xsl:value-of select="curschool"/>
810         </xsl:otherwise>
811     </xsl:choose>
812 </td>
813 <td class="datacell" align="center">
814     <xsl:choose>
815         <xsl:when test="prevschool=''>
816             #160;
817         </xsl:when>
818         <xsl:otherwise>
819             <xsl:value-of select="prevschool"/>
820         </xsl:otherwise>
821     </xsl:choose>
822 </td>
823 </tr>
824 </table>
825 </td>
826 </tr>
827 </table>
828 </td>
829 </tr>
830 <tr>
831     <td valign="top" align="left">
832     <table border="0" cellspacing="0" cellpadding="2" valign="top" align="left">
833         <tr>
834             <td valign="top" align="left">
835                 <table border="0" cellspacing="0" cellpadding="2" valign="top" align="left">
836                     <tr>
837                         <td class="dialoginfohead" valign="bottom" align="center">Homes#160;Address</td>
838                     </tr>
839                     <tr>
840                         <xsl:apply-templates select="pladdress"/>
841                     </tr>
842                 </table>
843             </td>
844             <td valign="top" align="left">
845                 <table border="0" cellspacing="0" cellpadding="2" align="left">
846                     <tr>
847                         <td class="dialoginfohead" valign="bottom" align="center">Mailing#160;Address</td>
848                     </tr>
849                     <tr>
850                         <xsl:apply-templates select="pmailaddress"/>
851                     </tr>
852                 </table>
853             </td>
854             <td align="left" valign="top">
855                 <table border="0" cellspacing="0" cellpadding="2" valign="top" align="left">
856                     <tr>
857                         <td class="dialoginfohead" align="center">Student Phone</td>
858                         <td class="dialoginfohead" align="center">Living With</td>
859                         <td class="dialoginfohead" align="center">Joint Custody</td>
860                     <td></td>
861                     <td></td>
862                 </tr>
863                 <tr>
864                     <td class="datacell" align="center">
865                         <xsl:apply-templates select="stuphone"/>
866                     </td>
867                     <td class="datacell" align="center">
868                         <xsl:choose>
869                             <xsl:when test="livingwith=''>
870                                 #160;
871                             </xsl:when>
872                             <xsl:otherwise>
873                                 <xsl:value-of select="livingwith"/>
874                             </xsl:otherwise>
875                         </xsl:choose>
876                     </td>
877                     <td class="datacell" align="center">
878                         <xsl:choose>
879                             <xsl:when test="joint=''>
880                                 #160;
881                             </xsl:when>
882                             <xsl:otherwise>

```

```

883         <xsl:value-of select="joint"/>
884     </xsl:otherwise>
885 </xsl:choose>
886 </td>
887 <td></td>
888 <td></td>
889 </tr>
890 <tr>
891     <td class="dialoginfohead" align="center">Privacy</td>
892     <td class="dialoginfohead" align="center">504</td>
893     <td class="dialoginfohead" align="center">Homeless</td>
894     <td class="dialoginfohead" align="center">Case&#160;Worker</td>
895     <td class="dialoginfohead" align="center">Case&#160;Worker&#160;Phone</td>
896 </tr>
897 <tr>
898     <xsl:choose>
899         <xsl:when test="privacy=''>
900             <td class="datacell" align="center">&#160;</td>
901         </xsl:when>
902         <xsl:when test="privacy='Y'>
903             <td class="warningfill" align="center"><xsl:value-of select="privacy"/></td>
904         </xsl:when>
905         <xsl:when test="privacy='Y'>
906             <td class="warningfill" align="center"><xsl:value-of select="privacy"/></td>
907         </xsl:when>
908         <xsl:otherwise>
909             <td class="datacell" align="center"><xsl:value-of select="privacy"/></td>
910         </xsl:otherwise>
911     </xsl:choose>
912     <td class="datacell" align="center">
913         <xsl:choose>
914             <xsl:when test="five04=''>
915                 &#160;
916             </xsl:when>
917             <xsl:otherwise>
918                 <xsl:value-of select="five04"/>
919             </xsl:otherwise>
920         </xsl:choose>
921     </td>
922     <td class="datacell" align="center">
923         <xsl:choose>
924             <xsl:when test="homeless=''>
925                 &#160;
926             </xsl:when>
927             <xsl:otherwise>
928                 <xsl:value-of select="homeless"/>
929             </xsl:otherwise>
930         </xsl:choose>
931     </td>
932     <td class="datacell" align="center">
933         <xsl:choose>
934             <xsl:when test="caseworker=''>
935                 &#160;
936             </xsl:when>
937             <xsl:otherwise>
938                 <xsl:value-of select="caseworker"/>
939             </xsl:otherwise>
940         </xsl:choose>
941     </td>
942     <td class="datacell" align="center">
943         <xsl:apply-templates select="cwphone"/>
944     </td>
945 </tr>
946 </table>
947 </td>
948 </tr>
949 </table>
950 </td>
951 </tr>
952 <tr>
953     <td valign="top" align="left">
954         <table border="0" cellspacing="0" cellpadding="0">
955             <tr>
956                 <td class="datacell">
957                     <table border="0" cellspacing="0" cellpadding="2">
958                         <tr>
959                             <td>
960                                 <table border="0" cellspacing="0" cellpadding="2">
961                                     <tr>
962                                         <td colspan="2" class="dialoginfohead2" align="center">BILINGUAL INFO</td>
963                                     </tr>
964                                     <tr>
965                                         <td class="dialoginfohead" align="right">Bilg Elig</td>
966                                         <td class="datacell2" align="center">
967                                             <xsl:choose>
968                                                 <xsl:when test="bielig=''>
969                                                     &#160;
970                                                 </xsl:when>
971                                                 <xsl:otherwise>
972                                                     <xsl:value-of select="bielig"/>
973                                                 </xsl:otherwise>
974                                             </xsl:choose>
975                                         </td>
976                                     </tr>
977                                     <tr>
978                                         <td class="dialoginfohead" align="right">Country</td>
979                                         <td class="datacell2" align="center">
980                                             <xsl:choose>

```

```

981         <xsl:when test="country=''">
982             &#160;
983         </xsl:when>
984         <xsl:otherwise>
985             <xsl:value-of select="country"/>
986         </xsl:otherwise>
987     </xsl:choose>
988 </td>
989 </tr>
990 <tr>
991     <td class="dialoginfohead" align="right">Primary</td>
992     <td class="datacell2" align="center">
993         <xsl:choose>
994             <xsl:when test="pri=''">
995                 &#160;
996             </xsl:when>
997             <xsl:otherwise>
998                 <xsl:value-of select="pri"/>
999             </xsl:otherwise>
1000         </xsl:choose>
1001     </td>
1002 </tr>
1003 <tr>
1004     <td class="dialoginfohead" align="right">Home</td>
1005     <td class="datacell2" align="center">
1006         <xsl:choose>
1007             <xsl:when test="home=''">
1008                 &#160;
1009             </xsl:when>
1010             <xsl:otherwise>
1011                 <xsl:value-of select="home"/>
1012             </xsl:otherwise>
1013         </xsl:choose>
1014     </td>
1015 </tr>
1016 </table>
1017 </td>
1018 </tr>
1019 </table>
1020 </td>
1021 <td class="datacell">
1022     <table border="0" cellspacing="0" cellpadding="2">
1023         <tr>
1024             <td>
1025                 <table border="0" cellspacing="0" cellpadding="2">
1026                     <tr>
1027                         <td colspan="2" class="dialoginfohead2" align="center">SPECIAL ED INFO</td>
1028                     </tr>
1029                     <tr>
1030                         <td class="dialoginfohead" align="right">Level</td>
1031                         <td class="datacell2" align="center">
1032                             <xsl:choose>
1033                                 <xsl:when test="level=''">
1034                                     &#160;
1035                                 </xsl:when>
1036                                 <xsl:otherwise>
1037                                     <xsl:value-of select="level"/>
1038                                 </xsl:otherwise>
1039                             </xsl:choose>
1040                         </td>
1041                     </tr>
1042                     <tr>
1043                         <td class="dialoginfohead" align="right">Disability</td>
1044                         <td class="datacell2" align="left">
1045                             <xsl:apply-templates select="disabilities"/>
1046                         </td>
1047                     </tr>
1048                     <tr>
1049                         <td class="dialoginfohead" align="right">Type Svc</td>
1050                         <td class="datacell2" align="center">
1051                             <xsl:choose>
1052                                 <xsl:when test="svctype=''">
1053                                     &#160;
1054                                 </xsl:when>
1055                                 <xsl:otherwise>
1056                                     <xsl:value-of select="svctype"/>
1057                                 </xsl:otherwise>
1058                             </xsl:choose>
1059                         </td>
1060                     </tr>
1061                     <tr>
1062                         <td class="dialoginfohead" align="right">Unit</td>
1063                         <td class="datacell2" align="center">
1064                             <xsl:choose>
1065                                 <xsl:when test="unit=''">
1066                                     &#160;
1067                                 </xsl:when>
1068                                 <xsl:otherwise>
1069                                     <xsl:value-of select="unit"/>
1070                                 </xsl:otherwise>
1071                             </xsl:choose>
1072                         </td>
1073                     </tr>
1074                 </table>
1075             </td>
1076         </tr>
1077     </table>
1078 </td>

```

```

1079     </tr>
1080 </table>
1081 </td>
1082 </tr>
1083 </table>
1084 </xsl:template>
1085
1086 <xsl:template match="parentinfo">
1087 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
1088 <tr>
1089 <td colspan="3" align="left" valign="top">
1090 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
1091 <tr>
1092 <td class="dialoginfohead2" align="center">Parent/Guardian&#160;1</td>
1093 <td class="dialoginfohead" align="center">Relationship</td>
1094 <td class="dialoginfohead" align="center">Home&#160;Phone</td>
1095 <td class="dialoginfohead" align="center">Unlisted</td>
1096 <td class="dialoginfohead" align="center">Email</td>
1097 <td class="dialoginfohead" align="center">Employer</td>
1098 <td class="dialoginfohead" align="center">Work&#160;Phone</td>
1099 <td class="dialoginfohead" align="center">Ext.</td>
1100 <td class="dialoginfohead" align="center">Pager</td>
1101 </tr>
1102 <tr>
1103 <td class="datacell" align="center">
1104 <xsl:choose>
1105 <xsl:when test="pname=''">
1106 &#160;
1107 </xsl:when>
1108 <xsl:otherwise>
1109 <xsl:value-of select="pname"/>
1110 </xsl:otherwise>
1111 </xsl:choose>
1112 </td>
1113 <td class="datacell" align="center">
1114 <xsl:choose>
1115 <xsl:when test="prelation=''">
1116 &#160;
1117 </xsl:when>
1118 <xsl:otherwise>
1119 <xsl:value-of select="prelation"/>
1120 </xsl:otherwise>
1121 </xsl:choose>
1122 </td>
1123 <td class="datacell" align="center" nowrap="nowrap">
1124 <xsl:apply-templates select="piphone"/>
1125 </td>
1126 <xsl:choose>
1127 <xsl:when test="punlisted=''">
1128 <td class="datacell" align="center">&#160;</td>
1129 </xsl:when>
1130 <xsl:when test="punlisted='u'">
1131 <td class="warningfill" align="center">UNLISTED</td>
1132 </xsl:when>
1133 <xsl:when test="punlisted='U'">
1134 <td class="warningfill" align="center">UNLISTED</td>
1135 </xsl:when>
1136 <xsl:otherwise>
1137 <td class="datacell" align="center"><xsl:value-of select="punlisted"/></td>
1138 </xsl:otherwise>
1139 </xsl:choose>
1140 <td class="datacell" align="center">
1141 <xsl:choose>
1142 <xsl:when test="plemail=''">
1143 &#160;
1144 </xsl:when>
1145 <xsl:otherwise>
1146 <a><xsl:attribute name="href">mailto:<xsl:value-of select="plemail"/></xsl:attribute><xsl:value-of select="plemail"/></a>
1147 </xsl:otherwise>
1148 </xsl:choose>
1149 </td>
1150 <td class="datacell" align="center">
1151 <xsl:choose>
1152 <xsl:when test="plemployer=''">
1153 &#160;
1154 </xsl:when>
1155 <xsl:otherwise>
1156 <xsl:value-of select="plemployer"/>
1157 </xsl:otherwise>
1158 </xsl:choose>
1159 </td>
1160 <td class="datacell" align="center" nowrap="nowrap">
1161 <xsl:apply-templates select="pworkphone"/>
1162 </td>
1163 <td class="datacell" align="center">
1164 <xsl:choose>
1165 <xsl:when test="plext=''">
1166 &#160;
1167 </xsl:when>
1168 <xsl:otherwise>
1169 <xsl:value-of select="plext"/>
1170 </xsl:otherwise>
1171 </xsl:choose>
1172 </td>
1173 <td class="datacell" align="center" nowrap="nowrap">
1174 <xsl:apply-templates select="pipager"/>
1175 </td>
1176 </tr>

```

```
1177 <tr>
1178 <td>&#160;</td>
1179 <td align="left" valign="top" colspan="8">
1180 <table cellspacing="0" cellpadding="1">
1181 <tr>
1182 <td valign="top" align="left">
1183 <table border="0" cellspacing="0" cellpadding="2" valign="top" align="left">
1184 <tr>
1185 <td class="dialoginfohead" valign="bottom" align="center">Home&#160;Address</td>
1186 </tr>
1187 <tr>
1188 <xsl:apply-templates select="pladdress"/>
1189 </tr>
1190 </table>
1191 </td>
1192 <td valign="top" align="left">
1193 <table border="0" cellspacing="0" cellpadding="2" align="left">
1194 <tr>
1195 <td class="dialoginfohead" valign="bottom" align="center">Mailing&#160;Address</td>
1196 </tr>
1197 <tr>
1198 <xsl:apply-templates select="plmailaddress"/>
1199 </tr>
1200 </table>
1201 </td>
1202 </tr>
1203 </table>
1204 </td>
1205 </tr>
1206 </table>
1207 </td>
1208 <tr>
1209 <td colspan="3" align="left" valign="top">
1210 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
1211 <tr>
1212 <td class="dialoginfohead2" align="center">Parent/Guardian&#160;2</td>
1213 <td class="dialoginfohead" align="center">Relationship</td>
1214 <td class="dialoginfohead" align="center">Home&#160;Phone</td>
1215 <td class="dialoginfohead" align="center">Email</td>
1216 <td class="dialoginfohead" align="center">Employer</td>
1217 <td class="dialoginfohead" align="center">Work&#160;Phone</td>
1218 <td class="dialoginfohead" align="center">Ext.</td>
1219 <td class="dialoginfohead" align="center">Pager</td>
1220 </tr>
1221 <tr>
1222 <td class="datacell" align="center">
1223 <xsl:choose>
1224 <xsl:when test="p2name=''>
1225 &#160;
1226 </xsl:when>
1227 <xsl:otherwise>
1228 <xsl:value-of select="p2name"/>
1229 </xsl:otherwise>
1230 </xsl:choose>
1231 </td>
1232 <td class="datacell" align="center">
1233 <xsl:choose>
1234 <xsl:when test="p2relation=''>
1235 &#160;
1236 </xsl:when>
1237 <xsl:otherwise>
1238 <xsl:value-of select="p2relation"/>
1239 </xsl:otherwise>
1240 </xsl:choose>
1241 </td>
1242 <td class="datacell" align="center" nowrap="nowrap">
1243 <xsl:apply-templates select="p2phone"/>
1244 </td>
1245 <td class="datacell" align="center">
1246 <xsl:choose>
1247 <xsl:when test="p2email=''>
1248 &#160;
1249 </xsl:when>
1250 <xsl:otherwise>
1251 <a><xsl:attribute name="href">mailto:<xsl:value-of select="p2email"/></xsl:attribute><xsl:value-of select="p2email"/></a>
1252 </xsl:otherwise>
1253 </xsl:choose>
1254 </td>
1255 <td class="datacell" align="center">
1256 <xsl:choose>
1257 <xsl:when test="p2employer=''>
1258 &#160;
1259 </xsl:when>
1260 <xsl:otherwise>
1261 <xsl:value-of select="p2employer"/>
1262 </xsl:otherwise>
1263 </xsl:choose>
1264 </td>
1265 <td class="datacell" align="center" nowrap="nowrap">
1266 <xsl:apply-templates select="p2workphone"/>
1267 </td>
1268 <td class="datacell" align="center">
1269 <xsl:choose>
1270 <xsl:when test="p2ext=''>
1271 &#160;
1272 </xsl:when>
1273 <xsl:otherwise>
```

```

1275         <xsl:value-of select="p2ext"/>
1276     </xsl:otherwise>
1277 </xsl:choose>
1278 </td>
1279 <td class="datacell" align="center" nowrap="nowrap">
1280     <xsl:apply-templates select="p2pager"/>
1281 </td>
1282 </tr>
1283 <tr>
1284 <td colspan="7">&#160;</td>
1285 <td colspan="7" align="left" valign="top">
1286     <table cellpadding="0" cellspacing="0" valign="top" align="left">
1287         <tr>
1288             <td>
1289                 <table border="0" cellspacing="0" cellpadding="2" valign="top" align="left">
1290                     <tr>
1291                         <td class="dialoginfohead" valign="bottom" align="center">Address</td>
1292                     </tr>
1293                     <tr>
1294                         <xsl:apply-templates select="p2address"/>
1295                     </tr>
1296                 </table>
1297             </td>
1298             <td align="bottom" align="left">
1299                 <table border="0" cellspacing="0" cellpadding="2" valign="bottom" align="left">
1300                     <tr>
1301                         <td class="dialoginfohead" align="left" valign="center">Mail&#160;to?</td>
1302                         <td class="datacell" align="left" valign="center">
1303                             <xsl:choose>
1304                                 <xsl:when test="p2mailhere=''">
1305                                     &#160;
1306                                 </xsl:when>
1307                                 <xsl:otherwise>
1308                                     &#160;<xsl:value-of select="p2mailhere"/>&#160;
1309                                 </xsl:otherwise>
1310                             </xsl:choose>
1311                         </td>
1312                     </tr>
1313                 </table>
1314             </td>
1315         </tr>
1316     </table>
1317 </td>
1318 </tr>
1319 </table>
1320 </td>
1321 </tr>
1322 <tr>
1323 <td colspan="3" align="left" valign="top">
1324     <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
1325         <tr>
1326             <td class="dialoginfohead2" align="center">Parent/Guardian&#160;3</td>
1327             <td class="dialoginfohead" align="center">Relationship</td>
1328             <td class="dialoginfohead" align="center">Home&#160;Phone</td>
1329             <td class="dialoginfohead" align="center">Email</td>
1330             <td class="dialoginfohead" align="center">Employer</td>
1331             <td class="dialoginfohead" align="center">Work&#160;Phone</td>
1332             <td class="dialoginfohead" align="center">Ext.</td>
1333             <td class="dialoginfohead" align="center">Pager</td>
1334         </tr>
1335         <tr>
1336             <td class="datacell" align="center">
1337                 <xsl:choose>
1338                     <xsl:when test="p3name=''">
1339                         &#160;
1340                     </xsl:when>
1341                     <xsl:otherwise>
1342                         <xsl:value-of select="p3name"/>
1343                     </xsl:otherwise>
1344                 </xsl:choose>
1345             </td>
1346             <td class="datacell" align="center">
1347                 <xsl:choose>
1348                     <xsl:when test="p3relation=''">
1349                         &#160;
1350                     </xsl:when>
1351                     <xsl:otherwise>
1352                         <xsl:value-of select="p3relation"/>
1353                     </xsl:otherwise>
1354                 </xsl:choose>
1355             </td>
1356             <td class="datacell" align="center" nowrap="nowrap">
1357                 <xsl:apply-templates select="p3phone"/>
1358             </td>
1359             <td class="datacell" align="center">
1360                 <xsl:choose>
1361                     <xsl:when test="p3email=''">
1362                         &#160;
1363                     </xsl:when>
1364                     <xsl:otherwise>
1365                         <a><xsl:attribute name="href">mailto:<xsl:value-of select="p3email"/></xsl:attribute><xsl:value-of select="p3email"/></a>
1366                     </xsl:otherwise>
1367                 </xsl:choose>
1368             </td>
1369             <td class="datacell" align="center">
1370                 <xsl:choose>
1371                     <xsl:when test="p3employer=''">
1372                         &#160;

```

```

1373     </xsl:when>
1374     <xsl:otherwise>
1375         <xsl:value-of select="p3employer"/>
1376     </xsl:otherwise>
1377 </xsl:choose>
1378 </td>
1379 <td class="datacell" align="center" nowrap="nowrap">
1380     <xsl:apply-templates select="p3workphone"/>
1381 </td>
1382 <td class="datacell" align="center">
1383     <xsl:choose>
1384         <xsl:when test="p3ext=''>
1385             &#160;
1386         </xsl:when>
1387         <xsl:otherwise>
1388             <xsl:value-of select="p3ext"/>
1389         </xsl:otherwise>
1390     </xsl:choose>
1391 </td>
1392 <td class="datacell" align="center" nowrap="nowrap">
1393     <xsl:apply-templates select="p3pager"/>
1394 </td>
1395 </tr>
1396 <tr>
1397     <td&#160;</td>
1398     <td align="left" valign="top" colspan="7">
1399         <table cellpadding="0" cellspacing="0" valign="top" align="left">
1400             <tr>
1401                 <td>
1402                     <table border="0" cellspacing="0" cellpadding="2" valign="top" align="left">
1403                         <tr>
1404                             <td class="dialoginfohead" valign="bottom" align="center">Address</td>
1405                         </tr>
1406                         <tr>
1407                             <xsl:apply-templates select="p3address"/>
1408                         </tr>
1409                     </table>
1410                 </td>
1411                 <td valign="bottom" align="left">
1412                     <table border="0" cellspacing="0" cellpadding="2" valign="bottom" align="left">
1413                         <tr>
1414                             <td class="dialoginfohead" align="left" valign="center">Mail&#160;to?</td>
1415                             <td class="datacell" align="left" valign="center">
1416                                 <xsl:choose>
1417                                     <xsl:when test="p3mailhere=''>
1418                                         &#160;
1419                                     </xsl:when>
1420                                     <xsl:otherwise>
1421                                         &#160;<xsl:value-of select="p3mailhere"/>&#160;
1422                                     </xsl:otherwise>
1423                                 </xsl:choose>
1424                             </td>
1425                         </tr>
1426                     </table>
1427                 </td>
1428             </tr>
1429         </table>
1430 </td>
1431 </tr>
1432 <tr>
1433     <td align="left" valign="bottom" height="80" colspan="4">
1434         <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
1435             <tr><td colspan="5" height="12px" class="dialoginfohead2" valign="bottom">Cell Phone Numbers</td></tr>
1436             <tr>
1437                 <td class="dialoginfohead" align="center" valign="bottom">Name</td>
1438                 <td class="dialoginfohead" align="center" valign="bottom">Cell&#160;Phone</td>
1439                 <td&#160;</td>
1440                 <td class="dialoginfohead" align="center" valign="bottom">Name</td>
1441                 <td class="dialoginfohead" align="center" valign="bottom">Cell&#160;Phone</td>
1442             </tr>
1443             <tr>
1444                 <td class="datacell" valign="top" nowrap="nowrap">
1445                     <xsl:choose>
1446                         <xsl:when test="cpname=''>
1447                             &#160;
1448                         </xsl:when>
1449                         <xsl:otherwise>
1450                             <xsl:value-of select="cpname"/>
1451                         </xsl:otherwise>
1452                     </xsl:choose>
1453                 </td>
1454                 <td class="datacell" valign="top" nowrap="nowrap">
1455                     <xsl:apply-templates select="cpphone"/>
1456                 </td>
1457                 <td&#160;</td>
1458                 <td class="datacell" valign="top" nowrap="nowrap">
1459                     <xsl:choose>
1460                         <xsl:when test="cqname=''>
1461                             &#160;
1462                         </xsl:when>
1463                         <xsl:otherwise>
1464                             <xsl:value-of select="cqname"/>
1465                         </xsl:otherwise>
1466                     </xsl:choose>
1467                 </td>
1468                 <td class="datacell" valign="top" nowrap="nowrap">
1469                     <xsl:apply-templates select="cqphone"/>
1470                 </td>

```



```
1471         </tr>
1472     </table>
1473 </td>
1474 </tr>
1475 </table>
1476 </td>
1477 </tr>
1478 </table>
1479 </xsl:template>
1480
1481 <xsl:template match="contactphone1">
1482 <xsl:choose>
1483 <xsl:when test="area="">
1484     &#160;
1485 </xsl:when>
1486 <xsl:otherwise>
1487     (<xsl:value-of select="area"/>)
1488 </xsl:otherwise>
1489 </xsl:choose>
1490 &#160;<xsl:value-of select="phone"/>
1491 </xsl:template>
1492
1493 <xsl:template match="contactphone2">
1494 <xsl:choose>
1495 <xsl:when test="area="">
1496     &#160;
1497 </xsl:when>
1498 <xsl:otherwise>
1499     (<xsl:value-of select="area"/>)
1500 </xsl:otherwise>
1501 </xsl:choose>
1502 &#160;<xsl:value-of select="phone"/>
1503 </xsl:template>
1504
1505 <xsl:template match="contactphone3">
1506 <xsl:choose>
1507 <xsl:when test="area="">
1508     &#160;
1509 </xsl:when>
1510 <xsl:otherwise>
1511     (<xsl:value-of select="area"/>)
1512 </xsl:otherwise>
1513 </xsl:choose>
1514 &#160;<xsl:value-of select="phone"/>
1515 </xsl:template>
1516
1517 <xsl:template match="contactphone4">
1518 <xsl:choose>
1519 <xsl:when test="area="">
1520     &#160;
1521 </xsl:when>
1522 <xsl:otherwise>
1523     (<xsl:value-of select="area"/>)
1524 </xsl:otherwise>
1525 </xsl:choose>
1526 &#160;<xsl:value-of select="phone"/>
1527 </xsl:template>
1528
1529 <xsl:template match="phonenumber">
1530 <xsl:choose>
1531 <xsl:when test="area="">
1532     &#160;
1533 </xsl:when>
1534 <xsl:otherwise>
1535     (<xsl:value-of select="area"/>)
1536 </xsl:otherwise>
1537 </xsl:choose>
1538 &#160;<xsl:value-of select="phone"/>
1539 </xsl:template>
1540
1541 <xsl:template match="cpphone">
1542 <xsl:choose>
1543 <xsl:when test="area="">
1544     &#160;
1545 </xsl:when>
1546 <xsl:otherwise>
1547     (<xsl:value-of select="area"/>)
1548 </xsl:otherwise>
1549 </xsl:choose>
1550 &#160;<xsl:value-of select="phone"/>
1551 </xsl:template>
1552
1553
1554 <xsl:template match="cqphone">
1555 <xsl:choose>
1556 <xsl:when test="area="">
1557     &#160;
1558 </xsl:when>
1559 <xsl:otherwise>
1560     (<xsl:value-of select="area"/>)
1561 </xsl:otherwise>
1562 </xsl:choose>
1563 &#160;<xsl:value-of select="phone"/>
1564 </xsl:template>
1565
1566
1567 <xsl:template match="plphone">
1568 <xsl:choose>
```

```
1569 <xsl:when test="area="">
1570     &#160;
1571 </xsl:when>
1572 <xsl:otherwise>
1573     (<xsl:value-of select="area"/>)
1574 </xsl:otherwise>
1575 </xsl:choose>
1576 &#160;<xsl:value-of select="phone"/>
1577 </xsl:template>
1578
1579 <xsl:template match="p1workphone">
1580 <xsl:choose>
1581     <xsl:when test="area="">
1582         &#160;
1583     </xsl:when>
1584     <xsl:otherwise>
1585         (<xsl:value-of select="area"/>)
1586     </xsl:otherwise>
1587 </xsl:choose>
1588 &#160;<xsl:value-of select="phone"/>
1589 </xsl:template>
1590
1591 <xsl:template match="p1pager">
1592 <xsl:choose>
1593     <xsl:when test="area="">
1594         &#160;
1595     </xsl:when>
1596     <xsl:otherwise>
1597         (<xsl:value-of select="area"/>)
1598     </xsl:otherwise>
1599 </xsl:choose>
1600 &#160;<xsl:value-of select="phone"/>
1601 </xsl:template>
1602
1603 <xsl:template match="p2phone">
1604 <xsl:choose>
1605     <xsl:when test="area="">
1606         &#160;
1607     </xsl:when>
1608     <xsl:otherwise>
1609         (<xsl:value-of select="area"/>)
1610     </xsl:otherwise>
1611 </xsl:choose>
1612 &#160;<xsl:value-of select="phone"/>
1613 </xsl:template>
1614
1615 <xsl:template match="p2workphone">
1616 <xsl:choose>
1617     <xsl:when test="area="">
1618         &#160;
1619     </xsl:when>
1620     <xsl:otherwise>
1621         (<xsl:value-of select="area"/>)
1622     </xsl:otherwise>
1623 </xsl:choose>
1624 &#160;<xsl:value-of select="phone"/>
1625 </xsl:template>
1626
1627 <xsl:template match="p2pager">
1628 <xsl:choose>
1629     <xsl:when test="area="">
1630         &#160;
1631     </xsl:when>
1632     <xsl:otherwise>
1633         (<xsl:value-of select="area"/>)
1634     </xsl:otherwise>
1635 </xsl:choose>
1636 &#160;<xsl:value-of select="phone"/>
1637 </xsl:template>
1638
1639 <xsl:template match="p3phone">
1640 <xsl:choose>
1641     <xsl:when test="area="">
1642         &#160;
1643     </xsl:when>
1644     <xsl:otherwise>
1645         (<xsl:value-of select="area"/>)
1646     </xsl:otherwise>
1647 </xsl:choose>
1648 &#160;<xsl:value-of select="phone"/>
1649 </xsl:template>
1650
1651 <xsl:template match="p3workphone">
1652 <xsl:choose>
1653     <xsl:when test="area="">
1654         &#160;
1655     </xsl:when>
1656     <xsl:otherwise>
1657         (<xsl:value-of select="area"/>)
1658     </xsl:otherwise>
1659 </xsl:choose>
1660 &#160;<xsl:value-of select="phone"/>
1661 </xsl:template>
1662
1663 <xsl:template match="p3pager">
1664 <xsl:choose>
1665     <xsl:when test="area="">
1666         &#160;
```

```
1667 </xsl:when>
1668 <xsl:otherwise>
1669   (<xsl:value-of select="area"/>)
1670 </xsl:otherwise>
1671 </xsl:choose>
1672 &#160;<xsl:value-of select="phone"/>
1673 </xsl:template>
1674
1675 <xsl:template match="stuphone">
1676 <xsl:choose>
1677   <xsl:when test="area="">
1678     &#160;
1679   </xsl:when>
1680   <xsl:otherwise>
1681     (<xsl:value-of select="area"/>)
1682   </xsl:otherwise>
1683 </xsl:choose>
1684 &#160;<xsl:value-of select="phone"/>
1685 </xsl:template>
1686
1687 <xsl:template match="cwphone">
1688 <xsl:choose>
1689   <xsl:when test="area="">
1690     &#160;
1691   </xsl:when>
1692   <xsl:otherwise>
1693     (<xsl:value-of select="area"/>)
1694   </xsl:otherwise>
1695 </xsl:choose>
1696 &#160;<xsl:value-of select="phone"/>
1697 </xsl:template>
1698
1699
1700 <xsl:template match="address">
1701 <td class="datacell" align="left"><xsl:value-of select="street"/><br/>&#160;<xsl:value-of select="state"/>&#160;&#160;&#160;<xsl:value-of select="zip"/></td>
1702 </xsl:template>
1703
1704
1705 <xsl:template match="pladdress">
1706 <td class="datacell" align="left">
1707   <xsl:value-of select="street"/><br/>
1708   <xsl:choose>
1709     <xsl:when test="city="">
1710       &#160;
1711     </xsl:when>
1712     <xsl:otherwise>
1713       <xsl:value-of select="city"/>,&#160;
1714     </xsl:otherwise>
1715 </xsl:choose>
1716 <xsl:choose>
1717   <xsl:when test="state="">
1718     &#160;&#160;&#160;
1719   </xsl:when>
1720   <xsl:otherwise>
1721     <xsl:value-of select="state"/>.&#160;&#160;&#160;
1722   </xsl:otherwise>
1723 </xsl:choose>
1724 <xsl:value-of select="zip"/>
1725 </td>
1726 </xsl:template>
1727
1728 <xsl:template match="pmailaddress">
1729 <td class="datacell" align="left">
1730   <xsl:value-of select="street"/><br/>
1731   <xsl:choose>
1732     <xsl:when test="city="">
1733       &#160;
1734     </xsl:when>
1735     <xsl:otherwise>
1736       <xsl:value-of select="city"/>,&#160;
1737     </xsl:otherwise>
1738 </xsl:choose>
1739 <xsl:choose>
1740   <xsl:when test="state="">
1741     &#160;&#160;&#160;
1742   </xsl:when>
1743   <xsl:otherwise>
1744     <xsl:value-of select="state"/>.&#160;&#160;&#160;
1745   </xsl:otherwise>
1746 </xsl:choose>
1747 <xsl:value-of select="zip"/>
1748 </td>
1749 </xsl:template>
1750
1751 <xsl:template match="p2address">
1752 <td class="datacell" align="left">
1753   <xsl:value-of select="street"/><br/>
1754   <xsl:choose>
1755     <xsl:when test="city="">
1756       &#160;
1757     </xsl:when>
1758     <xsl:otherwise>
1759       <xsl:value-of select="city"/>,&#160;
1760     </xsl:otherwise>
1761 </xsl:choose>
1762 <xsl:choose>
1763   <xsl:when test="state="">
1764     &#160;&#160;&#160;
```

```
1765 </xsl:when>
1766 <xsl:otherwise>
1767 <xsl:value-of select="state"/>.&#160;&#160;&#160;
1768 </xsl:otherwise>
1769 </xsl:choose>
1770 <xsl:value-of select="zip"/>
1771 </td>
1772 </xsl:template>
1773
1774 <xsl:template match="p3address">
1775 <td class="datacell1" align="left">
1776 <xsl:value-of select="street"/><br/>
1777 <xsl:choose>
1778 <xsl:when test="city=''">
1779 &#160;
1780 </xsl:when>
1781 <xsl:otherwise>
1782 <xsl:value-of select="city"/>,&#160;
1783 </xsl:otherwise>
1784 </xsl:choose>
1785 <xsl:choose>
1786 <xsl:when test="state=''">
1787 &#160;&#160;&#160;&#160;
1788 </xsl:when>
1789 <xsl:otherwise>
1790 <xsl:value-of select="state"/>.&#160;&#160;&#160;&#160;
1791 </xsl:otherwise>
1792 </xsl:choose>
1793 <xsl:value-of select="zip"/>
1794 </td>
1795 </xsl:template>
1796
1797 <xsl:template match="disabilities">
1798 <table border="0" cellspacing="0" cellpadding="2" align="left" valign="top">
1799 <xsl:apply-templates/>
1800 </table>
1801 </xsl:template>
1802
1803 <xsl:template match="db">
1804 <tr><td class="dialoginfohead" align="left"><xsl:value-of select="code"/></td><td class="dialoginfohead" align="left"><-</td><td class="dialoginfohead" align="left"><xsl:value-of select="desc"/></td></tr>
1805 </xsl:template>
1806
1807 <xsl:template name="contentstyle">
1808 .datacell {font-size: 10px; font-family: Arial, Verdana; color: #3E6595; border-left: 1 solid; border-right: 1 solid; border-top: 2 solid; border-bottom: 2 solid ;border-color:#3E6595;}
1809 .datacell2 {font-size: 10px; font-family: Arial, Verdana; color: #3E6595; border-left: 2 solid; border-right: 2 solid; border-top: 1 solid; border-bottom: 1 solid ;border-color:#3E6595;}
1810 .datacell_wt {font-size: 10px; font-family: Arial, Verdana; color: #3E6595; border-left: 1 solid; border-right: 1 solid; border-top: 2 solid; border-bottom: 2 solid ;border-color:#3E6595;background-color:#ffffff;}
1811 .dialoginfohead {font-size: 10px; font-family: Arial, Verdana; font-weight: bold; color: #3E6595;}
1812 .persistantBox {border: 1 solid #3E6595; }
1813 .warningfill {background-color:#dd0000;font-size: 10px; font-family: Arial, Verdana; color: #ffffff; border-left: 1 solid; border-right: 1 solid; border-top: 2 solid; border-bottom: 2 solid ;border-color:#3E6595;}
1814 .ncwarningfill {background-color:#dd0000;font-size: 10px; font-family: Arial, Verdana; color: #FFFF00; border-left: 2 solid; border-right: 2 solid; border-top: 1 solid; border-bottom: 1 solid ;border-color:#3E6595;}
1815 .caution {font-size: 10px; font-family: Arial, Verdana; font-weight: bold;border: 1 solid #cc0000;background-color:#fff000;color:#cc0000;}
1816 .gobox {font-size: 10px; font-family: Arial, Verdana; font-weight: bold;border: 1 solid #cc0000;background-color:#008800;color:#cccc00;}
1817 .healthbox {font-size: 10px; font-family: Arial, Verdana; font-weight: bold;border: 1 solid #cc0000;background-color:#FFFFFF;color:#dd0000;}
1818 .dialoginfohead2 {font-size: 11px; font-family: Arial, Verdana; font-weight: bold; color: #3E6595;}
1819 .tabtext {font-size: 11px; font-family: Arial, Verdana; font-weight: bold; color: #3E6595;text-decoration:none}
1820 .buttonclass {font-size: 10px; font-family: Arial, Verdana; font-weight: bold; height:20;}
1821 .teacherinfo {font-size: 11px; font-family: Arial, Verdana; color: #3E6595;}
1822 .teacherinfo2 {font-size: 13px; font-family: Arial, Verdana; color: #3E6595;}
1823 .singleBorder {border: 1 solid #3E6595;}
1824 .title2 {font-size: 18px; font-family:Comic Sans MS, Arial, Helvetica, sans-serif; color: #3E6595;}
1825 .mainborder {background-color:#C5E5ff; border-top:8px; border-bottom:8px; border-left:8px; border-right:8px; border-style:ridge; border-color:#3E65ff; height:100%; width:100%;margin:0;padding:0;background-image: url("/images/logo:
1826 .logoffbutton {font-size: 11px; font-family: Arial, Verdana; font-weight: bold; height:30; width:60; color:#aa0000;border: 3 #ff0000 outset}
1827 </xsl:template>
1828
1829 </xsl:stylesheet>
```

```

1  /*****
2  studentinfo.js
3  -- 1/15/2002 Daniel J. Moore
4
5  Used with studentinfo-html.xml and StudentInfo.java to display Student Information to Teachers and School Staff.
6  This script controls the operation of the user interface and controls the functioning of Tabs.
7
8  FUNCTIONS:
9
10     init()
11         Initialize display
12
13     displaySearchResults()
14         Start the search specified by the entries in the search form.
15
16     getThisStudent(studentid)
17         PARAMETERS:
18         [studentid = string of student id to retrieve.]
19
20         Get the student information for this specific student
21
22     replaceSpaces(strIn)
23         PARAMETERS:
24         [strIn = input string.]
25
26         RETURNS:
27         [String with all spaces replaced with &#160;]
28
29         Replaces the spaces in a given string with the appropriate escape code.
30
31     newSearch()
32         Clear the contents of the search box.
33
34     TabSelectTable(iSelectedTab, iHorzPerChar, iVertLineHeight, iOverallTabHeight)
35         PARAMETERS:
36         [iSelectedTab = The selected tab]
37         [iHorzPerChar = Used to set the number of pixels per character to adjust the horizontal size of tabs]
38         [iVertLineHeight = height of tabs in pixels]
39         [iOverallTabHeight = height of tab section (overall)]
40
41         Used to control the tabs and display the selected hidden 'div' according to which tab was selected.
42         Tabs in this script have corners that are actually tables with various background colors in each
43         cell to create the look of corners of the tabs. This is done to ensure quick redrawing of tabs
44         as different tabs are selected without having to wait for gif images to redraw.
45
46         Tab data is contained in two arrays;
47         arrTabLabel (contains Tab Titles)
48         arrDisplayDiv (name of corresponding Hidden DIV to display when the tab is selected).
49         These two arrays are global and populated in studentinfo-html.xml.
50
51         NOTE: This is only guaranteed to work with IE 4.0 or greater...
52
53     getSelectedIndex(el, stvalue)
54         PARAMETERS:
55         [el = document element to test]
56         [stvalue = value to search for]
57
58         RETURNS:
59         [The index for an item that matches stvalue in a select list, integer
60         (Returns -1 if no index found)]
61
62         Used to aid in setting the selected item in a dropdown list to a value passed in
63         from the XML data
64
65 -----
66 REVISIONS
67
68     Initially created 1-10-2002 DJM
69
70     2-5-2002 DJM
71         Adjusted to work with studentinfo-html.xml
72
73
74
75
76 *****/
77 var TabCount = 0;
78
79 //These variables are used to control colors of tab parts as tabs are selected
80 //and displays changed...
81 var strSelectedColor = "#ffffff"; //Background color of selected tab
82 var strTextSelectedColor = "#DD0000"; //Text color of selected tab
83 var strTextUnSelectedColor = "#3E6595"; //Text color of unselected tabs
84 var strBgColor = "#C5E5ff"; //Background color
85 var strSolidLine = "#3E6595"; //Color of the lines in the tabs
86 var strLightAls = "#b4c7de"; //Used to provide light aliasing in some pixels of corners
87 var strDarkAls = "#638cbe"; //Used to provide dark aliasing in some pixels of corners
88
89 //stylesheet to use for text in tabs...
90 var strTextStyle = "tabtext";
91
92 //Initialize
93 function init(){
94     TabSelectTable(1,8,5,40);
95
96 }
97
98

```

```
99
100 //start the search specified...
101 function displaySearchResults(){
102     //arrTabLabel = new Array("Search Results");//
103     //arrDisplayDiv = new Array("StudentsFoundList");//
104     //TabSelectTable(1,8,5,40);
105     document.StuSearch.method.value = "studentsearch";
106     document.StuSearch.submit();
107 }
108
109 //get this specific student
110 function getThisStudent(studentid){
111     document.StuSearch.method.value = "getstudent";
112     document.StuSearch.studentfind.value = studentid;
113     document.StuSearch.submit();
114 }
115
116 //REPLACE SPACES WITH $#160;...
117 function replaceSpaces(strIn){
118     var strOut = "";
119     var re = ' ';
120     if(strIn.length>0){
121         strOut = strIn;
122         while (strOut.search(re) != -1){
123             strOut = strIn.replace(re, "&#160;");
124             strIn = strOut;
125         }
126     }
127     return strOut;
128 }
129
130 //Clear the contents of the search box
131 function newSearch(){
132     document.StuSearch.searchstulast.value = "";
133     document.StuSearch.searchstuid.value = "";
134     document.StuSearch.searchstufirst.value = "";
135     document.StuSearch.searchstuhomerom.value = "";
136 }
137
138 //log out of this application...
139 function goLogout(){
140     document.StuSearch.method.value = "logout";
141     document.StuSearch.submit();
142
143     //window.location.href="https://sequoia.inside.seattleschools.org/area/tools/ToolBox?method=logout";//
144     //
145 }
146
147 //Used to control the tabs and display the selected 'div' according to which tab was selected.
148 function TabSelectTable(iSelectedTab, iHorzPerChar, iVertLineHeight, iOverallTabHeight){
149
150     var iSelectedIndex;
151
152     //First determine which tab was selected...
153     if (iSelectedTab < 1 || isNaN(iSelectedTab) || iSelectedTab > arrTabLabel.length){
154         //initial setup... default to the first tab... (if other default tab desired, make change here)
155         iSelectedIndex = 0;
156     }else{
157         iSelectedIndex = iSelectedTab - 1;
158     }
159
160     //pixel width per char is set here... (may need to tweak this for proper display width if using var. width char set).
161     //...this is used to determine the tab width.
162     var iCharAveWidth = 7;
163     if(!isNaN(iHorzPerChar) && iHorzPerChar > 0){
164         iCharAveWidth = iHorzPerChar;
165     }
166
167     //Each tab width is set by a horizontal line whose size is determined by the max # of characters in all the tab strings...
168
169     //Determin the max char count...
170     var maxchar = 0;
171     for(i=0; i< arrTabLabel.length; i++){
172         if(arrTabLabel[i].length > maxchar){
173             maxchar = arrTabLabel[i].length;
174         }
175     }
176
177     //set horizontal line width...
178     var iHorzLineWidth = iCharAveWidth * maxchar;
179     var strHorzLine = iHorzLineWidth;
180
181     //set virtical line height (virtical lines between tabs
182     var iTabLineHeight = 16;
183     if(!isNaN(iVertLineHeight) && iVertLineHeight > 0){
184         iTabLineHeight = iVertLineHeight;
185     }
186
187     //establish columns needed to create tabs ==> (tab count) + 2
188     var iTotColumns = arrTabLabel.length + 2;
189
190     var setStyle = "";
191     setStyle += "<style type='text/css'><!--";
192     setStyle += " .emptycell {height:2;width:2;font-size: 2px;}";
193     setStyle += " .cellLine {height:2;width:2;font-size: 2px;color: " + strSolidLine + ";background-color: " + strSolidLine + ";}";
194     setStyle += " .cellLineA1s {height:2;width:2;font-size: 2px;color: " + strLightA1s + ";background-color: " + strLightA1s + ";}";
195     setStyle += " .cellLineA1s2 {height:2;width:2;font-size: 2px;color: " + strDarkA1s + ";background-color: " + strDarkA1s + ";}";
196     setStyle += " .cellLineA1s2 {height:2;width:2;font-size: 2px;color: " + strDarkA1s + ";background-color: " + strDarkA1s + ";}";
```

```

197 setStyle += ".cellselect {height:2;width:2;font-size: 2px;color: " + strSelectedColor + ";background-color: " + strSelectedColor + "};"
198 setStyle += ".tabtextsel {text-align:center;font-size: 12px; font-family: Arial, Verdana; cursor:hand; font-weight: bold; color: " + strTextSelectedColor + "};"
199 setStyle += ".text-decoration:none;background-color:" + strSelectedColor + ";border-left: solid 1 " + strSolidLine + ";border-right: solid 1 " + strSolidLine + "; }"
200 setStyle += ".tabtext {text-align:center; font-size: 12px; font-family: Arial, Verdana; cursor:hand; font-weight: bold; color: " + strTextUnSelectedColor + "};"
201 setStyle += ".text-decoration:none;background-color:" + strBgColor + ";border-left: solid 1 " + strSolidLine + ";border-right: solid 1 " + strSolidLine + ";}"
202 setStyle += ".tabtext1 {text-align:center;font-size: 12px; font-family: Arial, Verdana; font-weight: bold; color: " + strSolidLine + ";text-decoration:none;"
203 setStyle += "background-color:" + strBgColor + ";border-right: solid 1 " + strSolidLine + "};"
204 setStyle += ".tabtext2 {text-align:center;font-size: 12px; font-family:Arial, Verdana; font-weight: bold; color: " + strSolidLine + ";text-decoration:none;"
205 setStyle += "background-color:" + strBgColor + ";border-left: solid 1 " + strSolidLine + "};"
206 setStyle += "-></style>"
207
208
209 var StrReturn = "<TABLE border=\"0\" cellspacing=\"0\" cellpadding=\"0\" align=\"center\" valign=\"top\" bgcolor=\"\" + strBgColor + " height=\"\" + iOverallTabHeight + "\" width=\"100%\">";
210 var StrTBorder = "<TR>"
211 var StrTop = "<TR>"
212 var StrMid = "<TR>";
213 var StrBot = "<TR>";
214
215 r = arrTabLabel.length;
216 TabCount = r;
217 for(t=0; t < r ; t++){
218
219     if(t==0){//first tab
220
221         if( t == iSelectedIndex){ //this tab is selected
222
223             StrTBorder += "<td ALIGN=\"LEFT\" HEIGHT=\"\" + iVertLineHeight + "\" WIDTH=\"8\" style=\"font-size:2px;\">#160;</td>"
224             StrTop += "<td ALIGN=\"LEFT\" WIDTH=\"8\" style=\"font-size:2px;\">#160;</td>";
225             StrMid += "<td ALIGN=\"LEFT\" WIDTH=\"7\" class=\"tabtext1\">#160;</td>";
226             StrBot += "<td ALIGN=\"LEFT\" WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:2px;\">";
227             StrBot += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
228             StrBot += "<TR><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"cellLine\">.</TD></TR>";
229             StrBot += "<TR><TD class=\"emptycell\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellselect\">.</TD></TR>";
230             StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls2\">.</TD><TD class=\"cellselect\">.</TD><TD class=\"cellselect\">.</TD></TR>";
231             StrBot += "</TABLE></td>";
232
233             StrTBorder += "<td style=\"font-size:2px;\">#160;</td><td style=\"font-size:2px;\">#160;</td><td style=\"font-size:2px;\">#160;</td>";
234
235             StrTop += "<td ALIGN=\"left\" WIDTH=\"8\" HEIGHT=\"6\">";
236             StrTop += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"left\">";
237             StrTop += "<TR><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"cellLine\">.</TD></TR>";
238             StrTop += "<TR><TD class=\"emptycell\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"cellselect\">.</TD></TR><TR>";
239             StrTop += "<TD class=\"cellLine\">.</TD><TD class=\"cellselect\">.</TD><TD class=\"cellselect\">.</TD><TD class=\"cellselect\">.</TD></TR></TABLE></td>";
240             StrTop += "<td onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");";
241             StrTop += "return false;\" style=\"background-color: " + strSelectedColor + ";cursor:hand;border-top: 2 solid " + strSolidLine + " ";
242             StrTop += "width: \" + strHorzLine + \"; height:10px;font-size:2px;\">#160;</td>";
243             StrTop += "<td><TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"left\">";
244             StrTop += "<TR><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls2\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
245             StrTop += "<TR><TD class=\"cellselect\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"emptycell\">.</TD></TR>";
246             StrTop += "<TR><TD class=\"cellselect\">.</TD><TD class=\"cellselect\">.</TD><TD class=\"cellselect\">.</TD><TD class=\"cellLine\">.</TD></TR>";
247             StrTop += "</TABLE></td>";
248
249             StrMid += "<td colspan=\"3\" align=\"CENTER\" valign=\"middle\">";
250             StrMid += " onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");return false;\"";
251             StrMid += " class=\"tabtextsel\" width=\"\" + (strHorzLine + 15) + "\"> " + replaceSpaces(arrTabLabel[t]) + " </td>";
252
253             StrBot += "<td ALIGN=\"left\" bgcolor=\"\" + strSelectedColor + "\"></td>";
254             StrBot += "<td onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");return false;\" ALIGN=\"left\" HEIGHT=\"6\" bgcolor=\"\" + strSelectedColor + "\"></td>";
255
256             if(t == r-1){//There are no tabs after this one
257
258                 StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" ALIGN=\"left\" bgcolor=\"\" + strSelectedColor + "\"></td><td HEIGHT=\"6\" WIDTH=\"8\">";
259                 StrBot += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
260                 StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
261                 StrBot += "<TR><TD class=\"cellselect\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"emptycell\">.</TD></TR>";
262                 StrBot += "<TR><TD class=\"cellselect\">.</TD><TD class=\"cellselect\">.</TD><TD class=\"cellLineAls2\">.</TD><TD class=\"cellLine\">.</TD></TR>";
263                 StrBot += "</TABLE></td>";
264
265             }else{//There are other tabs
266
267                 StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" ALIGN=\"left\" bgcolor=\"\" + strSelectedColor + "\"></td>";
268             }
269
270         }
271     }else{//not selected
272
273         StrTBorder += "<td ALIGN=\"LEFT\" HEIGHT=\"\" + iVertLineHeight + "\" WIDTH=\"8\" style=\"font-size:2px;\">#160;</td>"
274         StrTop += "<td ALIGN=\"LEFT\" WIDTH=\"8\" style=\"font-size:2px;\">#160;</td>";
275         StrMid += "<td ALIGN=\"LEFT\" WIDTH=\"7\" class=\"tabtext1\">#160;</td>";
276         StrBot += "<td ALIGN=\"LEFT\" WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:2px;\">";
277         StrBot += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
278         StrBot += "<TR><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"cellLine\">.</TD></TR>";
279         StrBot += "<TR><TD class=\"emptycell\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls\">.</TD></TR>";
280         StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD></TR>";
281         StrBot += "</TABLE></td>";
282
283         StrTBorder += "<td style=\"font-size:2px;\">#160;</td><td style=\"font-size:2px;\">#160;</td><td style=\"font-size:2px;\">#160;</td>";
284
285         StrTop += "<td ALIGN=\"left\" WIDTH=\"8\" HEIGHT=\"6\">";
286         StrTop += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" bgcolor=\"\" + strBgColor + "\" align=\"left\">";
287         StrTop += "<TR><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"cellLineAls\">.</TD></TR>";
288         StrTop += "<TR><TD class=\"emptycell\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"emptycell\">.</TD></TR>";
289         StrTop += "<TR><TD class=\"cellLine\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
290         StrTop += "</TABLE></td>";
291         StrTop += "<td onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");";
292         StrTop += "return false;\" style=\"border-top: 2 solid " + strSolidLine + " "; width: \" + strHorzLine + \"; cursor:hand;height:10px;";
293         StrTop += "background-color: " + strBgColor + " ;height:6px;font-size:1px;\">#160;</td>";
294

```

```

295 StrTop += "<td><TABLE CELLSPACING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" bgcolor=\"\" + strBgColor + "\" align=\"center\">";
296 StrTop += "<TR><TD class=\"cellLine\"></TD><TD class=\"cellLineA2\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD></TR>";
297 StrTop += "<TR><TD class=\"emptycell\"></TD><TD class=\"cellLineA1\"></TD><TD class=\"cellLine\"></TD><TD class=\"emptycell\"></TD></TR>";
298 StrTop += "<TR><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD><TD class=\"cellLine\"></TD></TR>";
299 StrTop += "</TABLE></td>";
300
301 StrMid += "<td colspan=\"3\" align=\"center\" valign=\"middle\">";
302 StrMid += " onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");return false;\"";
303 StrMid += " class=\"tabtext\" width=\"\" + (strHorzLine + 15) + \"\">&#160;\" + replaceSpaces(arrTabLabel[t]) + "&#160;</td>";
304
305 StrBot += "<td ALIGN=\"left\" HEIGHT=\"6\" style=\"font-size:1px;border-bottom: 2 solid \" + strSolidLine + "; height:6;\">&#160;</td>";
306 StrBot += "<td onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");return false;\"";
307 StrBot += " WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:1px;border-bottom: 2 solid \" + strSolidLine + "; height:6;\">&#160;</td>";
308
309
310 if (t == r-1){//There are no tabs after this one
311
312     StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:1px;border-bottom: 2 solid \" + strSolidLine + "; height:6;\">&#160;</td>";
313
314 }else{//There are other tabs
315
316     if ( t+1 == iSelectedIndex){
317         StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:2px;\">";
318         StrBot += "<TABLE CELLSPACING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"right\">";
319         StrBot += "<TR><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD></TR>";
320         StrBot += "<TR><TD class=\"emptycell\"></TD><TD class=\"cellLineA1\"></TD><TD class=\"cellLine\"></TD><TD class=\"cellselect\"></TD></TR>";
321         StrBot += "<TR><TD class=\"cellLine\"></TD><TD class=\"cellLineA2\"></TD><TD class=\"cellselect\"></TD><TD class=\"cellselect\"></TD></TR>";
322         StrBot += "</TABLE></td>";
323
324     }else{
325         StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:1px;border-bottom: 2 solid \" + strSolidLine + "; height:6;\">&#160;</td>";
326     }
327
328
329
330 }
331
332 }
333
334 }else{
335
336     if ( t == iSelectedIndex){ //this tab is selected
337
338         if ( t-1 == iSelectedIndex){
339             StrBot += "<td HEIGHT=\"6\"><TABLE CELLSPACING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
340             StrBot += "<TR><TD class=\"cellLine\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD></TR>";
341             StrBot += "<TR><TD class=\"cellselect\"></TD><TD class=\"cellLine\"></TD><TD class=\"cellLineA1\"></TD><TD class=\"emptycell\"></TD></TR>";
342             StrBot += "<TR><TD class=\"cellselect\"></TD><TD class=\"cellselect\"></TD><TD class=\"cellLineA2\"></TD><TD class=\"cellLine\"></TD></TR>";
343             StrBot += "</TABLE></td>";
344
345         }
346
347
348
349
350
351
352 StrTBorder += "<td style=\"font-size:2px;\">&#160;</td><td style=\"font-size:2px;\">&#160;</td><td style=\"font-size:2px;\">&#160;</td>";
353
354 StrTop += "<td WIDTH=\"8\">";
355 StrTop += "<TABLE CELLSPACING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
356 StrTop += "<TR><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD><TD class=\"cellLineA2\"></TD><TD class=\"cellLine\"></TD></TR>";
357 StrTop += "<TR><TD class=\"emptycell\"></TD><TD class=\"cellLine\"></TD><TD class=\"cellLineA1\"></TD><TD class=\"cellselect\"></TD></TR>";
358 StrTop += "<TR><TD class=\"cellLine\"></TD><TD class=\"cellselect\"></TD><TD class=\"cellselect\"></TD><TD class=\"cellselect\"></TD></TR>";
359 StrTop += "</TABLE></td>";
360 StrTop += "<td onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");";
361 StrTop += "return false;\" style=\"background-color: \" + strSelectedColor + "; cursor:hand;";
362 StrTop += "border-top: 2 solid \" + strSolidLine + "; width:\"+strHorzLine+\";height:10;height:6;font-size:2px;\">&#160;</td>";
363 StrTop += "<td WIDTH=\"8\">";
364 StrTop += "<TABLE CELLSPACING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
365 StrTop += "<TR><TD class=\"cellLine\"></TD><TD class=\"cellLineA2\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD></TR>";
366 StrTop += "<TR><TD class=\"cellselect\"></TD><TD class=\"cellLineA1\"></TD><TD class=\"cellLine\"></TD><TD class=\"emptycell\"></TD></TR>";
367 StrTop += "<TR><TD class=\"cellselect\"></TD><TD class=\"cellselect\"></TD><TD class=\"cellselect\"></TD><TD class=\"cellLine\"></TD></TR>";
368 StrTop += "</TABLE></td>";
369
370 StrMid += "<td colspan=\"3\" align=\"center\" valign=\"middle\">";
371 StrMid += " onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");";
372 StrMid += "return false;\" class=\"tabtextsel\" width=\"\" + (strHorzLine + 15) + \"\">&#160;\" + replaceSpaces(arrTabLabel[t]) + "&#160;</td>";
373
374 StrBot += "<td bgcolor=\"\" + strSelectedColor + \"\"></td><td onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");";
375 StrBot += "return false;\" bgcolor=\"\" + strSelectedColor + \"\"></td>";
376
377
378
379
380 if (t == r-1){//There are no tabs after this one
381
382     StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" bgcolor=\"\" + strSelectedColor + \"\"></td>";
383     StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\">";
384     StrBot += "<TABLE CELLSPACING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
385     StrBot += "<TR><TD class=\"cellLine\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD><TD class=\"emptycell\"></TD></TR>";
386     StrBot += "<TR><TD class=\"cellselect\"></TD><TD class=\"cellLine\"></TD><TD class=\"cellLineA1\"></TD><TD class=\"emptycell\"></TD></TR>";
387     StrBot += "<TR><TD class=\"cellselect\"></TD><TD class=\"cellselect\"></TD><TD class=\"cellLineA2\"></TD><TD class=\"cellLine\"></TD></TR>";
388     StrBot += "</TABLE></td>";
389
390 }else{//There are other tabs
391
392     StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" bgcolor=\"\" + strSelectedColor + \"\"></td>";

```



```
393     }
394
395
396
397     }else{//not selected
398
399
400     if( t-1 == iSelectedIndex){
401         StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\">";
402         StrBot += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
403         StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
404         StrBot += "<TR><TD class=\"cellselect\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"emptycell\">.</TD></TR>";
405         StrBot += "<TR><TD class=\"cellselect\">.</TD><TD class=\"cellselect\">.</TD><TD class=\"cellLineAls2\">.</TD><TD class=\"cellLine\">.</TD></TR>";
406         StrBot += "</TABLE></td>";
407     }else{
408         StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:2px;\">";
409         StrBot += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
410         StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
411         StrBot += "<TR><TD class=\"cellLineAls\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"emptycell\">.</TD></TR>";
412         StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD></TR>";
413         StrBot += "</TABLE></td>";
414     }
415
416
417     StrTBorder += "<td style=\"font-size:2px;\">&#160;</td><td style=\"font-size:2px;\">&#160;</td><td style=\"font-size:2px;\">&#160;</td>";
418
419     StrTop += "<td><TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" bgcolor=\"\" + strBgColor + "\" align=\"center\">";
420     StrTop += "<TR><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"cellLineAls2\">.</TD><TD class=\"cellLine\">.</TD></TR>";
421     StrTop += "<TR><TD class=\"emptycell\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"emptycell\">.</TD></TR>";
422     StrTop += "<TR><TD class=\"cellLine\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
423     StrTop += "</TABLE></td>";
424     StrTop += "<td onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");\"";
425     StrTop += "return false;\" style=\"border-top: 2 solid \" + strSolidLine + \"; cursor:hand; width:\"+strHorzLine+\"; height:10;\"";
426     StrTop += "background-color:\" + strBgColor + \"; height:6; font-size:1px;\">&#160;</td>";
427     StrTop += "<td><TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" bgcolor=\"\" + strBgColor + "\" align=\"center\">";
428     StrTop += "<TR><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls2\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
429     StrTop += "<TR><TD class=\"emptycell\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"emptycell\">.</TD></TR>";
430     StrTop += "<TR><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"cellLine\">.</TD></TR>";
431     StrTop += "</TABLE></td>";
432
433     StrMid += "<td colspan=\"3\" align=\"center\" valign=\"middle\">";
434     StrMid += " onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");return false;\"";
435     StrMid += " class=\"tabtext\" width=\"\" + (strHorzLine + 15) + "\">&#160;\" + replaceSpaces(arrTabLabel[t]) + "&#160;</td>";
436
437     StrBot += "<td onClick=\"TabSelectTable(\"+(t+1)+\", \" + iCharAveWidth + \", \" + iVertLineHeight + \", \" + iOverallTabHeight + \");\"";
438     StrBot += "return false;\" HEIGHT=\"6\" style=\"font-size:1px;border-bottom: 2 solid \" + strSolidLine + \"; height:6;\">&#160;</td>";
439
440
441
442
443     if(t == r-1){//There are no tabs after this one
444         StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:1px;border-bottom: 2 solid \" + strSolidLine + \"; height:6;\">&#160;</td>";
445         StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:2px;\">";
446         StrBot += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
447         StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
448         StrBot += "<TR><TD class=\"cellLineAls\">.</TD><TD class=\"cellLineAls2\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD></TR>";
449         StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellLine\">.</TD></TR>";
450         StrBot += "</TABLE></td>";
451     }
452     }else{//There are other tabs
453
454     if( t+1 == iSelectedIndex){
455         StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:2px;\">";
456         StrBot += "<TABLE CELLPADDING=\"0\" CELLSPACING=\"0\" BORDER=\"0\" WIDTH=\"8\" HEIGHT=\"6\" align=\"center\">";
457         StrBot += "<TR><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"emptycell\">.</TD><TD class=\"cellLine\">.</TD></TR>";
458         StrBot += "<TR><TD class=\"emptycell\">.</TD><TD class=\"cellLineAls\">.</TD><TD class=\"cellLine\">.</TD><TD class=\"cellselect\">.</TD></TR>";
459         StrBot += "<TR><TD class=\"cellLine\">.</TD><TD class=\"cellLineAls2\">.</TD><TD class=\"cellselect\">.</TD><TD class=\"cellselect\">.</TD></TR>";
460         StrBot += "</TABLE></td>";
461     }else{
462         StrBot += "<td WIDTH=\"8\" HEIGHT=\"6\" style=\"font-size:1px;border-bottom: 2 solid \" + strSolidLine + \"; height:6;\">&#160;</td>";
463     }
464
465
466     }
467
468
469     }
470
471     }
472
473     }
474
475     StrTBorder += "<td style=\"font-size:2px;\">&#160;</td>";
476     StrTBorder += "<td style=\"font-size:2px;\">&#160;</td>";
477
478     StrTop += "<td style=\"font-size:2px;\">&#160;</td>";
479     StrTop += "<td style=\"font-size:2px;\">&#160;</td>\"; //width:100%
480
481     StrMid += "<td colspan=\"2\" class=\"tabtext2\">&#160;</td>\"; // width=\"100%\"
482
483     StrBot += "<td HEIGHT=\"6\" style=\"font-size:1px;border-bottom: 2 solid \" + strSolidLine + \"; height:6;background-color:\" + strBgColor + "\">&#160;</td>";
484
485     StrTBorder += "</TR>";
486     StrTop += "</TR>";
487     StrMid += "</TR>";
488     StrBot += "</TR>";
489
490     StrReturn += setStyle + StrTBorder + StrTop + StrMid + StrBot;
```

```
491     StrReturn += "</TABLE>";
492
493     document.getElementById("tabbar").innerHTML = StrReturn;
494     document.getElementById("tabcontent").innerHTML = document.getElementById(arrDisplayDiv[iSelectedIndex]).innerHTML;
495
496 }
497
498 //return the selected index for an item that matches stvalue...
499 function getSelectedIndex(el, stvalue){
500     if(el.length > 0){
501         var existIndex = -1;
502         for(var n=0;n<el.length;n++){
503             var strID = el.options[n].getAttribute("id").toUpperCase();
504             if(strID == stvalue.toUpperCase()){
505                 existIndex = n;
506             }
507         }
508         if(existIndex != -1){
509             return existIndex;
510         }
511     }
512     return 0;
513 }
514
515 }
```

```
1 ECHO OFF
2 REM =====
3 REM import_Teacher2sis_files.cmd
4 REM
5 REM Used to import the following files into the Teacher2SIS database:
6 REM
7 REM CENT-EXTRACT.TXT - List of SchoolID, Student ID, Birthdate, First Name, Last Name
8 REM SCHOOL-EXTRACT.TXT - Relates School ID, School Name, Principal Name
9 REM STBL-EXTRACT.TXT - Lookup tables for Student Info.
10 REM STUD-BILG.TXT - Bi Lingual information for each student
11 REM STUD-EXTRACT.TXT - Details for this student
12 REM STUD-SPED.TXT - Special Ed details for each student
13 REM TEAC-DIVI.TXT - Teacher to Home Room and School ID
14 REM TEAC-EMPID.TXT - Relates Teacher Employee ID, School Teacher ID, and School ID
15 REM
16 REM Once each file is imported, the file name is changed to done-XXXXXXXX MM DD YYYY.txt
17 REM If there is an error during import, the file is renamed to error-XXXXXXXX MM DD YYYY.txt
18 REM (where MM is Month, DD is Day and YYYY is the year when this command file was run).
19 REM
20 REM These files are imported into tables named for each import file with '-IMPORT' endings
21 REM in the SQL database.
22 REM
23 REM After all available tables have been imported, SQL stored procedure ' ' is started
24 REM that will condition import data for proper display.
25 REM
26 REM A Log file is created in the same directory as the files to be imported.
27 REM
28 REM USAGE:
29 REM import_Teacher2sis_files.cmd <parameters...>
30 REM PARAMETERS:
31 REM <Server name of SQL database> <Admin Login> <Admin Password> <Path to import files>
32 REM
33 REM BLAT utility must be present in C:\WINNT\SYSTEM32 and installed using:
34 REM blat -install snocex01.seattleschools.org FTPUser@seattleschools.org
35 REM (BLAT.EXE can be found at \\Snocex01\admin\blat)
36 REM -----
37 REM REVISION HISTORY:
38 REM 2-14-2002 Daniel J. Moore
39 REM INITIAL CREATION.
40 REM
41 REM 3-21-2002 Daniel J. Moore
42 REM Commented out processing of CENT-EXTRACT.TXT (not used)
43 REM
44 REM =====
45
46 REM E-MAIL TO...
47 SET recip=djmoore@seattleschools.org,gmlantz@seattleschools.org,dwlow@seattleschools.org,bgrubel@seattleschools.org
48
49
50 REM *****
51 REM Assign paramers to variables
52
53 set SQL=%1
54 set AdminLogin=%2
55 set AdminPassword=%3
56 set PathFiles=%4
57 set DB=Teacher2SIS
58
59 REM *****
60 for /F "tokens=1-4 delims=/ " %i in ('date /t') do (
61 set DayOfWeek=%i
62 set Month=%j
63 set Day=%k
64 set Year=%l
65 set Date=%i %j/%k/%l )
66
67 REM *** Temp Directory and log variables (used only in the batch process) *****
68 set log=%PathFiles%\Import_Teacher-SIS_%Month%-_%Day%-_%Year%.log
69 REM set log=Import_Teacher-SIS_%Month%-_%Day%-_%Year%.log
70
71 REM *****
72 REM *** Print the Date and Time when the update process is beginning
73 REM *****
74
75 ECHO '>> %log%'
76 ECHO '*****IMPORT TO TEACHER-SIS STARTED***** >> %log%'
77 DATE /T >> %log%
78 TIME /T >> %log%
79 ECHO '***** >> %log%'
80 ECHO '>> %log%'
81
82 :doSTBL-EXTRACT
83
84 ECHO ***** Starting processes for STBL-EXTRACT.TXT >> %log%
85 if not exist %PathFiles%\STBL-EXTRACT.TXT ( ECHO %PathFiles%\STBL-EXTRACT.TXT doesn't exist, moving to next file... && goto doCENT-EXTRACT) >> %log%
86
87 ECHO Purging contents of [STBL-EXTRACT-IMPORT] in %DB% on %SQL%... >> %log%
88 OSQL /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "DELETE [STBL-EXTRACT-IMPORT]" >> %log%
89
90 ECHO BCPing the contents of STBL-EXTRACT.TXT into %DB% on %SQL%... >> %log%
91 BCP Teacher2SIS.[STBL-EXTRACT-IMPORT] in %PathFiles%\STBL-EXTRACT.TXT -t '|' -c -S %SQL% -U %AdminLogin% -P %AdminPassword% >> %log%
92
93 IF ERRORLEVEL == 1 ( echo Error occurred while bcp'ing in data... changing the name of %PathFiles%\STBL-EXTRACT.TXT to error-STBL-EXTRACT_%Month%-_%Day%-_%Year%.TXT and moving to next file && ren %PathFiles%\STBL-EXTRACT.TXT
94 error-STBL-EXTRACT_%Month%-_%Day%-_%Year%.TXT && goto doCENT-EXTRACT) >> %log%
95
96 if exist %PathFiles%\done-STBL-EXTRACT_%Month%-_%Day%-_%Year%.TXT (del %PathFiles%\done-STBL-EXTRACT_%Month%-_%Day%-_%Year%.TXT /Q)
97 ren %PathFiles%\STBL-EXTRACT.TXT done-STBL-EXTRACT_%Month%-_%Day%-_%Year%.TXT >> %log%
```

```
98 REM *****
99 :doCENT-EXTRACT
100 REM ECHO '>> %log%
101 REM ECHO ***** Starting processes for CENT-EXTRACT.TXT >> %log%
102 REM if not exist %PathFiles%\CENT-EXTRACT.TXT ( ECHO %PathFiles%\CENT-EXTRACT.TXT doesn't exist, moving to next file... && goto doSCHOOL-EXTRACT) >> %log%
103
104 REM ECHO Purging contents of [CENT-EXTRACT-IMPORT] in %DB% on %SQL%... >> %log%
105 REM OSQL /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "DELETE [CENT-EXTRACT-IMPORT]" >> %log%
106
107 REM ECHO BCPing the contents of CENT-EXTRACT.TXT into %DB% on %SQL%... >> %log%
108 REM BCP Teacher2SIS..[CENT-EXTRACT-IMPORT] in %PathFiles%\CENT-EXTRACT.TXT -t "|" -c -S %SQL% -U %AdminLogin% -P %AdminPassword% >> %log%
109
110 REM IF ERRORLEVEL == 1 ( echo Error occurred while bcp'ing in data... changing the name of %PathFiles%\CENT-EXTRACT.TXT to error-CENT-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT and moving to next file && ren %PathFiles%\CENT-EXTRACT.TXT
111 error-CENT-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT && goto doSCHOOL-EXTRACT) >> %log%
112
113 REM if exist %PathFiles%\done-CENT-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT (del %PathFiles%\done-CENT-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT /Q)
114 REM ren %PathFiles%\CENT-EXTRACT.TXT done-CENT-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT >> %log%
115
116 REM *****
117 :doSCHOOL-EXTRACT
118 ECHO '>> %log%
119 ECHO ***** Starting processes for SCHOOL-EXTRACT.TXT >> %log%
120 if not exist %PathFiles%\SCHOOL-EXTRACT.TXT ( ECHO %PathFiles%\SCHOOL-EXTRACT.TXT doesn't exist, moving to next file... && goto doSTUD-BILG) >> %log%
121
122 ECHO Purging contents of [SCHOOL-EXTRACT-IMPORT] in %DB% on %SQL%... >> %log%
123 OSQL /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "DELETE [SCHOOL-EXTRACT-IMPORT]" >> %log%
124
125 ECHO BCPing the contents of SCHOOL-EXTRACT.TXT into %DB% on %SQL%... >> %log%
126 BCP Teacher2SIS..[SCHOOL-EXTRACT-IMPORT] in %PathFiles%\SCHOOL-EXTRACT.TXT -t "|" -c -S %SQL% -U %AdminLogin% -P %AdminPassword% >> %log%
127
128 IF ERRORLEVEL == 1 ( echo Error occurred while bcp'ing in data... changing the name of %PathFiles%\SCHOOL-EXTRACT.TXT to error-SCHOOL-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT and moving to next file && ren %PathFiles%\SCHOOL-EXTRACT.TXT
129 error-SCHOOL-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT && goto doSTUD-BILG) >> %log%
130
131 if exist %PathFiles%\done-SCHOOL-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT (del %PathFiles%\done-SCHOOL-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT /Q)
132 ren %PathFiles%\SCHOOL-EXTRACT.TXT done-SCHOOL-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT >> %log%
133
134 REM *****
135 :doSTUD-BILG
136 ECHO '>> %log%
137 ECHO ***** Starting processes for STUD-BILG.TXT >> %log%
138 if not exist %PathFiles%\STUD-BILG.TXT ( ECHO %PathFiles%\STUD-BILG.TXT doesn't exist, moving to next file... && goto doSTUD-EXTRACT) >> %log%
139
140 ECHO Purging contents of [STUD-BILG-IMPORT] in %DB% on %SQL%... >> %log%
141 OSQL /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "DELETE [STUD-BILG-IMPORT]" >> %log%
142
143 ECHO BCPing the contents of STUD-BILG.TXT into %DB% on %SQL%... >> %log%
144 BCP Teacher2SIS..[STUD-BILG-IMPORT] in %PathFiles%\STUD-BILG.TXT -t "|" -c -S %SQL% -U %AdminLogin% -P %AdminPassword% >> %log%
145
146 IF ERRORLEVEL == 1 ( echo Error occurred while bcp'ing in data... changing the name of %PathFiles%\STUD-BILG.TXT to error-STUD-BILG_%%Month%%-%%Day%%-%%Year%%.TXT and moving to next file && ren %PathFiles%\STUD-BILG.TXT
147 error-STUD-BILG_%%Month%%-%%Day%%-%%Year%%.TXT && goto doSTUD-EXTRACT) >> %log%
148
149 if exist %PathFiles%\done-STUD-BILG_%%Month%%-%%Day%%-%%Year%%.TXT (del %PathFiles%\done-STUD-BILG_%%Month%%-%%Day%%-%%Year%%.TXT /Q)
150 ren %PathFiles%\STUD-BILG.TXT done-STUD-BILG_%%Month%%-%%Day%%-%%Year%%.TXT >> %log%
151
152 REM *****
153 :doSTUD-EXTRACT
154 ECHO '>> %log%
155 ECHO ***** Starting processes for STUD-EXTRACT.TXT >> %log%
156 if not exist %PathFiles%\STUD-EXTRACT.TXT ( ECHO %PathFiles%\STUD-EXTRACT.TXT doesn't exist, moving to next file... && goto doSTUD-SPED) >> %log%
157
158 ECHO Purging contents of [STUD-EXTRACT-IMPORT] in %DB% on %SQL%... >> %log%
159 OSQL /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "DELETE [STUD-EXTRACT-IMPORT]" >> %log%
160
161 ECHO BCPing the contents of STUD-EXTRACT.TXT into %DB% on %SQL%... >> %log%
162 BCP Teacher2SIS..[STUD-EXTRACT-IMPORT] in %PathFiles%\STUD-EXTRACT.TXT -t "|" -c -S %SQL% -U %AdminLogin% -P %AdminPassword% >> %log%
163
164 IF ERRORLEVEL == 1 ( echo Error occurred while bcp'ing in data... changing the name of %PathFiles%\STUD-EXTRACT.TXT to error-STUD-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT and moving to next file && ren %PathFiles%\STUD-EXTRACT.TXT
165 error-STUD-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT && goto doSTUD-SPED) >> %log%
166
167 if exist %PathFiles%\done-STUD-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT (del %PathFiles%\done-STUD-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT /Q)
168 ren %PathFiles%\STUD-EXTRACT.TXT done-STUD-EXTRACT_%%Month%%-%%Day%%-%%Year%%.TXT >> %log%
169
170 REM *****
171 :doSTUD-SPED
172 ECHO '>> %log%
173 ECHO ***** Starting processes for STUD-SPED.TXT >> %log%
174 if not exist %PathFiles%\STUD-SPED.TXT ( ECHO %PathFiles%\STUD-SPED.TXT doesn't exist, moving to next file... && goto doTEAC-DIVI) >> %log%
175
176 ECHO Purging contents of [STUD-SPED-IMPORT] in %DB% on %SQL%... >> %log%
177 OSQL /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "DELETE [STUD-SPED-IMPORT]" >> %log%
178
179 ECHO BCPing the contents of STUD-SPED.TXT into %DB% on %SQL%... >> %log%
180 BCP Teacher2SIS..[STUD-SPED-IMPORT] in %PathFiles%\STUD-SPED.TXT -t "|" -c -S %SQL% -U %AdminLogin% -P %AdminPassword% >> %log%
181
182 IF ERRORLEVEL == 1 ( echo Error occurred while bcp'ing in data... changing the name of %PathFiles%\STUD-SPED.TXT to error-STUD-SPED_%%Month%%-%%Day%%-%%Year%%.TXT and moving to next file && ren %PathFiles%\STUD-SPED.TXT
183 error-STUD-SPED_%%Month%%-%%Day%%-%%Year%%.TXT && goto doTEAC-DIVI) >> %log%
184
185 if exist %PathFiles%\done-STUD-SPED_%%Month%%-%%Day%%-%%Year%%.TXT (del %PathFiles%\done-STUD-SPED_%%Month%%-%%Day%%-%%Year%%.TXT /Q)
186 ren %PathFiles%\STUD-SPED.TXT done-STUD-SPED_%%Month%%-%%Day%%-%%Year%%.TXT >> %log%
187
188 REM *****
189 :doTEAC-DIVI
190 ECHO '>> %log%
191 ECHO ***** Starting processes for TEAC-DIVI.TXT >> %log%
192 if not exist %PathFiles%\TEAC-DIVI.TXT ( ECHO %PathFiles%\TEAC-DIVI.TXT doesn't exist, moving to next file... && goto doTEAC-EMPID) >> %log%
193
194 ECHO Purging contents of [TEAC-DIVI-IMPORT] in %DB% on %SQL%... >> %log%
195 OSQL /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "DELETE [TEAC-DIVI-IMPORT]" >> %log%
```

```
191
192 ECHO BCPing the contents of TEAC-DIVI.TXT into %DB% on %SQL%... >> %log%
193 BCP Teacher2SIS..[TEAC-DIVI-IMPORT] in %PathFiles%\TEAC-DIVI.TXT -t "|" -c -S %SQL% -U %AdminLogin% -P %AdminPassword% >> %log%
194
195 IF ERRORLEVEL == 1 ( echo Error occurred while bcp'ing in data... changing the name of %PathFiles%\TEAC-DIVI.TXT to error-TEAC-DIVI_%%Month%%-%%Day%%-%%Year%%.TXT and moving to next file && ren %PathFiles%\TEAC-DIVI.TXT
error-TEAC-DIVI_%%Month%%-%%Day%%-%%Year%%.TXT && goto doTEAC-EMPID) >> %log%
196
197 if exist %PathFiles%\done-TEAC-DIVI_%%Month%%-%%Day%%-%%Year%%.TXT (del %PathFiles%\done-TEAC-DIVI_%%Month%%-%%Day%%-%%Year%%.TXT /Q)
198 ren %PathFiles%\TEAC-DIVI.TXT done-TEAC-DIVI_%%Month%%-%%Day%%-%%Year%%.TXT >> %log%
199
200 REM *****
201 :doTEAC-EMPID
202 ECHO '>> %log%
203 ECHO ***** Starting processes for TEAC-EMPID.TXT >> %log%
204 if not exist %PathFiles%\TEAC-EMPID.TXT ( ECHO %PathFiles%\TEAC-EMPID.TXT doesn't exist, starting DB processing of imported tables... && goto updateDB) >> %log%
205
206 ECHO Purging contents of [TEAC-EMPID-IMPORT] in %DB% on %SQL%... >> %log%
207 OSQ /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "DELETE [TEAC-EMPID-IMPORT]" >> %log%
208
209 ECHO BCPing the contents of TEAC-EMPID.TXT into %DB% on %SQL%... >> %log%
210 BCP Teacher2SIS..[TEAC-EMPID-IMPORT] in %PathFiles%\TEAC-EMPID.TXT -t "|" -c -S %SQL% -U %AdminLogin% -P %AdminPassword% >> %log%
211
212 IF ERRORLEVEL == 1 ( echo Error occurred while bcp'ing in data... changing the name of %PathFiles%\TEAC-EMPID.TXT to error-TEAC-EMPID_%%Month%%-%%Day%%-%%Year%%.TXT and moving to next file && ren %PathFiles%\TEAC-EMPID.TXT
error-TEAC-EMPID_%%Month%%-%%Day%%-%%Year%%.TXT && goto AllDone) >> %log%
213
214 if exist %PathFiles%\done-TEAC-EMPID_%%Month%%-%%Day%%-%%Year%%.TXT (del %PathFiles%\done-TEAC-EMPID_%%Month%%-%%Day%%-%%Year%%.TXT /Q)
215 ren %PathFiles%\TEAC-EMPID.TXT done-TEAC-EMPID_%%Month%%-%%Day%%-%%Year%%.TXT >> %log%
216
217 REM *****
218 REM *****
219
220 :updateDB
221 ECHO '>> %log%
222 ECHO '>> %log%
223 ECHO '>> %log%
224 ECHO ***** Starting DB Processing in Teacher2SIS >> %log%
225 OSQ /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "sp_updateAll" >> %log%
226
227 IF ERRORLEVEL == 1 ( echo Error occurred in DB Processing in Teacher2SIS... && goto Endit) >> %log%
228
229 REM *****
230 REM *****
231
232 :updateReports
233 ECHO '>> %log%
234 ECHO '>> %log%
235 ECHO '>> %log%
236 ECHO ***** Starting DB Copying to in Teacher2SISReports >> %log%
237
238 OSQ /U %AdminLogin% /P %AdminPassword% /S %SQL% /d %DB% /Q "sp_upReportData" >> %log%
239
240
241 IF ERRORLEVEL == 1 ( echo Error occurred in DB Processing in Teacher2SISReports... && goto Endit) >> %log%
242
243
244 REM *****
245 REM *****
246
247 goto AllDone
248
249 :AllDone
250 ECHO '>> %log%
251
252 ECHO '>> %log%
253
254 ECHO '>> %log%
255
256 ECHO *****IMPORT TO TEACHER-SIS PROCESS STOPPED***** >> %log%
257 DATE /T >> %log%
258 TIME /T >> %log%
259 ECHO *****END***** >> %log%
260
261 ECHO '>> %log%
262 ECHO '>> %log%
263 ECHO Batch process completed successfully!! >> %log%
264 blat %log% -t %recips% -s "Successful completion of import_Teacher2sis_files.cmd!!"
265 Goto End
266
267 :Endit
268 ECHO '>> %log%
269 ECHO *****IMPORT TO TEACHER-SIS PROCESS STOPPED***** >> %log%
270 DATE /T >> %log%
271 TIME /T >> %log%
272 ECHO *****END***** >> %log%
273
274 ECHO '>> %log%
275 ECHO '>> %log%
276 ECHO An Error has occurred. This batch process has been terminated. >> %log%
277 blat %log% -t %recips% -s "There was an error running import_Teacher2sis_files.cmd ... Errors Occured!!!"
278
279 :End
280
281
```

```
1 CREATE PROCEDURE sp_purgeAll AS
2
3 EXEC sp_purgeAllImports
4
5 EXEC sp_purgeAllDisplays
6 GO
7
8
```

```
9 CREATE PROCEDURE sp_purgeAllDisplays AS
10
11 DELETE [CENT-EXTRACT]
12
13 DELETE [SCHOOL-EXTRACT]
14
15 DELETE [STBL-EXTRACT]
16
17 DELETE [STUD-BILG]
18
19 DELETE [STUD-EXTRACT]
20
21 DELETE [STUD-SEARCH]
22
23 DELETE [STUD-SPED]
24
25 DELETE [TEAC-DIVI]
26
27 DELETE [TEAC-EMPID]
28 GO
29
30
```

```
31 CREATE PROCEDURE sp_purgeAllImports AS
32
33 DELETE [CENT-EXTRACT-IMPORT]
34
35 DELETE [SCHOOL-EXTRACT-IMPORT]
36
37 DELETE [STBL-EXTRACT-IMPORT]
38
39 DELETE [STUD-BILG-IMPORT]
40
41 DELETE [STUD-EXTRACT-IMPORT]
42
43 DELETE [STUD-SPED-IMPORT]
44
45 DELETE [TEAC-DIVI-IMPORT]
46
47 DELETE [TEAC-EMPID-IMPORT]
48 GO
49
50
```



```
51 CREATE PROCEDURE sp_purgeStudentDataOnly AS
52
53 DELETE [STUD-BILG]
54
55 DELETE [STUD-EXTRACT]
56
57 DELETE [STUD-SEARCH]
58
59 DELETE [STUD-SPED]
60 GO
61
62
```

```
63 CREATE PROCEDURE sp_updateAll AS
64
65 print "starting processing of imported data..."
66
67 /* STUD-EXTRACT */
68 IF EXISTS (SELECT * FROM [STUD-EXTRACT-IMPORT] )
69 BEGIN
70 print "PROCESSING STUD-EXTRACT-IMPORT"
71 DELETE FROM [STUD-EXTRACT]
72 WHERE [STUDENT_ID] IN
73     (SELECT [STUDENT_ID]
74      FROM [STUD-EXTRACT-IMPORT])
75 INSERT INTO [STUD-EXTRACT]
76     SELECT *
77     FROM [STUD-EXTRACT-IMPORT]
78 DELETE [STUD-EXTRACT-IMPORT]
79 DELETE [STUD-SEARCH]
80 INSERT INTO [STUD-SEARCH] ( SCHOOL_ID, STUDENT_ID, LASTNAME, FIRSTNAME, MIDDLENAME, BIRTH, SEX, HR, GRADE, STUD_STATUS, ACCESS_SCHOOL)
81 SELECT [STUD-EXTRACT].SCHOOL_ID, [STUD-EXTRACT].STUDENT_ID, [STUD-EXTRACT].LASTNAME, [STUD-EXTRACT].FIRSTNAME, [STUD-EXTRACT].MIDDLENAME, [STUD-EXTRACT].BIRTH, [STUD-EXTRACT].SEX, [STUD-EXTRACT].HR, [STUD-EXTRACT].GRADE,
82 [STUD-EXTRACT].STUD_STATUS, [STUD-EXTRACT].ACCESS_SCHOOL
83 FROM [STUD-EXTRACT]
84 ORDER BY [STUD-EXTRACT].SCHOOL_ID, [STUD-EXTRACT].HR, [STUD-EXTRACT].LASTNAME, [STUD-EXTRACT].FIRSTNAME;
85 END
86
87 /* STUD-SPED */
88 IF EXISTS (SELECT * FROM [STUD-SPED-IMPORT] )
89 BEGIN
90 print "PROCESSING STUD-SPED-IMPORT"
91 DELETE FROM [STUD-SPED]
92 WHERE [STUDENT_ID] IN
93     (SELECT [STUDENT_ID]
94      FROM [STUD-SPED-IMPORT])
95 INSERT INTO [STUD-SPED]
96     SELECT *
97     FROM [STUD-SPED-IMPORT]
98 DELETE [STUD-SPED-IMPORT]
99 END
100
101
102 /* CENT-EXTRACT */
103 IF EXISTS (SELECT * FROM [CENT-EXTRACT-IMPORT] )
104 BEGIN
105 print "PROCESSING CENT-EXTRACT-IMPORT"
106 DELETE FROM [CENT-EXTRACT]
107 WHERE [STUDENT_ID] IN
108     (SELECT [STUDENT_ID]
109      FROM [CENT-EXTRACT])
110
111 INSERT INTO [CENT-EXTRACT]
112     SELECT *
113     FROM [CENT-EXTRACT-IMPORT]
114
115 DELETE [CENT-EXTRACT-IMPORT]
116 END
117
118 /* STUD-BILG */
119 IF EXISTS (SELECT * FROM [STUD-BILG-IMPORT] )
120 BEGIN
121 print "PROCESSING STUD-BILG-IMPORT"
122 DELETE FROM [STUD-BILG]
123 WHERE [STUDENT_ID] IN
124     (SELECT [STUDENT_ID]
125      FROM [STUD-BILG-IMPORT])
126
127 INSERT INTO [STUD-BILG]
128     SELECT *
129     FROM [STUD-BILG-IMPORT]
130
131 DELETE [STUD-BILG-IMPORT]
132 END
133
134 /* TEAC-DIVI */
135 IF EXISTS (SELECT * FROM [TEAC-DIVI-IMPORT] )
136 BEGIN
137 print "PROCESSING TEAC-DIVI-IMPORT"
138 DELETE [TEAC-DIVI]
139
140 INSERT INTO [TEAC-DIVI]
141     SELECT *
142     FROM [TEAC-DIVI-IMPORT]
143
144 DELETE [TEAC-DIVI-IMPORT]
145 END
146
147 /* TEAC-EMPID */
148 IF EXISTS (SELECT * FROM [TEAC-EMPID-IMPORT] )
149 BEGIN
150 print "PROCESSING TEAC-EMPID-IMPORT"
151 DELETE [TEAC-EMPID]
152
153 INSERT INTO [TEAC-EMPID]
154     SELECT *
155     FROM [TEAC-EMPID-IMPORT]
156
157 /*Adding test data */
158 INSERT INTO [TEAC-EMPID] ([SCHOOL_ID], [TEAC_ID], [EMP_ID]) VALUES ('0201', 'TES1', '111111111')
159 INSERT INTO [TEAC-EMPID] ([SCHOOL_ID], [TEAC_ID], [EMP_ID]) VALUES ('0011', 'TES2', '111111111')
```

```
160 INSERT INTO [TEAC-EMPID] ([SCHOOL_ID], [TEAC_ID], [EMP_ID]) VALUES ('0799', 'TES3', '123456789')
161 INSERT INTO [TEAC-EMPID] ([SCHOOL_ID], [TEAC_ID], [EMP_ID]) VALUES ('0201', 'TES4', '019105')
162 INSERT INTO [TEAC-EMPID] ([SCHOOL_ID], [TEAC_ID], [EMP_ID]) VALUES ('0011', 'TES5', '019105')
163 INSERT INTO [TEAC-EMPID] ([SCHOOL_ID], [TEAC_ID], [EMP_ID]) VALUES ('0117', 'TES6', '019105')
164
165 DELETE [TEAC-EMPID-IMPORT]
166 END
167
168 /* SCHOOL-EXTRACT */
169 IF EXISTS (SELECT * FROM [SCHOOL-EXTRACT-IMPORT] )
170 BEGIN
171 print "PROCESSING SCHOOL-EXTRACT-IMPORT"
172 DELETE [SCHOOL-EXTRACT]
173
174 INSERT INTO [SCHOOL-EXTRACT]
175     SELECT *
176     FROM [SCHOOL-EXTRACT-IMPORT] ORDER BY [SCHOOL_NAME]
177
178 DELETE [SCHOOL-EXTRACT-IMPORT]
179 END
180
181 /* STBL-EXTRACT */
182 IF EXISTS (SELECT * FROM [STBL-EXTRACT-IMPORT] )
183 BEGIN
184 print "PROCESSING STBL-EXTRACT-IMPORT"
185 DELETE [STBL-EXTRACT]
186
187 INSERT INTO [STBL-EXTRACT]
188     SELECT *
189     FROM [STBL-EXTRACT-IMPORT]
190
191 DELETE [STBL-EXTRACT-IMPORT]
192 END
193 GO
194
195
```

```
196 CREATE PROCEDURE sp_upReportData AS
197
198 print "starting processing of updating the reports database..."
199
200
201 /* STUD-EXTRACT */
202 print "copying STUD-EXTRACT to Reports Database Teacher2SISReports"
203
204 DELETE FROM Teacher2SISReports.dbo.[STUD-EXTRACT]
205
206 INSERT INTO Teacher2SISReports.dbo.[STUD-EXTRACT]
207     SELECT *
208     FROM [STUD-EXTRACT]
209
210
211 /* SCHOOL-EXTRACT */
212 print "copying SCHOOL-EXTRACT to Reports Database Teacher2SISReports"
213
214 DELETE FROM Teacher2SISReports.dbo.[SCHOOL-EXTRACT]
215
216 INSERT INTO Teacher2SISReports.dbo.[SCHOOL-EXTRACT]
217     SELECT *
218     FROM [SCHOOL-EXTRACT]
219
220
221 /* STUD-BILG */
222 print "copying STUD-BILG to Reports Database Teacher2SISReports"
223
224 DELETE FROM Teacher2SISReports.dbo.[STUD-BILG]
225
226 INSERT INTO Teacher2SISReports.dbo.[STUD-BILG]
227     SELECT *
228     FROM [STUD-BILG]
229
230
231 /* STBL-EXTRACT */
232 print "copying STBL-EXTRACT to Reports Database Teacher2SISReports "
233
234 DELETE FROM Teacher2SISReports.dbo.[STBL-EXTRACT]
235
236 INSERT INTO Teacher2SISReports.dbo.[STBL-EXTRACT]
237     SELECT *
238     FROM [STBL-EXTRACT]
239
240
241
242 /* STUD-SPED */
243 print "copying STUD-SPED to Reports Database Teacher2SISReports"
244
245 DELETE FROM Teacher2SISReports.dbo.[STUD-SPED]
246
247 INSERT INTO Teacher2SISReports.dbo.[STUD-SPED]
248     SELECT *
249     FROM [STUD-SPED]
250
251
252 /* TEAC-DIVI */
253 print "copying TEAC-DIVI to Reports Database Teacher2SISReports"
254
255 DELETE FROM Teacher2SISReports.dbo.[TEAC-DIVI]
256
257 INSERT INTO Teacher2SISReports.dbo.[TEAC-DIVI]
258     SELECT *
259     FROM [TEAC-DIVI]
260
261
262 /* TEAC-EMPID */
263 print "copying TEAC-EMPID to Reports Database Teacher2SISReports"
264
265 DELETE FROM Teacher2SISReports.dbo.[TEAC-EMPID]
266
267 INSERT INTO Teacher2SISReports.dbo.[TEAC-EMPID]
268     SELECT *
269     FROM [TEAC-EMPID]
270
271 GO
```