

Degree Plan Builder

By

Jeffery R Black

Submitted to

The faculty of the Information Engineering Technology Program
In Partial Fulfillment of the Requirements for
The Degree of Bachelor of Science
In Information Engineering Technology

University of Cincinnati
College of Applied Science

February 2001

Degree Plan Builder

By

Jeffery R Black

Submitted to
The faculty of the Information Engineering Technology Program
In Partial Fulfillment of the Requirements for
The Degree of Bachelor of Science
In Information Engineering Technology

© Copyright 2001 Jeffery R Black

The author grants to the Information Engineering Technology Program permission to reproduce and distribute copies of this document in whole or in part.

Author: Jeffery Black

Date

Faculty Project Advisor

Date

Department Head

Date

Acknowledgements

I would like to thank all my friends in the lab for their help in the creation of this project. Thank you to all the teachers that took time in their busy schedule to answer my many questions. And I want to give a huge thank you to my family that showed great patience and support throughout the duration of the project.

Table of Contents

Section	Page
Acknowledgements	i
Table of Contents	ii
List of Figures	iv
Abstract	v
1. Introduction to the Problem	1
1.1 Background	1
1.2 The Problem	1
2. Literature Review	2
3. Description of the Degree Plan Builder	3
3.1 Project Overview	3
3.2 User Profile	5
3.3 Design Protocols	5
3.4 Development Tools	7
4. Objectives of the Project	8
5. Design and Development	8
5.1 Timeline	8
5.2 Budget	10
6. Proof of Design	11
6.1 Student Web Front-end	11
6.1.1 Screen 1	11
6.1.2 Screen 2	12
6.1.3 Screen 3	13
6.1.4 Screen 4	13
6.1.5 Screen 5	14
6.1.6 Screen 6	15
6.1.7 Screen 7	15
6.1.8 Screen 8	16
6.1.9 Screen 9	17
6.2 Teacher Web Front-end	17
6.2.1 Screen 1	18
6.2.2 Screen 2	18
6.2.3 Screen 3	19
6.3 Administration Access Front-end	20
6.3.1 Screen 1	20

6.3.2 Screen 2	20
6.3.3 Screen 3	20
6.3.4 Screen 4	21
6.3.5 Screen 5	22
6.3.6 Screen 6	22
7. Conclusion and Recommendation	23
Appendix A	26
A.1 Graphical Representation of the SQL database	26
A.2 SQL Table Scripts	26
A.3 SQL View Scripts	28
A.4 SQL Stored Procedure Scripts	29
Appendix B	32
B.1 Student Web Front-End Code	32
B.1.1 Login.htm	32
B.1.2 Receivelogin.asp	33
B.1.3 Lookatinfo.asp	36
B.1.4 Enterinfo.asp	38
B.1.5 Updateinfo.asp	42
B.1.6 Viewclass.asp	45
B.1.7 Developdp.asp	48
B.1.8 Prereq.asp	60
B.1.9 Enterclass.asp	61
B.1.10 Getquarter.asp	64
B.1.11 Editdb.asp	66
B.1.12 Updateclass.asp	73
B.1.13 Viewcoop.asp	77
B.1.14 Degreeplan.asp	79
B.2 Teacher Web Front-End Code	81
B.2.1 Receiveteacherlogin.asp	81
B.2.2 Viewstudentclass.asp	83
B.2.3 Teacher.asp	86
Appendix C	89
C.1 Access Front-End Form Code	89
C.1.1 FrmAddCoopQuarter	89
C.1.2 FrmAllCoop	90
C.1.3 FrmClass	90
C.1.4 FrmCompany	92
C.1.5 FrmCoop	93
C.1.6 FrmMain	94
C.1.7 FrmStudent	97
References	99

List of Figures

Figure 1 Table of Hardware Prices	10
Figure 2 Table of Software Prices	11
Figure 3 Student Login Screen	11
Figure 4 Student Homepage	12
Figure 5 Updating Information	13
Figure 6 Viewing Degree Plan	14
Figure 7 Adding Classes	15
Figure 8 Editing Degree Plan	16
Figure 9 Available Jobs	17
Figure 10 Teacher's Login Screen	18
Figure 11 Choosing a Student	19
Figure 12 Student's Degree Plan	19
Figure 13 Class Information	20
Figure 14 Add/Remove Student	21
Figure 15 View Degree Plan	21
Figure 16 Company Information	22
Figure 17 Coop Form	22
Figure 18 Table Relationships	25

Abstract

This application is designed to allow the students of the College of Applied Science to build and maintain their degree plan. A degree plan is a schedule of a student's classes for his/her college career. The Web application is a database driven Web site that allows the student to view his/her degree plan, add and remove classes, or view jobs that are available through the Professional Practice and Career Placement office. The student advisors can use the Web application to approve or disapprove degree plans. The project also includes a Microsoft Access program for administration purposes. The Access program allows administrators to maintain student records and update classes and cooperative education positions from the database.

Degree Plan Builder

1. Introduction to the Problem

The Professional Practice and Career Placement (PPCP) Office at CAS has a Web page and a DOS program to access and edit a student's degree plan. The student uses these programs to plan their future coursework in their major. The student then prints out the degree plan and has it signed by his/her student advisor.

1.1 Background

The student's degree plan is a schedule of the student's classes for his/her career in college. The degree plan is divided into academic quarters. Each quarter has the class number, class title, the number of credit hours, and grade for each of the classes the student is taking for the quarter. If the student is on co-op, he/she would input co-op in the class title spot. After the student completes the quarter, the student has to use the program to edit his/her degree plan and fill in any class changes and input the grades for the quarter.

The students have two ways of accessing their degree plan. Through the PPCP's Web page or using a DOS program. Academic Software, Inc developed both methods of accessing the degree plan.

1.2 The Problem

The user interface for the old program has many problems in how it presents the data to the user. The field to input the class number is too small and only allowing seven characters. The UC class number has either eight or nine characters. The class name field does not allow for the full class name so the students has to abbreviate the class

name. The grades field only allows for one character; since UC uses +/- on its grading system so the grades field needs to allow input for two characters.

In addition, the DOS program is a 16-bit program, which will not run on the newer Pentium II (PII), and Pentium III (PIII) workstations. The Central Computer Lab on the fourth floor of the Administration building will soon be updated to have all PII and PIII-based workstations. Once the upgrade is complete, the DOS program will not run on any of the computers in this lab.

The DOS program interface is also cumbersome and discourages students from using it as a way to access their degree plans. The interface does not have cut and paste functionality. If a student changes sections and decides to take two quarters of classes or of co-op in a row, the student must delete his/her whole degree plan and re-create it. The program does not let the student know the requirements for each class: a student may schedule his/her college career with out realizing that a class may have a prerequisite.

The DOS program is stored on a 3½" floppy disk. 3½" floppy disks are not very reliable and can easily be destroyed. In the event that this happens, the degree plan is lost. The program does upload the degree plan to a database, but the program does not have the ability to download the degree plan to the disk. The student would have to buy a new disk and re-create the degree plan.

2. Literature Review

In doing research of literature on the web, I found that many of the major colleges and universities allow the student to download the form from a static Web page. Texas Tech follows this format. The student goes to one of Texas Tech's web page and downloads a Word file. The student prints the file and fills out the needed information.

He/she takes the form and has a college advisor signed the form before taking it to the office (1).

Many colleges like Texas A&M and Williams Baptist College do not have an online format for developing a degree plan. Students at both colleges must go to an office, obtain a degree plan form, then fill out the form and turn it into the office. This method may causes problems with students that work during the day or take classes online. Now these students have to make a trip to campus every time they need to change their degree plan (2).

The University of Cincinnati College of Applied Science uses an online degree plan system. This system allows the student to create and edit his/her degree plan online. After editing the degree plan, the student can print the form, have it sign and turn it in. After researching many universities I found that only CAS has an online degree plan system (3).

3. Description of The Degree Plan Builder

3.1 Project Overview

There are two user interfaces for this application. The student/teacher user interface is Web based and can be access via the Internet. The administration front-end was developed using Microsoft Access. Both interfaces access information stored in a SQL database.

The Web front end was created for both students and teachers. The students will use the web front-end to create, edit, and delete classes from their degree plan. The student logs into the application using a Student ID given to them by the PPCP office. The application will then ask the student to edit their personal information, add classes to

the degree plan, edit or delete existing classes in the degree plan, or look up co-op jobs. If the student decides on editing their personal information, the application brings up a screen with the current information, allows the student to add or change additional information, and prompts the user to enter or cancel the change. If the user decides to add classes, the program gives a list of drop-down boxes. The user can then pick the classes he/she wants to add and choose to enter the classes or cancel. The editing or deleting existing classes allows the user to pick a quarter in the degree plan and either updates the information (example: changing grades, moving it to another quarter) or deletes the class from the degree plan. The last option for the user is to look at available jobs. Going to this part of the Web site shows the student the jobs available based on his/her major.

The teacher portion of the Web site allows the teacher to view and approve or disapprove the degree plans of his/her students. The PPCP office gives the teacher a user name and password. The teacher can login to the Web site and receives a list of students the teacher advises in the development of their degree plan. The teacher chooses the name of the student and the Web site brings up that student's degree plan. The teacher is then prompted to approve or disapprove the degree plan.

The Microsoft Access front-end was developed so the administration can add, delete, and update student, class, and co-op Company information. The first screen asks the user to choose from editing student information, editing class information, editing co-op company information, adding student's co-op information, or looking at statistics for pay rates for students. Editing student, class, or co-op company information will allow the administrator add, remove, or update any of the current information. If the user

selects student's co-op information, the application brings up a window and prompts for a student number. After choosing a student number, the window shows the student's co-op information. This window also gives the user the ability to add a new co-op for this student. The statistics for pay rates of students is a set of reports that will give statistics on what the co-op companies are paying the students each quarter.

3.2 User Profile

The target user for the degree plan program is the student body at CAS. The target user will vary the amount of knowledge they have about browsing the Internet. Some users like Information Engineering Technology students are expected to be knowledgeable about browsing the Internet and should be able to use the Web application easily. Students in other fields that do not involve computers as much may have a short learning curve depending on their personal experience. With the Internet becoming more mainstream in society, most students should have enough experience browsing the Internet and will be able to use this Web site.

3.3 Design Protocols

I used the three following criteria to create the Web front-end for the user:

- Easily available to use
- Short learning curve to use
- A user friendly interface

To make the application easily available for the students to use, I decided to make it a Web application. Having the project on the Internet allows the student to have access to the application 24 hours a day, 7 days a week. The student also does not have to worry

about carrying around a floppy disk or any other type of storage media. All information the student needs and stores will be placed in a database accessed by the Web application.

The Web site was design to allow for a short learning curve for the students. Instead of having a very complicated Web application, the project has a simple interface for the student to add, update and drop classes. Most of the information the student will input will come from drop-down boxes, check boxes, and buttons. The user does not have to worry about going through complicated menus, typing in class information, or trying to figure out how to cut and paste from one section to another.

The Web front-end meets the criteria of having a friendly user interface by minimizing the amount of input from the user. The Web application does not need the user to know everything about the college course. By selecting a class name, the Web site will show the user the course number, the number of credit hours for the course and any prerequisites the course may have. The Web site also offers a help page that will give step-by-step directions on how to add, delete, and update classes on the degree plan. The help page loads in a separate Web browser so that the student can switch between the two browsers and follow the directions. The Web site also has a sample degree plan which is complete to give new students an idea of what a degree plan looks like.

The set up of the user interface also helps the application meet the criteria of having a friendly user interface. All the Web pages on the site have the same color scheme and fonts. Along with the same color scheme, the user will find the same navigation bar at the top and bottom of each page. The navigation bar will give the user the option to go to any of the main parts of the application without having to first go back

to the home page of the site. The color scheme and navigation bars will also stay the same from page to page to assure the user that he/she is still within the Web application.

3.4 Development Tools

The Web front-end was developed in Hyper Text Markup Language (HTML) and Active Server Pages (ASP) on a Microsoft (MS) Windows NT 4.0 Server (Win NT) running Internet Information Server (IIS). The user data is stored in a database on Microsoft SQL Server.

ASPs are scripts that run on IIS that create new Web pages each time a user request the Web page. This Web site gathers the request information for the SQL server and presents the information in HTML format to the user. This method was used for the Web application so the user can request his/her class information, view or change the data, and save the data in the SQL database.

I used Microsoft Visual Interdev to develop the ASP scripts and HTML files. I decided to use MS Visual Interdev because it was free through UC and I had a college course on MS Visual Interdev.

I used Paint Shop Pro to create any of the graphics used through out the Web site. Paint Shop Pro is a graphics program proved to the students for free in the fourth floor computer labs. The reasons for using Paint Shop Pro is I have four years experience using Paint Shop Pro for 4 year and Paint Shop Pro is free to use in the computer labs on the fourth floor of the Administration Building.

The access front-end was developed in Microsoft Access. I chose Access as the administrator front-end since the software was very cheap, I have been using it for 3 years and feel comfortable with developing in it, and Access is part of the Microsoft

Office Suite. MS Office is standard software in the office environment so I have less to worry about when considering if it will be compatible with the user's computer setup.

4. Objectives of the Project

Here are the objectives that were set for this project:

1. Allow logon with a student id number
2. The web page will help to answer basic questions about what is a degree plan.
3. The web page will be designed to show samples of a degree plan
4. The web page front-end will give students the ability to add to, update and view their degree plan.
5. The web-based front-end will give students the ability to check prerequisites for classes.
6. The Access front-end will be designed to allow administrators to add, remove and update co-op Companies, classes, and students
7. The web-based front-end will be design to show available jobs per student's major.
8. The Web front-end will be designed to allow teachers to look at and approve degree plans of their students.
9. The Access front-end will allow administrators to keep track of a student's progress with both classes and co-op quarters.
10. The Access front-end will give administrators statistics of the pay rates per major and year of students (Freshmen, sophomore, etc.)

5. Design and Development

5.1 Timeline

Fall Quarter 2000

September 4, 11

Research on using Access 2000 as a front end for SQL databases

Continued Research on ASP's and web page design

September 18,25

Continue research on Access 2000

Begin Creating Administration front end in Access 2000

October 9

Complete research on Access 2000

Finish the add, update and view classes and students functionality for the Admin

Front-end

Begin Reports for Statistic Uses.

Begin co-op Input Screen

October 16, 23

Finish Reports and co-op Input Screen

Begin Help Pages for Access Front-End

November 6,1

Begin Error Checking with Front-End

Begin Beta Testing of Access Front-End

Finish Help Pages

November 20, 27

Fix known errors

Beta-testing 2

December 4

Finish Access 2000 Front-end

Add Update Class information functionality to Web-based front-end

December 11, 18

Add Check Prerequisites functionality

Design theme to the web based front-end (color, navigation buttons, fonts, etc)

December 25

Apply theme to the web front-end

Winter Quarter 2000

January 1,8

Begin Beta-testing the Web-based front-end with OCAS students

January 15,22

Begin final documentation

Fix any bugs brought out through beta testing

January 29

Continue testing Web front-end

February 5,12

Continue debugging project

Go through programming code and clean it up(add comments etc.)

Finish first draft of final report

February 19, 26

Finish project and documentation

Prepare for final presentation

5.2 Budget

Hardware Requirements:

Web Server*	\$2,515.60
Database Server*	\$2,991.00
Subtotal of Hardware	\$5,506.60

Figure 1. Table of Hardware Prices

Software Requirements:

Visual InterDev 6.0**	\$889.95
MS SQL Server 7.0**	\$1,399.00
Internet Information Server 4.0	Comes with NT Server
MS Office 2000 Professional**	\$599.00
MS NT 4.0 Server**	\$809.00
Subtotal of Software	\$3,696.95
Total of Project	\$9,203.55

Figure 2. Table of Software Prices

*Prices taken from www.micronpc.com

**Prices taken from www.microsoft.com

OMI College of Applied Science Fourth Floor Computer Lab has the software and hardware resources to complete the project.

6. Proof of Design

6.1 Student Web Front-end

The following text will describe the sequence of steps a user would see on the Web-based front-end. All HTML and ASP code can be found in Appendix B.

6.1.1 Screen 1



Figure 3 Student Login Screen

The login.htm is the first page a user will see. This page welcomes the user to the page and gives a prompt for a Student password or a teacher user name and password. The page also has two links. The first link brings up a small window explaining to the user what the Degree Plan Builder is used for.

6.1.2 Screen 2

The receivelogin.asp is the page that verifies the user's student number. If the student enters a correct number, he/she will be given a set of links to navigate the Web site. If the student number is wrong, the application will prompt the user to go back to the login screen and try another number.

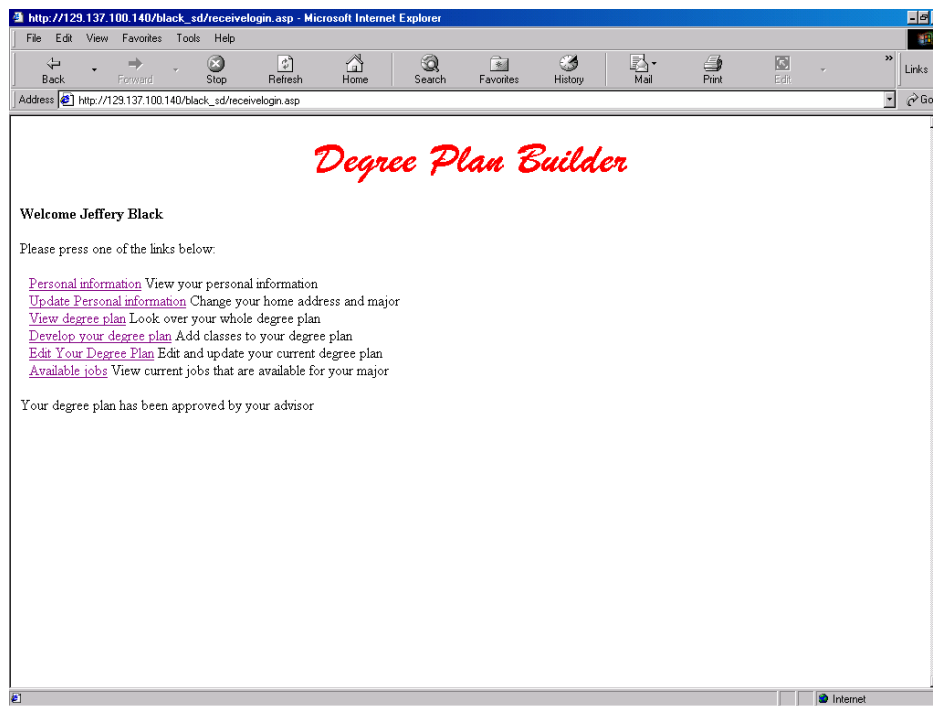


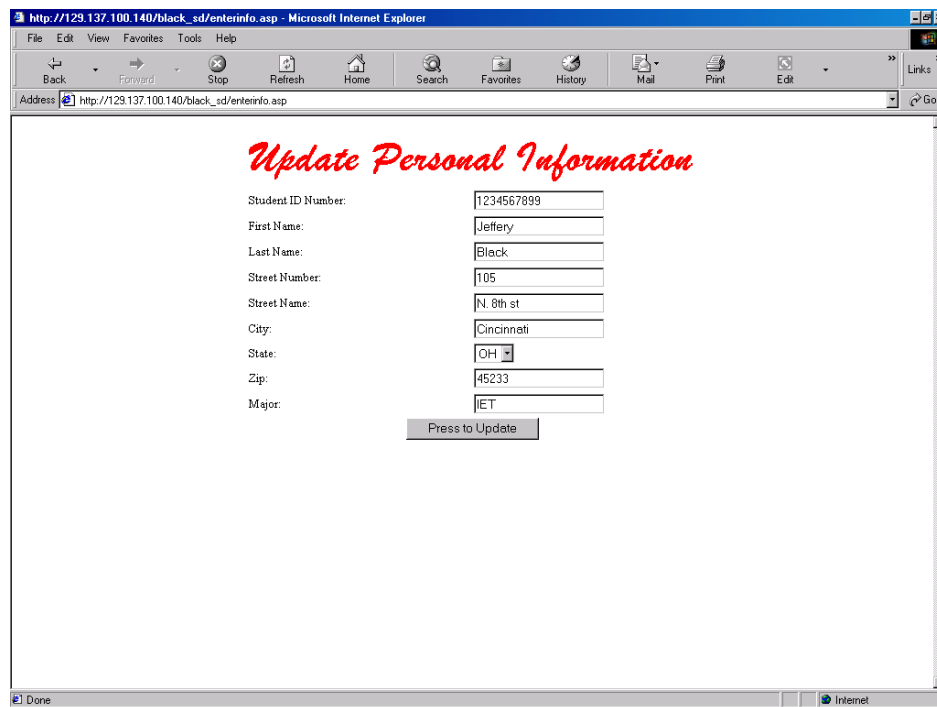
Figure 4 Student Homepage

From the receivelogin.asp page the student can navigate to view his personal information or current degree plan, change any information or classes, and view current jobs. The Web page also gives the user information on the approval status of their degree plan and help if needed.

6.1.3 Screen 3

The first option for the user is to view his/her personal information. This page allows the student to view his/her current information stored in the database. The page places the information on a table that is viewed by the user. Under the table, the page gives the user the option to change the information or go back to the home page. The page also has a link to the help page that will send the user to the specific part of the help page that gives information about the current page.

6.1.4 Screen 4



The screenshot shows a Microsoft Internet Explorer window with the address bar displaying `http://129.137.100.140/black_sd/enterinfo.asp`. The page content is titled *Update Personal Information* in red cursive font. Below the title is a form with the following fields and values:

Field	Value
Student ID Number:	1234567899
First Name:	Jeffery
Last Name:	Black
Street Number:	105
Street Name:	N. 8th st
City:	Cincinnati
State:	OH
Zip:	45233
Major:	IET

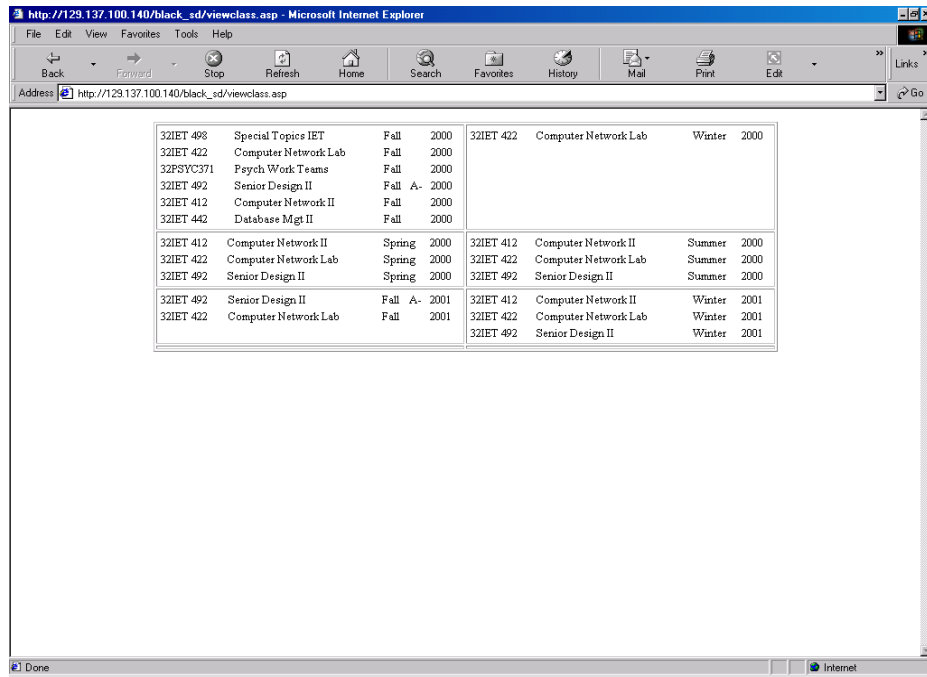
At the bottom of the form is a button labeled "Press to Update".

Figure 5 Updating Information

If the user decides to change his/her personal information, the user is sent to this page. This page gives the user several text boxes and a drop-down box to input his/her demographic information. After placing the information into the textboxes, the user can select Press to Update button to enter the data. The screen will show the information one more time in a table to confirm from the user that the data is correct. If the user selects

cancel, the application will allow the user to change the information again. If the user selects OK then the application will save the data in the database and bring the user back to the homepage.

6.1.5 Screen 5



The screenshot shows a Microsoft Internet Explorer window with the address bar displaying 'http://129.137.100.140/black_ed/viewclass.asp'. The main content area displays a table with two columns of class information. The table is divided into two sections by a horizontal line. The first section contains classes for Fall 2000, and the second section contains classes for Spring 2000, Fall A- 2000, and Winter 2000. Each row lists a course ID, course name, quarter, and year.

32IET 498	Special Topics IET	Fall	2000	32IET 422	Computer Network Lab	Winter	2000
32IET 422	Computer Network Lab	Fall	2000				
32PSYC371	Psych Work Teams	Fall	2000				
32IET 492	Senior Design II	Fall A-	2000				
32IET 412	Computer Network II	Fall	2000				
32IET 442	Database Mgt II	Fall	2000				
32IET 412	Computer Network II	Spring	2000	32IET 412	Computer Network II	Summer	2000
32IET 422	Computer Network Lab	Spring	2000	32IET 422	Computer Network Lab	Summer	2000
32IET 492	Senior Design II	Spring	2000	32IET 492	Senior Design II	Summer	2000
32IET 492	Senior Design II	Fall A-	2001	32IET 412	Computer Network II	Winter	2001
32IET 422	Computer Network Lab	Fall	2001	32IET 422	Computer Network Lab	Winter	2001
				32IET 492	Senior Design II	Winter	2001

Figure 6 Viewing Degree Plan

This screen shows the user his/her current classes in the degree plan. If the user currently has no classes in the degree plan, the screen will give the student two links. The first link will send the user to the homepage and the other link sends the user to the screen so he/she can place classes in the database. If the user has classes in the degree plan, the classes will be shown in a two-column table divided by each quarter. The screen will give links back to the homepage and the help page.

6.1.6 Screen 6

The screenshot shows a web browser window titled "http://129.137.100.140/black_sd/developdp.asp - Microsoft Internet Explorer". The address bar shows "http://129.137.100.140/black_sd/developdp.asp". The page content is titled "Degree Plan Development Page".

At the top, there are two dropdown menus: "Quarter" set to "Fall" and "Year" set to "2001". Below these are two columns of input fields: "Class Name" and "Class Number". The "Class Name" column has a dropdown menu with the following options: "Special Topics IET", "Computer Network II", "Database Mgt II", "Senior Design II", "Psych Work Teams", "CO-OP", and "Programming in C". The "Class Number" column has seven empty text boxes.

To the right of the "Class Number" column is a column for "Audit (Checked for Yes)" with radio buttons and the text "Audit". Next to it is a "Grade" column with dropdown menus, each showing a hyphen. To the right of the "Grade" column is a "Prerequisite" column with buttons labeled "Prerequisite".

At the bottom center, there is a button labeled "Enter Classes".

Figure 7 Adding Classes

The user will use this screen to place classes in his/her degree plan. The screen will have various drop-down boxes and textboxes. The user will be able to pick which class he/she wants to take from a list on the drop-down box. The screen will automatically give the user the class number and number of credit hours. The user will also need to pick the month, year, which class to audit (if any), and may place a grade for the class if the class was already taken. The prerequisite button will open a new window that will display any class requirements for the select class.

6.1.7 Screen 7

The user will go to this part of the Web site to edit classes currently in the degree plan. The Web site will first ask the student to pick a year and quarter to edit. If there are no classes in this quarter, the application will ask to input a different quarter. If the

user selects a quarter that has classes listed, the Web site will give a list of the class names and numbers.

Editing the Degree Plan

Check to Remove Class	Current Class Number	Current Class Name	Grade	Quarter	Year
☐	32IET 498	Special Topics IET		Fall	2000
☐	32IET 422	Computer Network Lab		Fall	2000
☐	32PSYC371	Psych Work Teams		Fall	2000
☐	32IET 492	Senior Design II	A-	Fall	2000
☐	32IET 412	Computer Network II		Fall	2000
☐	32IET 442	Database Mgt II		Fall	2000

Submit Changes Clear Changes

Figure 8 Editing Degree Plan

The student has the options of selecting classes to be removed from the degree plan, change the grade, or move the class to another quarter in the degree plan.

6.1.8 Screen 8

The Available Jobs screen shows the user available jobs within his major. The page determines the user's major and does a search of jobs marked available and in his major. The screen shows the name of the company, contact information, name and information of the job available, and when the job is available.

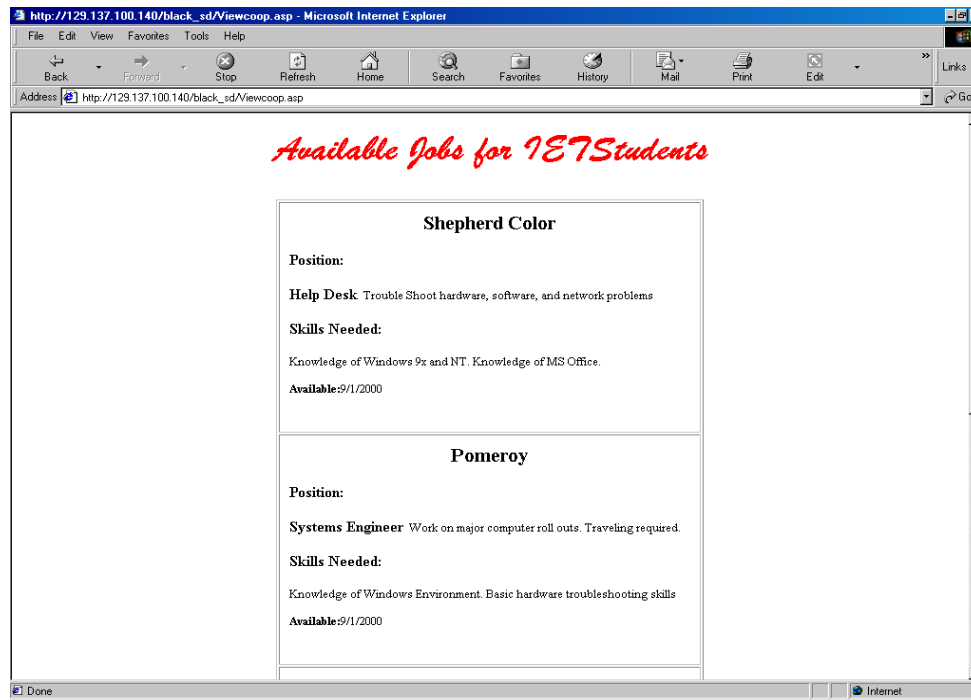


Figure 9 Available Jobs

6.1.9 Screen 9

The Help page will have information on how to add, update, and delete classes in a degree plan. The page will provide information about how to look for jobs, change the user's personal information and show examples of a completed degree plan. The page will be set up so when the user selects the help link for any page, the link will send the user to the information about the page they were sent from. If a student hits the help link on the Personal Information page, the Help page will go to the section that talks about how to input user information and how to change information already in the database.

6.2 Teacher Web Front-end

The following describes the sequence of steps a user would take to use the teacher front-end:

6.2.1 Screen 1

The login.htm is the first page a user will see. This page welcomes the user to the page and gives a prompt for a Student password or a teacher user name and password. The page also has two links. The first link brings up a small window explaining to the user what the Degree Plan Builder is used for.

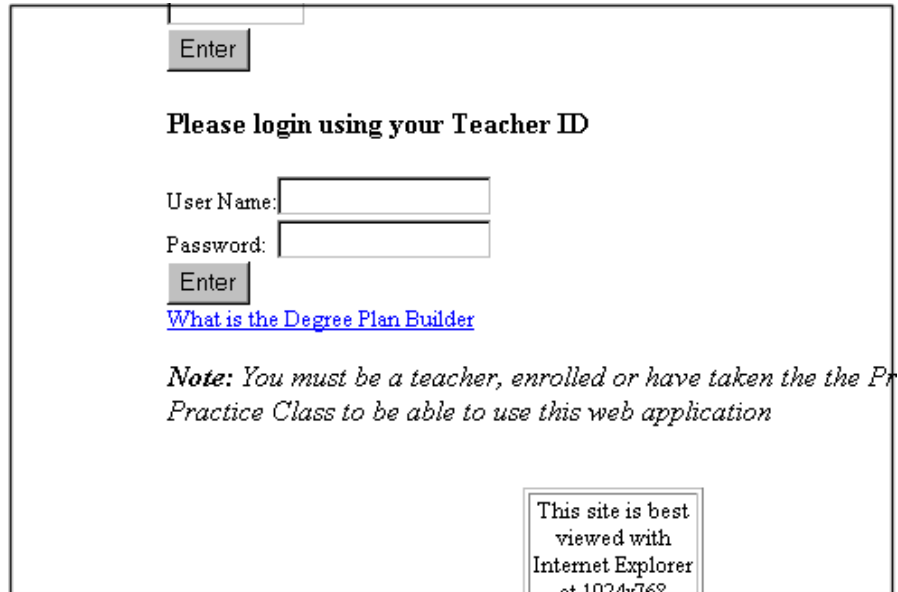
The screenshot shows a web browser window with a login form. At the top, there is a text input field followed by an 'Enter' button. Below this, the text 'Please login using your Teacher ID' is displayed. The form includes a 'User Name:' label with a text input field, and a 'Password:' label with a text input field. Below the password field is another 'Enter' button. A blue hyperlink 'What is the Degree Plan Builder' is located below the 'Enter' button. A note in italics states: 'Note: You must be a teacher, enrolled or have taken the the Pr Practice Class to be able to use this web application'. In the bottom right corner, a small box contains the text: 'This site is best viewed with Internet Explorer at 1024x768'.

Figure 10 Teacher's Login Screen

6.2.2 Screen 2

After logging into the application, the Web site gives the teacher a list of students for him/her to select. The list of students is based on the range of names the administration set for the teacher. After choosing a student, the user hits enter to view the degree plan.

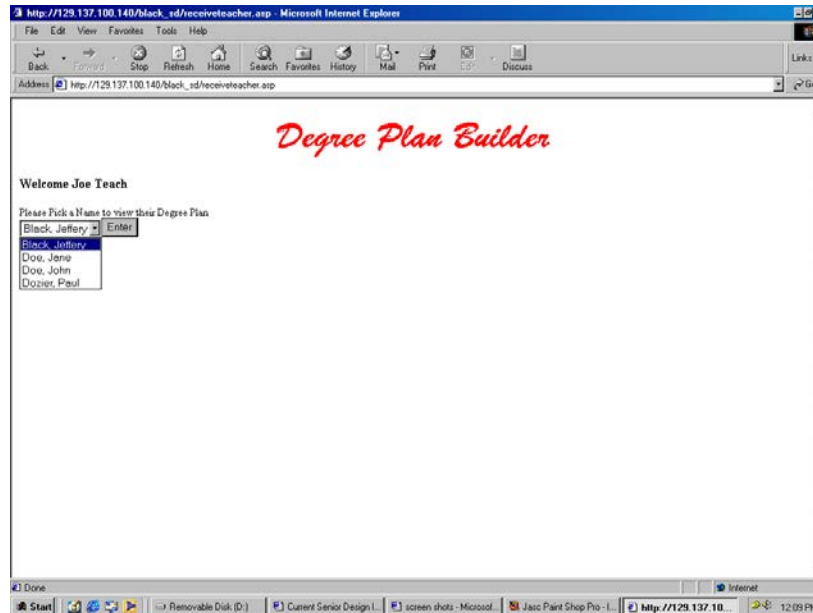


Figure 11 Choosing a Student

6.2.3 Screen 3

The final screen for the teacher Web front-end shows the teacher the student's degree plan. The screen shows the Student's name at the top of the screen, the degree plan in the middle, and a drop-down box at the bottom of the screen so the teacher can choose to approve or not approve the degree plan. Once the teacher submits this approval, the Web site takes the teacher back to the second screen.

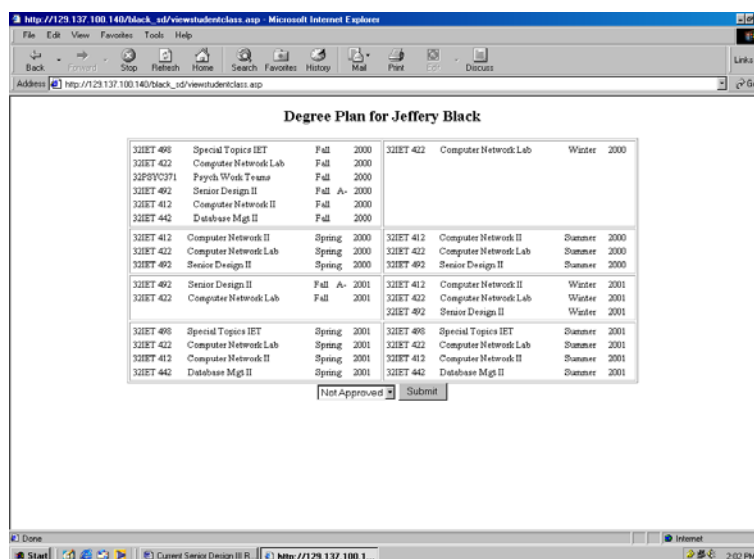


Figure 12 Student's Degree Plan

6.3 Administration Access front-end

The following describes the sequence of steps a user would see on the Access front-end. All code from the Access front-end may be found under Appendix C.

6.3.1 Screen 1

This screen opens automatically when the user opens the link to the project. This page is the main window for the Access front-end and has buttons to open the other input windows.

6.3.2 Screen 2

This screen allows the user to input class information. This screen allows the user to update or remove any current classes in the database. Along with adding classes, the administrator has the ability to add, update, or remove any prerequisites for each class

PrereqNum	PrereqName

Figure 13 Class Information

6.3.3 Screen 3

The user may use this screen to input the students into the database so that the students may access the Web based front-end. Along with putting the student into the database, the administrator can edit and delete students as needed.

frmStudent : Form

Student Number: 111111111

First Name: Jane

Last Name: Doe

Street Number: 111

Street Name: Halsey av

City: Cincinnati

State: OH

Zip: 45204

Major: MET

SecMajor:

◀ Add Student Delete Student ▶

Record: 1 of 10

Figure 14 Add/Remove Student

6.3.4 Screen 4

The Student's Degree Plan screen allows the administrator to view the development of the student's degree plan. The admin can pick a student, pick the year and quarter for the student, and the window will bring up the student's courses for that quarter.

Student's Degree Plan

Student Number Year Quarter Enter

Class Number	Class name	Student's Grade

Figure 15 View Degree Plan

6.3.5 Screen 5

The user uses this screen to input information about the co-op jobs. The user will also use this screen to update the availability status of the co-op jobs for the Web based front-end and run reports that deal with co-op jobs.

Major	Job Position	Job Description	Skills
IET	Help Desk	Trouble Shoot hardware, software,	Knowledge of \
IET	Intranet Web Admin	Create, and maintain web pages c	Knowledge of I

Figure 16 Company Information

6.3.6 Screen 6

The final screen for the Access front-end is the Student's co-op Form. This form allows the user to select a student, view the student's previous co-op jobs and allow for the user to add a co-op. If the user chooses to add a co-op, the window brings up a new window that allows the user to type in the information.

Qua	Company Name	Contact Name	Phone Number
1	Pomeroy	John Doe	5131234567
2	Pomeroy	John Doe	5131234567
3	Pomeroy	John Doe	5131234567
4	Pomeroy	John Doe	5131234567

Figure 17 Coop Form

7. Conclusion and Recommendations

The Degree Plan Builder was created to give the students at CAS an easier way to develop and maintain their degree plan instead of having to run a DOS program from a floppy or having to use a Web application with many user interface problems. With the ability to pick the class name and to receive all the needed information for a class, this Web site will allow the student to add classes without having to look up the information before hand. The Degree Plan Builder improves on the current Web application by being able to move classes around, picking courses and not typing the information in, being able to use the +/- grading scale, and checking to see if the degree plan is approved or not online.

There are a few recommendations for future development of the Degree Plan Builder. The Access front-end for the administrators is not complete. The Access front-end was made to provide a simple interface for an administrator to input students, companies, and classes into the database. It also allows the administrator to look at classes for a student and some statistics on pay rates per major. Future development should look at this front-end and add more functionality to make the front-end more useful to the administration. One place that needs improvement is adding, deleting and updating student or company information. Currently the administrator has to look at each student or company record one at a time until he/she finds the wanted record and edit the information. Adding the functionality to run a search for the wanted will cut down on the amount of time and effort the administrator uses to update the information.

Another part of the front-end that needs work is viewing a student's degree plan. Currently the Access program allows for a person to view a student's degree plan one quarter at a time. It may be useful if the person could view all of the degree plan at once or yearly instead of quarterly.

I also have some recommendations for future development of the Web front-end for the students. The front-end currently allows the student to move classes to different quarters on class at a time. Students tend to change co-op sections that mean they will have to move whole quarters around. The functionality of being able to rearranging whole quarters

would make it easier for these types of students to maintain the degree plan.

A future issue with the Web front-end will be scalability. If this program was to be implemented university wide there will be a problem with adding classes to the degree plan. With hundreds of classes offered each quarter, it would be hard for a student to look through the whole list of class names to find what he/she needs. Adding the ability to look for a class in particular college in the university would make the search a lot easier for the student.

One recommendation I received from many students was for the Web front-end project to be able to show statistics of pay rates for the whole major per quarter. The students mention that the current PPCP Web site shows each major and what the average pay for each quarter. This information would be useful for when they go on interviews with potential employers to have an idea of what their skills are worth in the market.

The final recommendation I can give for this project is for future developers to work with the PPCP office to have this project accepted to replace the current degree plan. This project was created without input from the PPCP. I would make a second revision of this project adding and removing functionality that was suggested from the PPCP office.

Appendix A.

A.1 Graphical Representation of the SQL Database

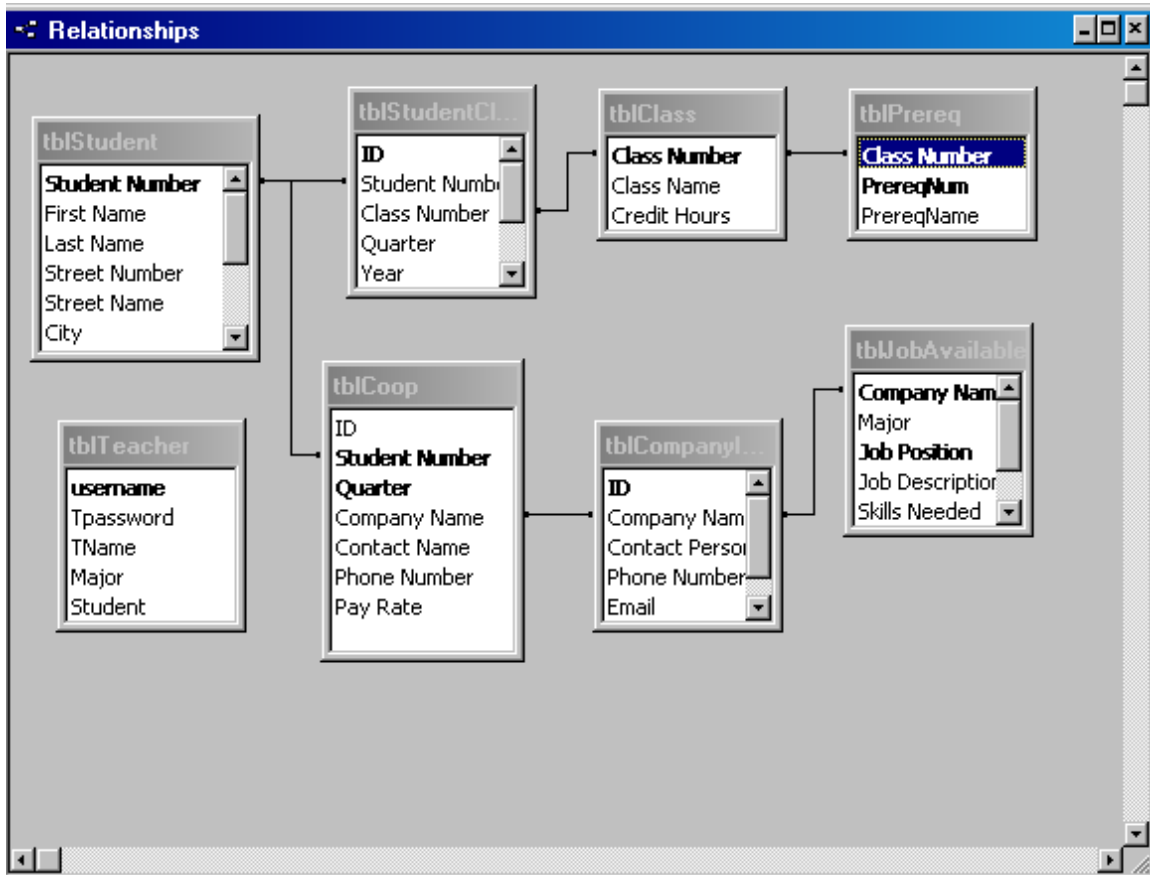


Figure 18 Table Relationships

A.2 SQL Table Scripts

GO

```
CREATE TABLE [dbo].[tblClass] (  
    [Class Number] [varchar] (10) NOT NULL ,  
    [Class Name] [varchar] (20) NULL ,  
    [Credit Hours] [int] NULL  
) ON [PRIMARY]  
GO
```

```
CREATE TABLE [dbo].[tblCompanyInfo] (  
    [ID] [int] IDENTITY (1, 1) NOT NULL ,  
    [Company Name] [varchar] (75) NULL ,  
    [Contact Person] [varchar] (50) NULL ,
```



```

        [Phone Number] [varchar] (20) NULL ,
        [Email] [varchar] (50) NULL ,
        [Fax] [varchar] (20) NULL
    ) ON [PRIMARY]
GO

CREATE TABLE [dbo].[tblCoop] (
    [ID] [int] IDENTITY (1, 1) NOT NULL ,
    [Student Number] [varchar] (10) NOT NULL ,
    [Quarter] [tinyint] NULL ,
    [Company Name] [varchar] (50) NULL ,
    [Contact Name] [varchar] (75) NULL ,
    [Phone Number] [varchar] (20) NULL ,
    [Pay Rate] [money] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[tblJobAvailable] (
    [Company Name] [varchar] (50) NULL ,
    [Major] [varchar] (50) NULL ,
    [Job Position] [varchar] (50) NULL ,
    [Job Description] [varchar] (200) NULL ,
    [Skills Needed] [varchar] (200) NULL ,
    [Available] [varchar] (10) NULL ,
    [Date Available] [datetime] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[tblPrereq] (
    [Class Number] [varchar] (10) NOT NULL ,
    [PrereqNum] [varchar] (10) NOT NULL ,
    [PrereqName] [varchar] (20) NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[tblStudent] (
    [Student Number] [varchar] (10) NOT NULL ,
    [First Name] [varchar] (20) NULL ,
    [Last Name] [varchar] (50) NULL ,
    [Street Number] [int] NULL ,
    [Street Name] [varchar] (10) NULL ,
    [City] [varchar] (50) NULL ,
    [State] [varchar] (50) NULL ,
    [Zip] [int] NULL ,
    [Major] [varchar] (5) NULL ,
    [SecMajor] [varchar] (5) NULL ,

```

```
        [Approved] [int] NULL
    ) ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[tblStudentClassGrade] (
    [ID] [int] IDENTITY (1, 1) NOT NULL ,
    [Student Number] [varchar] (10) NULL ,
    [Class Number] [varchar] (10) NULL ,
    [Quarter] [varchar] (10) NULL ,
    [Year] [int] NULL ,
    [Days] [varchar] (6) NULL ,
    [Audit] [varchar] (10) NULL ,
    [Grade] [varchar] (2) NULL
) ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[tblTeacher] (
    [username] [varchar] (20) NULL ,
    [Tpassword] [varchar] (75) NULL ,
    [TName] [varchar] (75) NULL ,
    [Major] [varchar] (4) NULL ,
    [Student] [varchar] (5) NULL
) ON [PRIMARY]
```

A.3 SQL View Scripts

```
GO
```

```
CREATE VIEW dbo.v_Coop
AS
SELECT [Student Number], Quarter, [Company Name],
    [Contact Name], [Phone Number], [Pay Rate]
FROM tblCoop
```

```
GO
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```
CREATE VIEW dbo.v_DegreePlan
AS
SELECT tblStudentClassGrade.[Class Number] AS ClassNum,
    tblClass.[Class Name] AS ClassName,
```

```

tblStudentClassGrade.Grade, tblStudentClassGrade.Quarter,
tblStudentClassGrade.Year AS Year,
tblStudentClassGrade.[Student Number] AS StudentNum
FROM tblStudentClassGrade INNER JOIN
tblClass ON
tblStudentClassGrade.[Class Number] = tblClass.[Class Number]
GO

```

```

CREATE VIEW dbo.v_OverallPayRates
AS
SELECT tblStudent.Major, AVG(tblCoop.[Pay Rate]) AS AvgRate,
MIN(tblCoop.[Pay Rate]) AS MinRate, MAX(tblCoop.[Pay Rate])
AS MaxRate
FROM tblCoop INNER JOIN
tblStudent ON
tblCoop.[Student Number] = tblStudent.[Student Number]
GROUP BY tblStudent.Major
GO

```

```

CREATE VIEW dbo.v_PayRatesQuarterly
AS
SELECT tblStudent.Major, tblCoop.Quarter, MIN(tblCoop.[Pay Rate])
AS MinRate, MAX(tblCoop.[Pay Rate]) AS MaxRate,
AVG(tblCoop.[Pay Rate]) AS AvgRate
FROM tblCoop INNER JOIN
tblStudent ON
tblCoop.[Student Number] = tblStudent.[Student Number]
GROUP BY tblStudent.Major, tblCoop.Quarter
GO

```

```

CREATE VIEW dbo.vYear
AS
SELECT distinct [Student Number], Year
FROM tblStudentClassGrade

```

A.3 SQL Stored Procedure Scripts

```

GO

```

```

CREATE procedure sp_InsertClass(@studentnumber varchar(10), @classnumber
varchar(10), @quarter varchar(10), @year int, @audit varchar(10), @grade varchar(2))
as

```

```

Begin
update tblStudent set Approved = 0 where [Student Number] = @studentnumber

```

end

```
if not exists(select id from tblstudentclassgrade where [Student Number] =  
@studentnumber and [Class Number] = @classnumber and Quarter = @quarter and Year  
= @year)
```

```
BEGIN
```

```
    Insert into tblStudentClassGrade ([Student Number], [Class Number], [Quarter],  
[Year], Audit, Grade) values (@studentnumber, @classnumber, @quarter, @year,  
@audit, @grade)
```

```
END
```

```
GO
```

```
CREATE procedure sp_selectStudent(@range1 varchar(10), @range2 varchar(10))  
as
```

```
BEGIN
```

```
Select * from tblStudent where [Last Name] like "[^ @Range1@range2]%" order by  
[Last name], [first name]
```

```
END
```

```
GO
```

```
CREATE procedure sp_UpdateClass(@studentnumber varchar(10), @classnumber  
varchar(10), @quarter varchar(10), @year int, @grade varchar(2), @quarter2  
varchar(10), @year2 int)
```

```
as
```

```
Begin
```

```
    update tblStudent set Approved = 0 where [Student Number] = @studentnumber  
end
```

```
if exists(select id from tblstudentclassgrade where [Student Number] = @studentnumber  
and [Class Number] = @classnumber and Quarter = @quarter and Year = @year)
```

```
BEGIN
```

```
    delete from tblstudentclassgrade where [Student Number] = @studentnumber and  
[Class Number] = @classnumber and Quarter = @quarter2 and Year = @year2
```

```
    Insert into tblStudentClassGrade ([Student Number], [Class Number], [Quarter],  
[Year], Grade) values (@studentnumber, @classnumber, @quarter, @year, @grade)
```

```
END
```

```
else
```

Begin

Update tblStudentClassGrade set [Student Number] = @studentnumber, [Class Number] = @classnumber, Quarter = @quarter, Year = @year, Grade = @grade where [Student Number] = @studentnumber and [Class Number] = @classnumber and Quarter = @quarter2 and Year = @year2

End

Appendix B.

B.1 Student Web Front-End Code

B.1.1 Login.htm

```
<html>
<head>
<meta NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<title></title>
<script language="Vbscript">
Function NewWindow()

window.open "Whatisit.htm","Hello","width=450,height=500"

end Function

sub cmdTeacher_onclick

frmLogin.action = "receiveteacher.asp"
frmLogin.submit

end sub
</script>
</head>
<body>
<form method="post" action="receivelogin.asp" name="frmLogin">

<center><IMG height=184 src="images\welcome.gif" width=402></center><br>

<table width="500" border="0" align="center">
<tr>
<td>
<h3>Please login using your&nbsp;Student ID</h3>
<P>
<input type="password" name="userID" size="9">
<br>
<input type="submit" value="Enter"><br></P>
<H3>Please login using your&nbsp;Teacher ID</H3>
<P>User Name:<INPUT id=txtUserName size=16 name=txtUserName><BR>
Password:&nbsp;&nbsp;&nbsp;<INPUT type="password" id=txtPassword size=16
name=txtPassword>
<br><INPUT id=cmdTeacher type=button value="Enter" name=cmdTeacher>
<br>
<A href="vbscript:NewWindow()">What is the Degree Plan Builder</A>
<br>
```

```

<br>
<font size="+1"><i><b>Note:</b> You must be a teacher or a student&nbsp;enrolled or
have taken the the Professional Practice Class to be able to use this web
application.</i></font><br>
<br>
<br></P>
</td>
</tr>
</table>
<table width="100" align="center" border="1">
<tr>
<td align="middle">
This site is best viewed with Internet Explorer at 1024x768 screen Resolution
</td>
</tr>
</table>
</form>
</body>
</html>

```

B.1.2 Receivelogin.asp

```

<%@ Language=VBScript %>
<script id="DebugDirectives" runat="server" language="javascript">
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</script>
<html>
<head>
<script LANGUAGE="JavaScript">
<!--
function imgover(imgname){
imgname.src = "arrow.gif"
}
function imgout(imgname){
imgname.src = "blank.gif"
}
// End -->
</script>
</head>
<body>
<form action=enterinfo.asp>

<%

```

```
dim sqlstr
dim strconn
dim conn
dim tmpID
dim tmpID2
dim strID
dim rs
```

```
'Get information from Login.asp(the userID) and places it into a session.contents variable
strID = Request("userID")
Session.Contents("sesUserID")=strID
```

```
'Check to see if the ID is correct length
if len(strID) <> 10 then
    Response.Write("<a href='\"'login.htm\"'>Sorry, wrong Student ID. Please Click here to
login again.</a>")
else
```

```
'open connection
strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")
conn.open strconn
```

```
'Gets student information--Needs to have error checking placed into it
sqlstr="SELECT * FROM tblStudent WHERE [Student Number] = " & strID
rs.Open sqlstr, conn, 2, 2
if rs.EOF or rs.BOF then
    Response.Write("<a href='\"'login.htm\"'>Sorry, wrong Student ID. Please Click here to
login again.</a>")
else
    'write the stuff
```

```
Response.Write("<center><img src='\"'images\logo.gif\"'></center><BR>")
```

```
Response.Write("<center><a href='\"'degreeplan.asp\"'>Home</a> | <a
href='\"'lookatinfo.asp\"'>View Personal Data</a> | <a href='\"'enterinfo.asp\"'>Edit
Personal Data</a> | <a href='\"'viewclass.asp\"'>View Degree Plan</a> | <a
href='\"'developdp.asp\"'>Add Classes</a> | <a href='\"'getquarter.asp\"'>Edit Degree
Plan</a> | <a href='\"'Viewcoop.asp\"'>View Jobs</a> | <a href='\"'help.htm\"'
target='\"' _blank\"'>Help</a></center>")
```

```
Response.Write("<h3 align='\"'center\"'>")
Response.write("Welcome ")
```



```

response.write rs("First Name")
Response.Write(" ")
Response.Write rs("Last Name")
Response.Write("</h3>")
Response.Write("<table border=0 align=""center"">")
Response.Write("<tr><td align=""center"">")
response.write("<font size=""4"">Please press one of the links below:<br>")
Response.Write("</td></tr>")
Response.Write("<tr><td>")
response.write("<BR><font size=""4"">")
response.write("<img SRC=""blank.gif"" name=""pic0"" WIDTH=""10""
HEIGHT=""10""><a href=""lookatinfo.asp"" onMouseOver=""imgover(pic0)""
onMouseOut=""imgout(pic0)"">Personal information</a> View your personal
information<BR>")
response.write("<img SRC=""lank.gif"" name=""pic1"" WIDTH=""10""
HEIGHT=""10""><a href=""enterinfo.asp"" onMouseOver=""imgover(pic1)""
onMouseOut=""imgout(pic1)"">Update Personal information</a> Change your home
address and major<BR>")
response.write("<img SRC=""blank.gif"" name=""pic2"" WIDTH=""10""
HEIGHT=""10""><a href=""viewclass.asp"" onMouseOver=""imgover(pic2)""
onMouseOut=""imgout(pic2)"">View degree plan</a> Look over your whole degree
plan<BR>")
response.write("<img SRC=""blank.gif"" name=""pic3"" WIDTH=""10""
HEIGHT=""10""><a href=""developdp.asp"" onMouseOver=""imgover(pic3)""
onMouseOut=""imgout(pic3)"">Develop your degree plan</a> Add classes to your
degree plan<BR>")
response.write("<img SRC=""blank.gif"" name=""pic5"" WIDTH=""10""
HEIGHT=""10""><a href=""getquarter.asp"" onMouseOver=""imgover(pic5)""
onMouseOut=""imgout(pic5)"">Edit Your Degree Plan</a> Edit and update your current
degree plan<BR>")
response.write("<img SRC=""blank.gif"" name=""pic4"" WIDTH=""10""
HEIGHT=""10""><a href=""Viewcoop.asp"" onMouseOver=""imgover(pic4)""
onMouseOut=""imgout(pic4)"">Available jobs</a> View current jobs that are available
for your major<BR><BR>")
Response.Write("</td></tr>")
Response.Write("<tr><td align=""center"">")
Response.Write "<font size=""4"">"
if rs("Approved") = 0 then
    Response.Write "Your degree plan has <B>not been approved</B> by your advisor"
else
    Response.Write "Your degree plan has <B>been approved</B> by your advisor"
end if

Response.Write("</td></tr>")
Response.Write("</table>")

```

```

response.write("</form>")
response.write("</body>")
response.write("</html>")
end if

```

```

end if
%>

```

B.1.3 Lookatinfo.asp

```

<%@ Language=VBScript %>
<html>
<head>
<meta NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</head>
<body>
<%

dim conn
dim rs
dim strconn
dim strsql

strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn=server.CreateObject("adodb.connection")
set rs=server.CreateObject("adodb.recordset")
set rs2=server.CreateObject("adodb.recordset")
conn.Open strconn

strsql="Select * from tblStudent where [Student Number] = " &
Session.Contents("sesUserID")
rs.Open strsql, conn, 2, 2

%>
<center><img SRC="images/perInfo.gif" WIDTH="368" HEIGHT="60"></center>
<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a> | <a href="help.htm"
target="_blank">Help</a></center><BR>
<BR>
<table border="1" align="center" width="400" cellpadding="5">

```

```

<tr>
  <td width="50%"><b>First Name</b></td>
  <td><%=rs("First Name")%></td>
</tr>
<tr>
  <td width="50%"><b>Last Name</b></td>
  <td><%=rs("Last Name")%></td>
</tr>
<tr>
  <td><b>Street Number</b></td>
  <td><%=rs("Street Number")%></td>
</tr>
<tr>
  <td><b>Street Name</b></td>
  <td><%=rs("Street Name")%></td>
</tr>
<tr>
  <td><b>City</b></td>
  <td><%=rs("City")%></td>
</tr>
<tr>
  <td><b>State</b></td>
  <td><%=rs("State")%></td>
</tr>
<tr>
  <td><b>Zip</b></td>
  <td><%=rs("Zip")%></td>
</tr>
<tr>
  <td><b>Major</b></td>
  <td><%=rs("Major")%></td>
</tr>
</table>
<BR>
<BR>
<table width="400" align="center" border=0>
<tr>
<td align="left" valign="top"><font size="4">
<a href="degreeplan.asp"><---Back to home</a>
<td align="right" valign="top"><font size="4">
<a href="enterinfo.asp">Change Personal Information---</a><br>
</tr>
</table>
</body>
</html>

```

B.1.4 Enterinfo.asp

```
<html>
<head>
<META name=VI60_defaultClientScript content=VBScript>
<meta NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<title></title>
<script language="Vbscript">

Sub cmdSubmit_onclick
dim tmpSubmit
dim tmpStNum
tmpSubmit = 0
tmpStnum=frmEnter.txtStreetNumber.value

if len(frmEnter.txtfirstname.value) > 20 then
    msgbox "You have too many characters for the first name(max of 20 characters)"
    frmEnter.txtfirstname.focus
    tmpsubmit=1
end if
if len(frmEnter.txtlastname.value) > 50 then
    msgbox "You have too many characters for the last name(max of 20 characters)"
    frmEnter.txtlastname.focus
    tmpsubmit=1
end if
if len(frmEnter.txtlastname.value) > 50 then
    msgbox "You have too many characters for the last name(max of 20 characters)"
    frmEnter.txtlastname.focus
    tmpsubmit=1
end if
for x = 1 to len(frmEnter.txtStreetNumber.value)

if asc(mid(tmpStNum, x, 1)) < 48 or asc(mid(tmpStNum, x, 1)) > 57 then
    if tmpsubmit=0 then
        msgbox "You you inputed an invalid street number"
    end if
    frmEnter.txtstreetnumber.focus
    tmpsubmit=1
end if
next

if len(frmEnter.txtStreetName.value) > 10 then
    msgbox "You inputed street name that is too long(max of 10 characters)."
```

```

if len(frmEnter.txtcity.value) > 50 then
    msgbox "You inputted a city name that is too long(max of 50 characters)"
    frmEnter.txtcity.focus
    tmpsubmit=1
end if

for x = 1 to len(frmEnter.txtZip.value)

if asc(mid(frmEnter.txtZip.value, x, 1)) < 48 or asc(mid(frmEnter.txtZip.value, x, 1)) >
57 then
    if tmpsubmit=0 then
        msgbox "You you inputed an invalid street number"
    end if
    frmEnter.txtZip.focus
    tmpsubmit=1
end if
next

if len(frmEnter.txtmajor.value)>3 then
    msgbox "You enter an invalid Major Name"
    frmEnter.txtmajor.focus
    tmpsubmit=1
end if
frmEnter.txtMajor.value = ucase(frmEnter.txtMajor.value)

if tmpsubmit = 0 then
    frmEnter.submit
end if

End Sub

Sub window_onload

frmEnter.txtState.value = frmEnter.hidstate.value

End Sub

</SCRIPT>
</head>
<body>
<form method="post" action="updateinfo.asp" name="frmEnter">
<%
dim sqlstr
dim strconn

```

```
dim conn
dim strID
dim tmpStr
dim tmpLocation
dim rs
dim compareStr
```

```
'Getting the address and taking the information needed out of it
tmpStr = Session.Contents("sesUserID")
```

```
strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")
set rs2 = server.createobject("adodb.recordset")
conn.open strconn
```

```
sqlstr="SELECT * FROM tblStudent WHERE [Student Number] = " & tmpStr
rs.Open sqlstr, conn, 2, 2
```

```
%>
<center><img SRC="images/update.gif" WIDTH="508" HEIGHT="62"></center>
<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a> | <a href="help.htm"
target="_blank">Help</a></center><BR>
<BR>
<table width="50%" align="center">
<tr>
<td width="50%">Student ID Number: </td>
<td width="50%"><input name="txtStudentNumber" value="<%=rs("Student
Number")%>" readonly></td>
</tr>
<tr>
<td>First Name:</td>
<td><input name="txtFirstName" value="<%=rs("First Name")%>"></td>
</tr>
<tr>
<td>Last Name:</td>
<td><input name="txtLastName" value="<%=rs("Last Name")%>"></td>
</tr>
<tr>
<td>Street Number:</td>
```

```

<td><input name="txtStreetNumber" value="<%=rs("Street Number")%>"></td>
</tr>
<tr>
<td>Street Name:</td>
<td><input name="txtStreetName" value="<%=rs("Street Name")%>"></td>
</tr>
<tr>
<td>City:</td>
<td><input name="txtCity" value="<%=rs("City")%>"></td>
</tr>
<tr>
<td>State:</td>
<td><input type="hidden" name="hidState" value="<%=rs("State")%>">
<SELECT id=txtState name=txtState>
<option value="AL">AL</option>
<option value="AK">AK</option>
<option value="AZ">AZ</option>
<option value="AR">AR</option>
<option value="CA">CA</option>
<option value="CO">CO</option>
<option value="CT">CT</option>
<option value="DE">DE</option>
<option value="DC">DC</option>
<option value="FL">FL</option>
<option value="GA">GA</option>
<option value="HI">HI</option>
<option value="ID">ID</option>
<option value="IL">IL</option>
<option value="IN">IN</option>
<option value="IA">IA</option>
<option value="KS">KS</option>
<option value="KY">KY</option>
<option value="LA">LA</option>
<option value="ME">ME</option>
<option value="MD">MD</option>
<option value="MA">MA</option>
<option value="MI">MI</option>
<option value="MN">MN</option>
<option value="MS">MS</option>
<option value="MO">MO</option>
<option value="MT">MT</option>
<option value="NE">NE</option>
<option value="NV">NV</option>
<option value="NH">NH</option>
<option value="NJ">NJ</option>
<option value="NM">NM</option>

```

```

<option value="NY">NY</option>
<option value="NC">NC</option>
<option value="ND">ND</option>
<option value="OH">OH</option>
<option value="OK">OK</option>
<option value="OR">OR</option>
<option value="PA">PA</option>
<option value="RI">RI</option>
<option value="SC">SC</option>
<option value="SD">SD</option>
<option value="TN">TN</option>
<option value="TX">TX</option>
<option value="UT">UT</option>
<option value="VT">VT</option>
<option value="VA">VA</option>
<option value="WA">WA</option>
<option value="WV">WV</option>
<option value="WI">WI</option>
<option value="WY">WY</option>
</SELECT>
</td>
</tr>
<tr>
<td>Zip:</td>
<td><input name="txtZip" value="<%=rs("Zip")%>"></td>
</tr>
<tr>
<td>Major:</td>
<td><input name="txtMajor" value="<%=rs("Major")%>"></td>
</table>
<center><input type="button" name="cmdSubmit" value="Press to Update"></center>
</form>
</body>
</html>

```

B.1.5 Updateinfo.asp

```

<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<Script language="Vbscript">
<!--
Sub cmdOk_onClick

window.navigate "degreeplan.asp"

```


End sub

Sub cmdCancel_onClick

window.navigate "enterinfo.asp"

End sub

-->

</Script>

</HEAD>

<BODY>

<form>

<%

dim strsql

dim strconn

dim conn

dim strID

dim tmpStr

'open connection to db server

strconn = "Driver={SQL

Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;DATABASE=jblacksd"

set conn = server.createobject("adodb.connection")

set rs=server.CreateObject("adodb.recordset")

conn.open strconn

tmpStr=session.Contents("sesUserID")

'update tblstudent

strsql="UPDATE tblStudent SET tblStudent.[First Name] ='" & request("txtFirstName") & "', tblStudent.[Last Name] ='" & request("txtLastName") & "', tblStudent.[Street Number] ='" & request("txtStreetNumber") & "', tblStudent.[Street Name] ='" & request("txtStreetName") & "', tblStudent.City ='" & request("txtCity") & "', tblStudent.State ='" & request("txtState") & "', tblStudent.Zip ='" & request("txtZip") & "', tblstudent.Major ='" & request("txtMajor") & "' Where(((tblStudent.[Student Number]) ='" & tmpStr & "'))"

'Response.Write strsql

conn.Execute(strsql)

'get recordsets to show student the new changes

```
strsql="Select * from tblStudent where [Student Number] = " &
Session.Contents("sesUserID")
rs.Open strsql, conn, 2, 2
```

```
%>
```

```
<center><font face="Brush Script MT" SIZE="36" color="#ff0000"><B><i>Updated
Information</i></B></font></center>
```

```
<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a> | <a href="help.htm"
target="_blank">Help</a></center><BR>
```

```
<BR>
```

```
<BR>
```

```
<table border=0 align="center" width="500">
```

```
<TR>
```

```
<TD>
```

```
Your personal information has been changed. Please take a second to review the below
changes to make sure they are correct.
```

```
</td>
```

```
</tr>
```

```
</table><BR>
```

```
<BR>
```

```
<table border=1 align="center" width="400">
```

```
<tr>
```

```
<td width="50%"><B>First Name</B></td>
```

```
<td><%=rs("First Name")%></td>
```

```
</tr>
```

```
<tr>
```

```
<td width="50%"><B>Last Name</b></td>
```

```
<td><%=rs("Last Name")%></td>
```

```
</tr>
```

```
<tr>
```

```
<td><B>Street Number</B></td>
```

```
<td><%=rs("Street Number")%></td>
```

```
</tr>
```

```
<tr>
```

```
<td><B>Street Name</B></td>
```

```
<td><%=rs("Street Name")%></td>
```

```
</tr>
```

```
<tr>
```

```
<td><B>City</B></td>
```

```
<td><%=rs("City")%></td>
```

```
</tr>
```

```
<tr>
```

```

        <td><B>State</B></td>
        <td><%=rs("State")%></td>
    </tr>
    <tr>
        <td><B>Zip</B></td>
        <td><%=rs("Zip")%></td>
    </tr>
    <tr>
        <td><B>Major</B></td>
        <td><%=rs("Major")%></td>
    </tr>
</table>
<BR><BR>
<center>
<input type="button" name="cmdOk" value="Save Changes">
<input type="button" name="cmdCancel" value="Make Changes">
</center>
</form>
</BODY>
</HTML>

```

B.1.6 Viewclass.asp

```

<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>
<form>
<%

dim sqlstr
dim strconn
dim conn
dim strID
dim rs
dim rs2
dim tmpQuarter
dim tmpcheck

strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")

```

```
set rs2 = server.createobject("adodb.recordset")
conn.open strconn
```

```
strsql="Select x.[First Name] as first, x.[Last Name] as last, s.[Class Number], o.[Class  
Name], s.Quarter, s.Year, S.Grade from tblStudent x inner join tblStudentClassGrade s  
on x.[Student Number]=s.[Student Number] inner join tblClass o on s.[Class Number]=  
o.[Class Number] where s.[Student Number]=" & Session.Contents("sesUserID") & "  
order by Quarter, Year"
```

```
rs.Open strsql, conn, 2, 2
```

```
strsql="select distinct Year from tblStudentclassgrade where [Student Number]=" &  
Session.Contents("sesUserID")
```

```
rs2.open strsql, conn, 2, 2
```

```
If rs2.EOF or rs2.BOF then
```

```
rs2.Close
```

```
strsql="select [First name] as first, [Last Name] as last from tblstudent where [student  
number]=" & Session.Contents("sesUserID")
```

```
rs2.Open strsql, conn, 2,2
```

```
Response.Write("<center><font face=""Brush Script MT"" SIZE=""38""  
color=""#FF0000""><B><i>Degree Plan for " & rs2("first") & " " & rs2("last") &  
"</i></B></font>")
```

```
Response.Write "<center><a href=""degreeplan.asp"">Home</a> | <a  
href=""lookatinfo.asp"">View Personal Data</a> | <a href=""enterinfo.asp"">Edit  
Personal Data</a> | <a href=""viewclass.asp"">View Degree Plan</a> | <a  
href=""developdp.asp"">Add Classes</a> | <a href=""getquarter.asp"">Edit Degree  
Plan</a> | <a href=""Viewcoop.asp"">View Jobs</a> | <a href=""help.htm""  
target=""_blank"">Help</a></center><BR>"
```

```
Response.Write "<BR>"
```

```
Response.Write("<font size=""+1"">")
```

```
Response.Write("You do not currently have any classes on your degreeplan")
```

```
Response.Write("</font><BR><BR>")
```

```
Response.Write("<a href=""degreeplan.asp"">Click here to go back to the  
homepage</a><BR>")
```

```
Response.Write("<a href=""developdp.asp"">Click here to add Classes to your degree  
plan</a><BR>")
```

```
else
```

```
Response.Write("<center><font face=""Brush Script MT"" SIZE=""36""  
color=""#FF0000""><B><i>Degree Plan for " & rs("first") & " " & rs("last") &  
"</i></B></font>")
```

```
Response.Write "<center><a href=""degreeplan.asp"">Home</a> | <a  
href=""lookatinfo.asp"">View Personal Data</a> | <a href=""enterinfo.asp"">Edit  
Personal Data</a> | <a href=""viewclass.asp"">View Degree Plan</a> | <a  
href=""developdp.asp"">Add Classes</a> | <a href=""getquarter.asp"">Edit Degree  
Plan</a> | <a href=""Viewcoop.asp"">View Jobs</a> | <a href=""help.htm""  
target=""_blank"">Help</a></center><BR>"
```

```
Response.Write "<BR>"
```

```
Response.Write "<table width=""800"" align=""center"" border=""1"">"
```

```
Do until rs2.EOF
```

```
Response.Write "<tr valign=""top"">"
```

```
Response.Write "<td width=""50%"">"
```

```
Response.Write "<table border=0 width=""100%"">"
```

```
Do until rs.EOF
```

```
If rs("Quarter")="Fall" and rs("year")=rs2("year") Then
```

```
Response.Write "<tr>"
```

```
Response.Write "<td>" & rs("Class Number") & "</td>"
```

```
Response.Write "<td>" & rs("Class Name") & "</td>"
```

```
Response.Write "<td>" & rs("Quarter") & "</td>"
```

```
Response.Write "<td>" & rs("Grade") & "</td>"
```

```
Response.Write "<td>" & rs("Year") & "</td>"
```

```
Response.Write "</tr>"
```

```
End if
```

```
rs.MoveNext
```

```
loop
```

```
rs.MoveFirst
```

```
Response.Write "</table>"
```

```
Response.Write "</td>"
```

```
Response.Write "<td width=""50%"">"
```

```
Response.Write "<table border=0 width=""100%"">"
```

```
Do until rs.EOF
```

```
If rs("Quarter")="Winter" and rs("year")=rs2("year") Then
```

```
Response.Write "<tr>"
```

```
Response.Write "<td>" & rs("Class Number") & "</td>"
```

```
Response.Write "<td>" & rs("Class Name") & "</td>"
```

```
Response.Write "<td>" & rs("Quarter") & "</td>"
```

```
Response.Write "<td>" & rs("Grade") & "</td>"
```

```
Response.Write "<td>" & rs("Year") & "</td>"
```

```
Response.Write "</tr>"
```

```
End if
```

```
rs.MoveNext
```

```
loop
```

```
rs.MoveFirst
```

```
Response.Write "</table>"
```

```
Response.Write "</td>"
```

```
Response.Write "</tr>"
```

```
Response.Write "<tr valign=""top"">"
```

```
Response.Write "<td width=""50%"">"
```

```
Response.Write "<table border=0 width=""100%"">"
```

```
Do until rs.EOF
```

```
If rs("Quarter")="Spring" and rs("year")=rs2("year") Then
```

```
Response.Write "<tr>"
```

```

        Response.Write "<td>" & rs("Class Number") & "</td>"
        Response.Write "<td>" & rs("Class Name") & "</td>"
        Response.Write "<td>" & rs("Quarter")& "</td>"
        Response.Write "<td>" & rs("Grade") & "</td>"
        Response.Write "<td>" & rs("Year") & "</td>"
        Response.Write "</tr>"
    End if
    rs.MoveNext
loop
    rs.MoveFirst
    Response.Write "</table> "
Response.Write "</td>"
Response.Write "<td width=""50%"">"
    Response.Write "<table border=0 width=""100%"">"
        Do until rs.EOF
            If rs("Quarter")="Summer" and rs("year")=rs2("year") Then
                Response.Write "<tr>"
                    Response.Write "<td>" & rs("Class Number") & "</td>"
                    Response.Write "<td>" & rs("Class Name") & "</td>"
                    Response.Write "<td>" & rs("Quarter") & "</td>"
                    Response.Write "<td>" & rs("Grade") & "</td>"
                    Response.Write "<td>" & rs("Year") & "</td>"
                    Response.Write "</tr>"
                End if
                rs.MoveNext
            loop
                rs.MoveFirst
                Response.Write "</table>"
            Response.Write "</td>"
            Response.Write "</tr>"
            rs2.MoveNext
        loop
        Response.Write "</table>"

    end if
%>
</form>
</BODY>
</HTML>

```

B.1.7 Developdp.asp

```

<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">

```

```

</HEAD>
<BODY>
<script language="Vbscript">
Function ShowPre(ClassNum)

    url="prereq.asp?" & ClassNum
    window.open url

End Function

</script>
<form method="Post" action="enterclass.asp" name="frmDP">
<%

dim rs
dim rs2
dim conn
dim tmpID
dim strconn
dim strsql
dim tmpyear

tmpyear = Year(date)

'opens connection to server
strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")
set rs2 = server.createobject("adodb.recordset")
conn.open strconn

'opens recordset from tblclass
strsql="Select * from tblClass"
rs.Open strsql, conn, 2, 2

'opens up second recordset from tblLookup
strsql="Select * from tblLookup"
rs2.Open strsql, conn, 2, 2
%>
<center><font face="Brush Script MT" SIZE="36" color="#FF0000"><B><i>Degree
Plan Development Page</b></i></font></center>

```

```

<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a> | <a href="help.htm"
target="_blank">Help</a></center><BR>
<table width="70%" align="center">
<tr>
<td>
Quarter <SELECT name="selQuarter" size="1">
<option value="Fall">Fall</option>
<option value="Winter">Winter</option>
<option value="Spring">Spring</option>
<option value="Summer">Summer</option>
</SELECT>
</td>
<td>
Year <SELECT id=selYear name=selYear>
<%
for X = tmpyear - 10 to tmpyear + 10

if x = tmpyear then
Response.Write("<option value=" & x & " selected>" & x & "</option>")
else
Response.Write("<option value=" & x & ">" & x & "</option>")
end if
next
%>
</SELECT>

</td>

</tr>
<tr>
<td align="center">
<b>Class Name</b>
</td>
<td align="center">
<b>Class Number</b>
</td>
<td align="center">
<b>Audit(Checked for Yes)</b>
</td>
<td align="center">
<b>Grade</b>
</td>

```



```

</tr>
<tr>
<td align="center">
<select name="selClass1" size="1" onchange="txtClass1.value=selClass1.value">
<option selected value="----">----</option>
<% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

<% rs.movenext %>
<% loop %>
</select>
</td>
<td align="center">
<INPUT type="text" name="txtClass1" value="" readonly>
</td>
<td align="center">
<INPUT type="radio" name="rdAudit" value="1">Audit

</td>
<td>
<SELECT id=selGrades1 name=selGrades1>
<option value="">--</option>
<OPTION value="A">A</OPTION>
<OPTION value="A-">A-</OPTION>
<OPTION value="B+">B+</OPTION>
<OPTION value="B">B</OPTION>
<OPTION value="B-">B-</OPTION>
<OPTION value="C+">C+</OPTION>
<OPTION value="C">C</OPTION>
<OPTION value="C-">C-</OPTION>
<OPTION value="D+">D+</OPTION>
<OPTION value="D">D</OPTION>
<OPTION value="D-">D-</OPTION>
<OPTION value="F">F</OPTION>
</SELECT>
</td>
<td>
<input type="button" value="Prerequisite" name="cmdPreClass1"
onClick="ShowPre(frmDP.selClass1.value)">
</td>
</tr>
<% rs.MoveFirst %><% rs2.MoveFirst %>
<tr>
<td align="center">
<select name="selClass2" size="1" onchange="txtClass2.value=selClass2.value">

```

```

<option selected value="">----</option>
<% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

<% rs.movenext %>
<% loop %>
</select><br>
</td>
<td align="center">
<INPUT type="text" name="txtClass2" readonly>
</td>
<td align="center">
<INPUT type="radio" name="rdaudit" value="2">Audit

</td>
<TD>
<SELECT id=selGrades2 name=selGrades2>
<option value="">--</option>
<OPTION value="A">A</OPTION>
<OPTION value="A-">A-</OPTION>
<OPTION value="B+">B+</OPTION>
<OPTION value="B">B</OPTION>
<OPTION value="B-">B-</OPTION>
<OPTION value="C+">C+</OPTION>
<OPTION value="C">C</OPTION>
<OPTION value="C-">C-</OPTION>
<OPTION value="D+">D+</OPTION>
<OPTION value="D">D</OPTION>
<OPTION value="D-">D-</OPTION>
<OPTION value="F">F</OPTION>
</SELECT>
</td>
<td>
<input type="button" value="Prerequisite" name="cmdPreClass2"
onClick="ShowPre(frmDP.selClass2.value)">
</td>

</tr>
<% rs.MoveFirst %><% rs2.MoveFirst %>
<tr>
<td align="center">
<select name="selClass3" size="1" onChange="txtClass3.value=selClass3.value">
<option selected value="">----</option>
<% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

```

```

<% rs.movenext %>
<% loop %>
</select><br>
</td>
<td align="center">
<INPUT type="text" name="txtClass3" readonly>
</td>
<td align="center">
<INPUT type="radio" name="rdaudit" value="3">Audit

</td>
<TD>
<SELECT id=selGrades3 name=selGrades3>
<option value="">--</option>
<OPTION value="A">A</OPTION>
<OPTION value="A-">A-</OPTION>
<OPTION value="B+">B+</OPTION>
<OPTION value="B">B</OPTION>
<OPTION value="B-">B-</OPTION>
<OPTION value="C+">C+</OPTION>
<OPTION value="C">C</OPTION>
<OPTION value="C-">C-</OPTION>
<OPTION value="D+">D+</OPTION>
<OPTION value="D">D</OPTION>
<OPTION value="D-">D-</OPTION>
<OPTION value="F">F</OPTION>
</SELECT>
</td>
<td>
<input type="button" value="Prerequisite" name="cmdPreClass3"
onClick="ShowPre(frmDP.selClass3.value)">
</td>
</tr>
<% rs.MoveFirst %><% rs2.MoveFirst %>
<tr>
<td align="center">
<select name="selClass4" size="1" onchange="txtClass4.value=selClass4.value">
<option selected value="">----</option>
<% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

<% rs.movenext %>
<% loop %>
</select><br>
</td>
<td align="center">

```

```

<INPUT type="text" name="txtClass4" readonly>
</td>
<td align="center">
<INPUT type="radio" name="rdaudit" value="4">Audit

</td>
<TD>
<SELECT id=selGrades4 name=selGrades4>
  <option value="">--</option>
  <OPTION value="A">A</OPTION>
  <OPTION value="A-">A-</OPTION>
  <OPTION value="B+">B+</OPTION>
  <OPTION value="B">B</OPTION>
  <OPTION value="B-">B-</OPTION>
  <OPTION value="C+">C+</OPTION>
  <OPTION value="C">C</OPTION>
  <OPTION value="C-">C-</OPTION>
  <OPTION value="D+">D+</OPTION>
  <OPTION value="D">D</OPTION>
  <OPTION value="D-">D-</OPTION>
  <OPTION value="F">F</OPTION>
</SELECT>
</td>
<td>
<input type="button" value="Prerequisite" name="cmdPreClass4"
onClick="ShowPre(frmDP.selClass4.value)">
</td>

</tr>
<% rs.MoveFirst %><% rs2.MoveFirst %>
<tr>
<td align="center">
  <select name="selClass5" size="1" onChange="txtClass5.value=selClass5.value">
    <option selected value="">----</option>
    <% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

    <% rs.movenext %>
    <% loop %>
  </select><br>
</td>
<td align="center">
  <INPUT type="text" name="txtClass5" readonly>
</td>
<td align="center">
  <INPUT type="radio" name="rdaudit" value="5">Audit

```

```

</td>
<TD>
<SELECT id=selGrades5 name=selGrades5>
  <option value="">--</option>
  <OPTION value="A">A</OPTION>
  <OPTION value="A-">A-</OPTION>
  <OPTION value="B+">B+</OPTION>
  <OPTION value="B">B</OPTION>
  <OPTION value="B-">B-</OPTION>
  <OPTION value="C+">C+</OPTION>
  <OPTION value="C">C</OPTION>
  <OPTION value="C-">C-</OPTION>
  <OPTION value="D+">D+</OPTION>
  <OPTION value="D">D</OPTION>
  <OPTION value="D-">D-</OPTION>
  <OPTION value="F">F</OPTION>
</SELECT>
</td>
<td>
<input type="button" value="Prerequisite" name="cmdPreClass5"
onClick="ShowPre(frmDP.selClass5.value)">
</td>
</tr>
<% rs.MoveFirst %><% rs2.MoveFirst %>
<tr>
<td align="center">
  <select name="selClass6" size="1" onChange="txtClass6.value=selClass6.value">
    <option selected value="">----</option>
    <% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

    <% rs.movenext %>
  <% loop %>
  </select><br>
</td>
<td align="center">
  <INPUT type="text" name="txtClass6" readonly>
</td>
<td align="center">
  <INPUT type="radio" name="rdaudit" value="6">Audit

</td>
<TD>
<SELECT id=selGrades6 name=selGrades6>
  <option value="">--</option>

```

```

<OPTION value="A">A</OPTION>
<OPTION value="A-">A-</OPTION>
<OPTION value="B+">B+</OPTION>
<OPTION value="B">B</OPTION>
<OPTION value="B-">B-</OPTION>
<OPTION value="C+">C+</OPTION>
<OPTION value="C">C</OPTION>
<OPTION value="C-">C-</OPTION>
<OPTION value="D+">D+</OPTION>
<OPTION value="D">D</OPTION>
<OPTION value="D-">D-</OPTION>
<OPTION value="F">F</OPTION>
</SELECT>
</td>
<td>
<input type="button" value="Prerequisite" name="cmdPreClass6"
onClick="ShowPre(frmDP.selClass6.value)">
</td>
</tr>
<% rs.MoveFirst %><% rs2.MoveFirst %>
<tr>
<td align="center">
<select name="selClass7" size="1" onChange="txtClass7.value=selClass7.value">
<option selected value="">----</option>
<% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

<% rs.movenext %>
<% loop %>
</select><br>
</td>
<td align="center">
<INPUT type="text" name="txtClass7" readonly>
</td>
<td align="center">
<INPUT type="radio" name="rdaudit" value="7">Audit

</td>
<TD>
<SELECT id=selGrades7 name=selGrades7>
<option value="">--</option>
<OPTION value="A">A</OPTION>
<OPTION value="A-">A-</OPTION>
<OPTION value="B+">B+</OPTION>
<OPTION value="B">B</OPTION>
<OPTION value="B-">B-</OPTION>

```

```

        <OPTION value="C+">C+</OPTION>
        <OPTION value="C">C</OPTION>
        <OPTION value="C-">C-</OPTION>
        <OPTION value="D+">D+</OPTION>
        <OPTION value="D">D</OPTION>
        <OPTION value="D-">D-</OPTION>
        <OPTION value="F">F</OPTION>
    </SELECT>
</td>
    <td>
        <input type="button" value="Prerequisite" name="cmdPreClass7"
        onClick="ShowPre(frmDP.selClass7.value)">
    </td>
</tr>
<% rs.MoveFirst %><% rs2.MoveFirst %>
<tr>
    <td align="center">
        <select name="selClass8" size="1" onChange="txtClass8.value=selClass8.value">
            <option selected value="">----</option>
            <% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

            <% rs.movenext %>
            <% loop %>
        </select><br>
    </td>
    <td align="center">
        <INPUT type="text" name="txtClass8" readonly>
    </td>
    <td align="center">
        <INPUT type="radio" name="rdaudit" value="8">Audit

</td>
<TD>
<SELECT id=selGrades8 name=selGrades8>
    <option value="">--</option>
    <OPTION value="A">A</OPTION>
    <OPTION value="A-">A-</OPTION>
    <OPTION value="B+">B+</OPTION>
    <OPTION value="B">B</OPTION>
    <OPTION value="B-">B-</OPTION>
    <OPTION value="C+">C+</OPTION>
    <OPTION value="C">C</OPTION>
    <OPTION value="C-">C-</OPTION>
    <OPTION value="D+">D+</OPTION>
    <OPTION value="D">D</OPTION>

```

```

        <OPTION value="D-">D-</OPTION>
        <OPTION value="F">F</OPTION>
    </SELECT>
</td>
<td>
<input type="button" value="Prerequisite" name="cmdPreClass8"
onClick="ShowPre(frmDP.selClass8.value)">
</td>
</tr>
<% rs.MoveFirst %><% rs2.MoveFirst %>
<tr>
<td align="center">
    <select name="selClass9" size="1" onChange="txtClass9.value=selClass9.value">
    <option selected value="">----</option>
    <% Do While Not rs.EOF %> <option value="<% = rs("Class Number") %>"><% =
rs("Class Name") %></option>

    <% rs.movenext %>
    <% loop %>
    </select><br>
</td>
<td align="center">
    <INPUT type="text" name="txtClass9" readonly>
</td>
<td align="center">
    <INPUT type="radio" name="rdaudit" value="9">Audit

</td>
<TD>
<SELECT id=selGrades9 name=selGrades9>
    <option value="">--</option>
    <OPTION value="A">A</OPTION>
    <OPTION value="A-">A-</OPTION>
    <OPTION value="B+">B+</OPTION>
    <OPTION value="B">B</OPTION>
    <OPTION value="B-">B-</OPTION>
    <OPTION value="C+">C+</OPTION>
    <OPTION value="C">C</OPTION>
    <OPTION value="C-">C-</OPTION>
    <OPTION value="D+">D+</OPTION>
    <OPTION value="D">D</OPTION>
    <OPTION value="D-">D-</OPTION>
    <OPTION value="F">F</OPTION>
</SELECT>
</td>
<td>

```



```



```

```

</table>
<center><input type="submit" value="Enter Classes"></center>

</form>
</BODY>
</HTML>

```

B.1.8 Prereq.asp

```

<%@ Language=VBScript %>
<%
Response.Write("<HTML>")
Response.Write("<HEAD>")
Response.Write("<META NAME='\"'GENERATOR'\"' Content='\"'Microsoft Visual
Studio 6.0'\"'>")
Response.Write("</HEAD>")
Response.Write("<BODY>")

dim tmpClassname
dim strconn
dim strsql
dim rs
dim conn

tmpClassname = Request.QueryString

if instr(tmpClassname, "%") > 0 then
    tmpclassname = left(tmpClassName, instr(tmpClassname, "%") - 1) & " " &
mid(tmpclassname, instr(tmpClassname, "%") + 3, len(tmpclassname) -
instr(tmpClassname, "%") + 3)
end if

'opens connection to server
strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")
set rs2 = server.createobject("adodb.recordset")
conn.open strconn

'opens recordset from tblPrereq
strsql="Select * from tblPrereq where [Class Number]='\"' & tmpClassName & '\"'
rs.Open strsql, conn, 2, 2

if rs.EOF or rs.BOF then

```

```
Response.Write("There are no Prerequisites for this class")
```

```
else
```

```
strsql="select * from tblClass where [Class Number]=" & tmpClassName & ""  
rs2.open strsql, conn, 2, 2
```

```
Response.Write("<h2 align=""center"">Prerequisite for " & rs2("Class Name") & "</h2>")  
Response.Write("<table border=1 cellpadding=5 align=""center"">")  
Response.Write("<tr>")  
Response.Write("<th>Class Number</th>")  
Response.Write("<th>Class Name</th>")
```

```
Response.Write("</tr>")  
do until rs.EOF  
Response.Write("<tr>")  
Response.Write("<td>")  
Response.Write rs("PrereqNum")  
Response.Write("<td>")  
Response.Write rs("PrereqName")
```

```
Response.Write("</tr>")  
rs.MoveNext  
loop  
Response.Write("</table>")  
end if  
Response.Write("<BR>")  
Response.Write("<BR>")  
Response.Write("<center>")  
Response.Write("<input type=""button"" value=""Close""  
onclick=""window.close()"">")  
Response.Write("</center>")  
Response.Write("</BODY>")  
Response.Write("</HTML>")
```

```
%>
```

B.1.9 Enterclass.asp

```
<%@ Language=VBScript %>  
<HTML>  
<HEAD>  
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">  
</HEAD>
```

<BODY>

<%

```
dim rs
dim conn
dim strSQL
dim strconn
dim tmpClassName
dim strSQL1
dim ClassName(10)
dim tmpGrades(10)
Dim I
Dim counter
dim audit
```

```
ClassName(1)=request("txtClass1")
ClassName(2)=request("txtClass2")
ClassName(3)=request("txtClass3")
ClassName(4)=request("txtClass4")
ClassName(5)=request("txtClass5")
ClassName(6)=request("txtClass6")
ClassName(7)=request("txtClass7")
ClassName(8)=request("txtClass8")
ClassName(9)=request("txtClass9")
ClassName(10)=request("txtClass10")
```

```
tmpGrades(1)=request("selGrades1")
tmpGrades(2)=request("selGrades2")
tmpGrades(3)=request("selGrades3")
tmpGrades(4)=request("selGrades4")
tmpGrades(5)=request("selGrades5")
tmpGrades(6)=request("selGrades6")
tmpGrades(7)=request("selGrades7")
tmpGrades(8)=request("selGrades8")
tmpGrades(9)=request("selGrades9")
tmpGrades(10)=request("selGrades10")
```

'Get counter to see how many class entres

'Makes the connection to the server

strconn = "Driver={SQL

Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D

ATABASE=jblacksd"

```
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")
set cmd = server.CreateObject("ADODB.Command")
```

```
conn.open strconn
```

```
'use the command object to create the stored procedure parameters
```

```
cmd.ActiveConnection = conn
cmd.CommandType = 4
cmd.CommandText = "sp_InsertClass"
'3 = Int, 1=Input, 4=size in bytes
cmd.Parameters.Append(cmd.CreateParameter("@studentnumber", 200, 1, 10, ""))
cmd.Parameters.Append(cmd.CreateParameter("@classnumber", 200, 1, 10, ""))
cmd.Parameters.Append(cmd.CreateParameter("@quarter", 200, 1, 10, ""))
cmd.Parameters.Append(cmd.CreateParameter("@year", 3, 1, 4, 0))
cmd.Parameters.Append(cmd.CreateParameter("@audit", 200, 1, 10, ""))
cmd.Parameters.Append(cmd.CreateParameter("@grade", 200, 1, 2, ""))
```

```
'creates sql string to input to tblClass and executes it
```

```
For I = 1 to 10
```

```
'check to see if the current class is audit
```

```
if int(request("rdaudit")) = I then
```

```
    audit = "Y"
```

```
else
```

```
    audit = "N"
```

```
end if
```

```
If ClassName(I) <> "" then
```

```
'strsql="Insert into tblStudentClassGrade ([Student Number], [Class Number], Quarter,
Year, Grade) values ('" & Session.Contents("sesUserID") & "', '" & ClassName(I) & "', '"
& request("selQuarter") & "', '" & request("selYear") & "', '" & tmpGrades(I) & "')
```

```
'Call the stored procedure sp_InsertClass and pass the variables to be inserted
```

```
'insert parameter values
```

```
'3 = Int, 1=Input, 4=size in bytes
```

```
cmd.Parameters("@studentnumber") = Session.Contents("sesUserID")
```

```
cmd.Parameters("@classnumber") = ClassName(I)
```

```
cmd.Parameters("@quarter") = Request.Form("selQuarter")
```

```
cmd.Parameters("@year") = Request.Form("selYear")
```

```
cmd.Parameters("@audit") = Audit
```

```
cmd.Parameters("@grade") = tmpGrades(I)
```

```
cmd.Execute iAffect
```

```
x=x+1
```

end if
Next

```
%>
<center><font face="Brush Script MT" SIZE="36" color="#FF0000"><B><i>Degree
Plan Development Page</b></i></font></center><BR>
<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a> | <a href="help.htm"
target="_blank">Help</a></center><BR>
<BR>
<BR>
<center><font size="4">
<%Response.Write x & " classes have been added to " & request("selQuarter") & " of "
& request("selYear") & " quarter" %>
</center>
<BR><BR>
<table width="400" border=0 align="center">
<tr>
<Td align="left"><a href="developdp.asp">To enter more information</a>
<td align="right"><a href="degreeplan.asp">Back to the Home Page</a>
</tr>
</table>

</BODY>
</HTML>
```

B.1.10 Getquarter.asp

```
<%@ Language=VBScript %>
<html>
<head>
<meta NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</head>
<body>
<form action="editdp.asp" method="Post">
<%
dim sqlstr
dim strconn
dim conn
```

```

dim tmpID
dim tmpID2
dim strID
dim rs

strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createObject("adodb.connection")
set rs = server.createObject("adodb.recordset")
conn.open strconn

strsql="Select distinct Year from tblstudentclassgrade where [Student Number] =" &
session.contents("sesUserID")
rs.Open strsql, conn, 2, 2
%>
<center><img SRC="images/edittile.gif" WIDTH="428" HEIGHT="62"></center><br>
<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a></center>
<BR>
<br>
<center>
<h3>Please Choose a Quarter and Year to Edit:</h3>
<%
if rs.EOF then
    Response.Write "You do not have any classes in your degree plan."

else
    response.write("<select name=""selYear"" size=""1"">")
    Do While Not rs.EOF
        response.write("<option value="" & rs("Year")& """">" & rs("Year") & "</option>")
        rs.movenext
    loop
    Response.Write "</select>"
    Response.Write "<select name=""selQuarter"" size=""1"">"
    Response.Write "<option value=""Fall"">Fall</option>"
    Response.Write "<option value=""Winter"">Winter</option>"
    Response.Write "<option value=""Spring"">Spring</option>"
    Response.Write "<option value=""Summer"">Summer</option>"
    Response.Write "</select>"
    Response.Write "<input type=""Submit"" Value=""Enter""><br>"
end if
%>

```

</center>

</form>

</body>

</html>

B.1.11 Editdb.asp

<%@ Language=VBScript %>

<html>

<head>

<meta NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">

</head>

<body>

<form action="updateclass.asp" method="Post">

<%

dim sqlstr

dim strconn

dim conn

dim tmpQuarter

dim tmpYear

dim strID

dim rs

dim counter

dim tmptxtName

dim tmpselYear

counter=0

strconn = "Driver={SQL

Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D

ATABASE=jblacksd"

set conn = server.createobject("adodb.connection")

set rs = server.createobject("adodb.recordset")

conn.open strconn

tmpquarter=Request("selQuarter")

tmpYear=request("selYear")

tmpselyear = Year(date)

strsql="select s.[Student Number], s.[Class Number], c.[Class Name], s.grade from
tblstudentclassgrade s inner join tblClass c on s.[Class Number]=c.[Class Number] where
s.[Student Number]=" & session.Contents("sesUserID") & " and Year=" & tmpYear & "
and Quarter=" & tmpQuarter & ""

rs.Open strsql, conn, 2, 2


```

%>
<center><img SRC="images/edittile.gif" WIDTH="428" HEIGHT="62"></center><br>
<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a></center>
<BR>
<BR>
<table align="center">
<tr>
<th> Check to Remove Class</th>
<th>Current Class Number</th>
<th>Current Class Name</th>
<th>Grade</th>
<th>Quarter</th>
<th>Year</th>
</tr>
<%
if rs.EOF then

Response.Write "<TR><TD colspan='6' align='center'><BR><BR><h2>You do
not have any classes for this quarter</h2><BR>"
end if
%>
<%Do Until rs.EOF%>
<tr>
<td align="center">
<%counter=counter +1%>
<%tmpName="chkRemove" & counter%>
<%session.Contents("sesCounter")=counter%>
<%Response.Write("<INPUT type='checkbox' name=''" & tmpName & "'">")%>
</td>
<td>
<%tmptxtName= "txtClassNumber" & counter%>
<input value="<%=rs("Class Number")%>" name="<%Response.Write tmptxtName%>"
readonly><br>
</td>
<td>
<%tmptxtName= "txtClassName" & counter%>
<input value="<%=rs("Class Name")%>" name="<%Response.write tmptxtName%>"
readonly><br>
</td>
<td>
<%

```

```
response.write("<SELECT name='\"selGrade\"' & counter & '\"\"\"\">")
select case rs("grade")
```

```
case "A"
```

```
Response.Write("<option value='\"\"\"\">--</option>")
Response.Write("<OPTION value='\"A\"' selected>A</OPTION>")
Response.Write("<OPTION value='\"A-\"'>A-</OPTION>")
Response.Write("<OPTION value='\"B+\"'>B+</OPTION>")
Response.Write("<OPTION value='\"B\"'>B</OPTION>")
Response.Write("<OPTION value='\"B-\"'>B-</OPTION>")
Response.Write("<OPTION value='\"C+\"'>C+</OPTION>")
Response.Write("<OPTION value='\"C\"'>C</OPTION>")
Response.Write("<OPTION value='\"C-\"'>C-</OPTION>")
Response.Write("<OPTION value='\"D+\"'>D+</OPTION>")
Response.Write("<OPTION value='\"D\"'>D</OPTION>")
Response.Write("<OPTION value='\"D-\"'>D-</OPTION>")
Response.Write("<OPTION value='\"F\"'>F</OPTION>")
```

```
case "A-"
```

```
Response.Write("<option value='\"\"\"\">--</option>")
Response.Write("<OPTION value='\"A\"'>A</OPTION>")
Response.Write("<OPTION value='\"A-\"' selected>A-</OPTION>")
Response.Write("<OPTION value='\"B+\"'>B+</OPTION>")
Response.Write("<OPTION value='\"B\"'>B</OPTION>")
Response.Write("<OPTION value='\"B-\"'>B-</OPTION>")
Response.Write("<OPTION value='\"C+\"'>C+</OPTION>")
Response.Write("<OPTION value='\"C\"'>C</OPTION>")
Response.Write("<OPTION value='\"C-\"'>C-</OPTION>")
Response.Write("<OPTION value='\"D+\"'>D+</OPTION>")
Response.Write("<OPTION value='\"D\"'>D</OPTION>")
Response.Write("<OPTION value='\"D-\"'>D-</OPTION>")
Response.Write("<OPTION value='\"F\"'>F</OPTION>")
```

```
case "B+"
```

```
Response.Write("<option value='\"\"\"\">--</option>")
Response.Write("<OPTION value='\"A\"'>A</OPTION>")
Response.Write("<OPTION value='\"A-\"'>A-</OPTION>")
Response.Write("<OPTION value='\"B+\"' selected>B+</OPTION>")
Response.Write("<OPTION value='\"B\"'>B</OPTION>")
Response.Write("<OPTION value='\"B-\"'>B-</OPTION>")
Response.Write("<OPTION value='\"C+\"'>C+</OPTION>")
Response.Write("<OPTION value='\"C\"'>C</OPTION>")
Response.Write("<OPTION value='\"C-\"'>C-</OPTION>")
Response.Write("<OPTION value='\"D+\"'>D+</OPTION>")
Response.Write("<OPTION value='\"D\"'>D</OPTION>")
Response.Write("<OPTION value='\"D-\"'>D-</OPTION>")
Response.Write("<OPTION value='\"F\"'>F</OPTION>")
```

```
case "B"
```

[illegible]

```
Response.Write("<OPTION value=""B"">B</OPTION>")
Response.Write("<OPTION value=""B-"">B-</OPTION>")
Response.Write("<OPTION value=""C+"">C+</OPTION>")
Response.Write("<OPTION value=""C"" selected>C</OPTION>")
Response.Write("<OPTION value=""C-"">C-</OPTION>")
Response.Write("<OPTION value=""D+"">D+</OPTION>")
Response.Write("<OPTION value=""D"">D</OPTION>")
Response.Write("<OPTION value=""D-"">D-</OPTION>")
Response.Write("<OPTION value=""F"">F</OPTION>")
case "C-"
Response.Write("<option value="">--</option>")
Response.Write("<OPTION value=""A"">A</OPTION>")
Response.Write("<OPTION value=""A-"">A-</OPTION>")
Response.Write("<OPTION value=""B+"">B+</OPTION>")
Response.Write("<OPTION value=""B"">B</OPTION>")
Response.Write("<OPTION value=""B-"">B-</OPTION>")
Response.Write("<OPTION value=""C+"">C+</OPTION>")
Response.Write("<OPTION value=""C"">C</OPTION>")
Response.Write("<OPTION value=""C-"" selected>C-</OPTION>")
Response.Write("<OPTION value=""D+"">D+</OPTION>")
Response.Write("<OPTION value=""D"">D</OPTION>")
Response.Write("<OPTION value=""D-"">D-</OPTION>")
Response.Write("<OPTION value=""F"">F</OPTION>")
case "D+"
Response.Write("<option value="">--</option>")
Response.Write("<OPTION value=""A"">A</OPTION>")
Response.Write("<OPTION value=""A-"">A-</OPTION>")
Response.Write("<OPTION value=""B+"">B+</OPTION>")
Response.Write("<OPTION value=""B"">B</OPTION>")
Response.Write("<OPTION value=""B-"">B-</OPTION>")
Response.Write("<OPTION value=""C+"">C+</OPTION>")
Response.Write("<OPTION value=""C"">C</OPTION>")
Response.Write("<OPTION value=""C-"">C-</OPTION>")
Response.Write("<OPTION value=""D+"" selected>D+</OPTION>")
Response.Write("<OPTION value=""D"">D</OPTION>")
Response.Write("<OPTION value=""D-"">D-</OPTION>")
Response.Write("<OPTION value=""F"">F</OPTION>")
case "D"
```

[illegible]

```

Response.Write("<OPTION value=""F"">F</OPTION>")

end select
Response.Write("</SELECT>")
%>
</td>
<td>
<SELECT name="selQuarter<%Response.Write counter%>" size="1"><BR>
<%
select case tmpquarter
case "Fall"
Response.Write("<option value=""Fall"">Fall</option>")
Response.Write("<option value=""Winter"">Winter</option>")
Response.Write("<option value=""Spring"">Spring</option>")
Response.Write("<option value=""Summer"">Summer</option>")
case "Winter"
Response.Write("<option value=""Fall"">Fall</option>")
Response.Write("<option value=""Winter"" selected>Winter</option>")
Response.Write("<option value=""Spring"">Spring</option>")
Response.Write("<option value=""Summer"">Summer</option>")
case "Spring"
Response.Write("<option value=""Fall"">Fall</option>")
Response.Write("<option value=""Winter"">Winter</option>")
Response.Write("<option value=""Spring"" selected>Spring</option>")
Response.Write("<option value=""Summer"">Summer</option>")
case "Summer"
Response.Write("<option value=""Fall"">Fall</option>")
Response.Write("<option value=""Winter"">Winter</option>")
Response.Write("<option value=""Spring"">Spring</option>")
Response.Write("<option value=""Summer"" selected>Summer</option>")
end select
%>

</SELECT>
</td>
<td>
<SELECT name="selYear<%Response.Write counter%>">
<%
for X = tmpselyear - 10 to tmpselyear + 10
tmpyear = int(tmpyear)
if x = tmpyear then
Response.Write("<option value="" & x & "" selected>" & x & "</option>")
else
Response.Write("<option value="" & x & "">" & x & "</option>")
end if
next

```

```

%>
</SELECT>
</td>
</tr>
<%rs.MoveNext%>
<%loop%>
</table>
<center><input type="submit" value="Submit Changes"><input type="reset"
value="Clear Changes"></center>
<input type="hidden" name="hidyear" value="<%Response.write tmpyear%>">
<input type="hidden" name="hidquarter" value="<%Response.write tmpquarter%>">
</form>
</body>
</html>

```

B.1.12 Updateclass.asp

```

<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>
<FORM>
<%
dim sqlstr
dim strconn
dim conn
dim tmpchkRemove(20)
dim tmpClassNumber(20)
dim tmpSelQuarter(20)
dim tmpselyear(20)
dim tmpgrade(20)
dim tmpYear(20)
dim strID
dim rs

strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")
set cmd = server.CreateObject("ADODB.Command")
conn.open strconn

tmpClassNumber(1)=request("txtClassNumber1")

```

tmpClassNumber(2)=request("txtClassNumber2")
tmpClassNumber(3)=request("txtClassNumber3")
tmpClassNumber(4)=request("txtClassNumber4")
tmpClassNumber(5)=request("txtClassNumber5")
tmpClassNumber(6)=request("txtClassNumber6")
tmpClassNumber(7)=request("txtClassNumber7")
tmpClassNumber(8)=request("txtClassNumber8")
tmpClassNumber(9)=request("txtClassNumber9")
tmpClassNumber(10)=request("txtClassNumber10")
tmpClassNumber(11)=request("txtClassNumber11")
tmpClassNumber(12)=request("txtClassNumber12")
tmpClassNumber(13)=request("txtClassNumber13")
tmpClassNumber(14)=request("txtClassNumber14")
tmpClassNumber(15)=request("txtClassNumber15")
tmpClassNumber(16)=request("txtClassNumber16")
tmpClassNumber(17)=request("txtClassNumber17")
tmpClassNumber(18)=request("txtClassNumber18")
tmpClassNumber(19)=request("txtClassNumber19")
tmpClassNumber(20)=request("txtClassNumber20")

tmpchkRemove(1)=request("chkRemove1")
tmpchkRemove(2)=request("chkRemove2")
tmpchkRemove(3)=request("chkRemove3")
tmpchkRemove(4)=request("chkRemove4")
tmpchkRemove(5)=request("chkRemove5")
tmpchkRemove(6)=request("chkRemove6")
tmpchkRemove(7)=request("chkRemove7")
tmpchkRemove(8)=request("chkRemove8")
tmpchkRemove(9)=request("chkRemove9")
tmpchkRemove(10)=request("chkRemove10")
tmpchkRemove(11)=request("chkRemove11")
tmpchkRemove(12)=request("chkRemove12")
tmpchkRemove(13)=request("chkRemove13")
tmpchkRemove(14)=request("chkRemove14")
tmpchkRemove(15)=request("chkRemove15")
tmpchkRemove(16)=request("chkRemove16")
tmpchkRemove(17)=request("chkRemove17")
tmpchkRemove(18)=request("chkRemove18")
tmpchkRemove(19)=request("chkRemove19")
tmpchkRemove(20)=request("chkRemove20")

tmpselquarter(1)=request("selQuarter1")
tmpselquarter(2)=request("selQuarter2")
tmpselquarter(3)=request("selQuarter3")
tmpselquarter(4)=request("selQuarter4")
tmpselquarter(5)=request("selQuarter5")

tmpselquarter(6)=request("selQuarter6")
tmpselquarter(7)=request("selQuarter7")
tmpselquarter(8)=request("selQuarter8")
tmpselquarter(9)=request("selQuarter9")
tmpselquarter(10)=request("selQuarter10")
tmpselquarter(11)=request("selQuarter11")
tmpselquarter(12)=request("selQuarter12")
tmpselquarter(13)=request("selQuarter13")
tmpselquarter(14)=request("selQuarter14")
tmpselquarter(15)=request("selQuarter15")
tmpselquarter(16)=request("selQuarter16")
tmpselquarter(17)=request("selQuarter17")
tmpselquarter(18)=request("selQuarter18")
tmpselquarter(19)=request("selQuarter19")
tmpselquarter(20)=request("selQuarter20")

tmpselyear(1)=request("selyear1")
tmpselyear(2)=request("selyear2")
tmpselyear(3)=request("selyear3")
tmpselyear(4)=request("selyear4")
tmpselyear(5)=request("selyear5")
tmpselyear(6)=request("selyear6")
tmpselyear(7)=request("selyear7")
tmpselyear(8)=request("selyear8")
tmpselyear(9)=request("selyear9")
tmpselyear(10)=request("selyear10")
tmpselyear(11)=request("selyear11")
tmpselyear(12)=request("selyear12")
tmpselyear(13)=request("selyear13")
tmpselyear(14)=request("selyear14")
tmpselyear(15)=request("selyear15")
tmpselyear(16)=request("selyear16")
tmpselyear(17)=request("selyear17")
tmpselyear(18)=request("selyear18")
tmpselyear(19)=request("selyear19")
tmpselyear(20)=request("selyear20")

tmpgrade(1)=request("selGrade1")
tmpgrade(2)=request("selGrade2")
tmpgrade(3)=request("selGrade3")
tmpgrade(4)=request("selGrade4")
tmpgrade(5)=request("selGrade5")
tmpgrade(6)=request("selGrade6")
tmpgrade(7)=request("selGrade7")
tmpgrade(8)=request("selGrade8")
tmpgrade(9)=request("selGrade9")

```

tmpgrade(10)=request("selGrade10")
tmpgrade(11)=request("selGrade11")
tmpgrade(12)=request("selGrade12")
tmpgrade(13)=request("selGrade13")
tmpgrade(14)=request("selGrade14")
tmpgrade(15)=request("selGrade15")
tmpgrade(16)=request("selGrade16")
tmpgrade(17)=request("selGrade17")
tmpgrade(18)=request("selGrade18")
tmpgrade(19)=request("selGrade19")
tmpgrade(20)=request("selGrade20")

```

'use the command object to create the stored procedure parameters

```

cmd.ActiveConnection = conn
cmd.CommandType = 4
cmd.CommandText = "sp_UpdateClass"
'3 = Int, 1=Input, 4=size in bytes
cmd.Parameters.Append(cmd.CreateParameter("@studentnumber", 200, 1, 10, ""))
cmd.Parameters.Append(cmd.CreateParameter("@classnumber", 200, 1, 10, ""))
cmd.Parameters.Append(cmd.CreateParameter("@quarter", 200, 1, 10, ""))
cmd.Parameters.Append(cmd.CreateParameter("@year", 3, 1, 4, 0))
cmd.Parameters.Append(cmd.CreateParameter("@grade", 200, 1, 2, ""))
cmd.Parameters.Append(cmd.CreateParameter("@quarter2", 200, 1, 10, ""))
cmd.Parameters.Append(cmd.CreateParameter("@year2", 3, 1, 4, 0))

```

for I= 1 to 20

if tmpchkRemove(I) = "on" then

```

'checks to see if class is marked for deletion. if so do it!
strsql="Delete from tblStudentClassGrade where [Student Number]= " &
session.Contents("sesUserID") & " and [Class Number] = " & tmpClassNumber(I)& "
and Quarter= " & request("hidquarter") & " and Year=" & Request("hidYear") & ""
conn.Execute strsql
'Response.Write strsql
'Response.Write "<BR>"

```

else

```

'checks to see if there is a value for classnumber...if so update
if tmpclassnumber(I) <> "" then
'Call the stored procedure sp_InsertClass and pass the variables to be inserted
'insert parameter values
'3 = Int, 1=Input, 4=size in bytes
cmd.Parameters("@studentnumber") = Session.Contents("sesUserID")
cmd.Parameters("@classnumber")= tmpClassNumber(I)
cmd.Parameters("@quarter") = tmpselquarter(I)
cmd.Parameters("@year") = tmpselYear(I)

```

```

cmd.Parameters("@grade") = tmpGrade(I)
cmd.Parameters("@quarter2") = request("hidquarter")
cmd.Parameters("@year2") = Request("hidYear")
cmd.Execute iAffect
' Response.Write request("hidquarter") & "<BR>"
' Response.Write Request("hidYear") & "<BR>"
' Response.Write "make change<BR>"
end if
end if

next
%>
<center><img SRC="images/edittile.gif" WIDTH="428" HEIGHT="62"></center><br>
<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a></center>
<BR>
<BR>
<table align="center">
<tr>
<td>
<h3 align="center">Your Degree Plan has been update for <%Response.Write
request("hidquarter") & " " & request("hidyear")%> </h3>
<BR>
<font size="4">
<a href="getquarter.asp">Edit another quarter</a><BR>
<a href="degreeplan.asp">Back Home</a>
</font>
</td>
</tr>
</table>
</FORM>

</BODY>
</HTML>

```

B.1.13 Viewcoop.asp

```

<%@ Language=VBScript %>
<%
response.write "<HTML>"
response.write "<HEAD>"

response.write "</HEAD>"

```

```
response.write "<BODY>"
```

```
dim sqlstr  
dim strconn  
dim conn  
dim strID  
dim rs  
dim tmpMajor
```

```
strconn = "Driver={SQL  
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D  
ATABASE=jblacksd"  
set conn = server.createobject("adodb.connection")  
set rs = server.createobject("adodb.recordset")  
set rs2 = server.createobject("adodb.recordset")  
conn.open strconn  
'Get the major of the current student  
strsql="Select Major from tblstudent where [Student Number] = " &  
Session.Contents("sesUserID")  
rs.Open strsql, conn, 2, 2
```

```
if isnull(rs("Major")) then
```

```
Response.Write "<a href='\"'enterinfo.asp\"'>Please specify a major on the Personal  
Information Page</a>"
```

```
else
```

```
'Use the major and get available jobs for the Major  
strsql = "Select * from tblJobAvailable where Major = '" & rs("Major") & "' and  
Available= 'Yes'"  
rs2.Open strsql, conn, 2, 2
```

```
Response.Write "<center><font face='\"'Brush Script MT\"' SIZE='\"'36\"'  
color='\"'#FF0000\"'><B><i>Available Jobs for " & rs2("Major") &  
"Students</i></B></font>"  
Response.Write "<center><a href='\"'degreeplan.asp\"'>Home</a> | <a  
href='\"'lookatinfo.asp\"'>View Personal Data</a> | <a href='\"'enterinfo.asp\"'>Edit  
Personal Data</a> | <a href='\"'viewclass.asp\"'>View Degree Plan</a> | <a  
href='\"'developdp.asp\"'>Add Classes</a> | <a href='\"'getquarter.asp\"'>Edit Degree  
Plan</a> | <a href='\"'Viewcoop.asp\"'>View Jobs</a> | <a href='\"'help.htm\"'  
target='\"'_blank\"'>Help</a></center><BR>"  
Response.Write "<BR><BR><BR>"  
Response.Write "<table border=1 width='\"'450\"' align='\"'center\"' cellpadding='\"'10\"'>"  
Do until rs2.EOF  
Response.Write "<tr>"
```

```

Response.Write "<td valign=""top"">"
Response.Write "<h2 align=""center"">" & rs2("Company Name") & "</h2>"

Response.Write "<h3>Position:</h3>"
Response.Write "<font size=""+1""><B>" & rs2("Job Position") &
"</b></font>&nbsp;&nbsp;&nbsp;"
Response.write rs2("Job Description")
Response.Write "<BR><BR>"
Response.Write "<h3>Skills Needed:</h3>"
Response.write rs2("Skills Needed")
Response.Write "<BR><BR>"
Response.Write "<B>Available:</B>" & rs2("Date Available")
Response.Write "<BR><BR><BR></td></tr>"
rs2.MoveNext
loop
Response.Write "</table>"
Response.Write "<BR>"
Response.Write "<a href=""degreeplan.asp"">Back to home</a>"

Response.Write "</BODY>"
Response.Write "</HTML>"

end if
%>

```

B.1.14 Degreeplan.asp

```

<%@ Language=VBScript %>
<html>
<head>
<meta NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<script LANGUAGE="JavaScript">
<!--
function imgover(imgname){
imgname.src = "arrow.gif"
}
function imgout(imgname){
imgname.src = "blank.gif"
}
// End -->
</script>
</head>
<body>
<%
dim conn
dim rs

```

```

dim strconn
dim strsql

set conn=server.CreateObject("adodb.connection")
set rs=server.CreateObject("adodb.recordset")
strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
conn.Open strconn

strsql="Select * from tblStudent where [Student Number] = " &
Session.Contents("sesUserId")
rs.Open strsql, conn, 2, 2

%>
<center><img SRC="images/logo.gif" WIDTH="375" HEIGHT="63"></center><BR>
<center><a href="degreeplan.asp">Home</a> | <a href="lookatinfo.asp">View Personal
Data</a> | <a href="enterinfo.asp">Edit Personal Data</a> | <a
href="viewclass.asp">View Degree Plan</a> | <a href="developdp.asp">Add
Classes</a> | <a href="getquarter.asp">Edit Degree Plan</a> | <a
href="Viewcoop.asp">View Jobs</a> | <a href="help.htm"
target="_blank">Help</a></center><BR>
<BR>
<h3 align="center">Welcome <%=rs("First Name")%>&nbsp;<%=rs("Last
Name")%></h3>
<table border=0 align="center">
<tr><td align="center">
<font size="+1">Please press one of the links below:
</td>
<tr>
<td><font size="+1">
<%
response.write("<img SRC=""blank.gif"" name=""pic0"" WIDTH=""10""
HEIGHT=""10""><a href=""lookatinfo.asp"" onMouseOver=""imgover(pic0)""
onMouseOut=""imgout(pic0)"">Personal information</a> View your personal
information<BR>")
response.write("<img SRC=""lank.gif"" name=""pic1"" WIDTH=""10""
HEIGHT=""10""><a href=""enterinfo.asp"" onMouseOver=""imgover(pic1)""
onMouseOut=""imgout(pic1)"">Update Personal information</a> Change your home
address and major<BR>")
response.write("<img SRC=""blank.gif"" name=""pic2"" WIDTH=""10""
HEIGHT=""10""><a href=""viewclass.asp"" onMouseOver=""imgover(pic2)""
onMouseOut=""imgout(pic2)"">View degree plan</a> Look over your whole degree
plan<BR>")
response.write("<img SRC=""blank.gif"" name=""pic3"" WIDTH=""10""
HEIGHT=""10""><a href=""developdp.asp"" onMouseOver=""imgover(pic3)""

```

```

onMouseOut=""imgout(pic3)"">Develop your degree plan</a> Add classes to your
degree plan<BR>")
response.write("<img SRC=""blank.gif"" name=""pic5"" WIDTH=""10""
HEIGHT=""10""><a href=""getquarter.asp"" onMouseOver=""imgover(pic5)""
onMouseOut=""imgout(pic5)"">Edit Your Degree Plan</a> Edit and update your current
degree plan<BR>")
response.write("<img SRC=""blank.gif"" name=""pic4"" WIDTH=""10""
HEIGHT=""10""><a href=""Viewcoop.asp"" onMouseOver=""imgover(pic4)""
onMouseOut=""imgout(pic4)"">Available jobs</a> View current jobs that are available
for your major<BR>")
Response.Write("<BR><BR></td></tr>")
Response.Write("<tr><td align=""center"">")
Response.Write "<font size=""4"">"
if rs("Approved") = 0 then
    Response.Write "Your degree plan has <B>not been approved</B> by your advisor"
else
    Response.Write "Your degree plan has <B>been approved</B> by your advisor"
end if
Response.Write("</td></tr>")
Response.Write "</table>"

%>
</font>
</body>
</html>

```

B.2 Teacher Web Front-End Code

B.2.1 Receiveteacherlogin.asp

```

<%@ Language=VBScript %>
<%
dim sqlstr
dim strconn
dim conn
dim tmpID
dim tmpID2
dim strID
dim rs

'open connection
strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")

```

```

set rs2 = server.createobject("adodb.recordset")
conn.open strconn

strsql="Select * from tblTeacher where UserName='" & request("txtUserName") & "' and
TPassword='" & request("txtPassword") & "'"
rs.Open strsql, conn, 2, 2

if rs.BOF then

    Response.Write("<a href='\"\"login.htm\"\">Sorry, wrong Teacher Login. Please Click here
to login again.</a>")
    session.Contents("sesTeacher")=0
    Response.Write "<BR>"
    Response.Write strsql
else
    tmpRange1=ucase(rs("Student"))
    tmprange2=lcase(rs("Student"))
    strsql="Select * from tblStudent where [Last Name] like '[' & tmpRange1 & tmprange2
& ']'%' order by [Last name], [first name]"
    rs2.Open strsql, conn,2,2

    if rs2.BOF then

        Response.Write("<a href='\"\"login.htm\"\">Sorry, there are no students for this
login.</a>")
        session.Contents("sesTeacher")=0
        Response.Write "<BR>"
    else

        Response.Write "<HTML>"
        Response.Write "<HEAD>"
        Response.Write "</HEAD>"
        Response.Write "<BODY>"
        Response.Write "<form action='\"\"viewstudentclass.asp\"\" method='\"\"post\"\"
name='\"\"frmTeach\"\">"
        Response.Write("<center><img src='\"\"images\logo.gif\"\"></center><BR>")
        Response.Write("<h3>")
        Response.write("Welcome ")
        response.write rs("TName")
        Response.Write("</h3>")
        Response.Write("Please Pick a Name to view their Degree Plan<BR>")
        Response.Write("<SELECT id=sclStudent name=sclStudent>")

        rs2.MoveFirst
        do until rs2.eof

```



```

        Response.Write("<OPTION value=""" & rs2("Student Number") & """">" &
rs2("Last Name") & ", " & rs2("first Name") & "</OPTION>")
        rs2.MoveNext
    loop
    Response.Write "</SELECT>"
    Response.Write "<input type="""submit"" value="""Enter"">"
    Response.Write "<input type="""hidden"" value=""" & request("txtUserName") & """"
name="""hidteacher"">"
    Response.Write "</form>"
    Response.Write "</BODY>"
    Response.Write "</HTML>"
end if
end if
%>

```

B.2.2 Viewstudentclass.asp

```

<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>
<form method=post action="teacher.asp">
<%

dim sqlstr
dim strconn
dim conn
dim strID
dim rs
dim rs2
dim tmpQuarter
dim tmpcheck

strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createobject("adodb.connection")
set rs = server.createobject("adodb.recordset")
set rs2 = server.createobject("adodb.recordset")
set rs3 = server.createobject("adodb.recordset")
conn.open strconn

```

```

strsql="Select s.[Class Number], o.[Class Name], s.Quarter, s.Year, S.Grade from
tblStudentClassGrade s inner join tblClass o on s.[Class Number]= o.[Class Number]
where s.[Student Number]=" & request("selStudent") & " order by Quarter, Year"
rs.Open strsql, conn, 2, 2
strsql="select distinct Year from tblStudentclassgrade where [Student Number]=" &
request("selStudent")
rs2.open strsql, conn, 2, 2
strsql="select [First name], [Last Name], Approved from tblStudent where [Student
Number]=" & request("selStudent")
rs3.open strsql, conn, 2, 2
If rs2.EOF or rs2.BOF then
    Response.Write("<center><font face='\"Brush Script MT\"' SIZE='\"36\"'
color='\"#FF0000\"'><B><i>Degree Plan for " & rs3("First Name") & " " & rs3("Last
Name") & "</b></i></font></center><BR>")
    response.write "<BR>"
    Response.Write("<center><font size='\"4\"'>")
    Response.Write("There is currently no classes for this degree plan")
    Response.Write("</font><BR>")
    Response.Write("<a href='\"teacher.asp\"'>Click here to go back to select another
student</a></center><BR><BR><BR>")
else
    Response.Write("<center><font face='\"Brush Script MT\"' SIZE='\"36\"'
color='\"#FF0000\"'><B><i>Degree Plan for " & rs3("First Name") & " " & rs3("Last
Name") & "</b></i></font></center><BR>")
end if

%>

```

```

<table width="70%" align="center" border="1">
<%Do until rs2.EOF%>
<tr valign="top">
<td width="50%">
    <table border=0 width="100%">
        <%Do until rs.EOF%>
            <%If rs("Quarter")="Fall" and rs("year")=rs2("year") Then%>
                <tr>
                    <td><%=rs("Class Number")%></td>
                    <td><%=rs("Class Name")%></td>
                    <td><%=rs("Quarter")%></td>
                    <td><%=rs("Grade")%></td>
                    <td><%=rs("Year")%></td>
                </tr>
            <%End if%>
            <%rs.MoveNext%>
        <%loop%>
    </table>

```

```

        <%rs.MoveFirst%>
    </table>
</td>
<td width="50%">
    <table border=0 width="100%">
        <%Do until rs.EOF%>
        <%If rs("Quarter")="Winter" and rs("year")=rs2("year") Then%>
            <tr>
                <td><%=rs("Class Number")%></td>
                <td><%=rs("Class Name")%></td>
                <td><%=rs("Grade")%></td>
                <td><%=rs("Quarter")%></td>
                <td><%=rs("Year")%></td>
            </tr>
            <%End if%>
            <%rs.MoveNext%>
        <%loop%>
        <%rs.MoveFirst%>
    </table>
</td>
</tr>
<tr valign="top">
<td width="50%">
    <table border=0 width="100%">
        <%Do until rs.EOF%>
        <%If rs("Quarter")="Spring" and rs("year")=rs2("year") Then%>
            <tr>
                <td><%=rs("Class Number")%></td>
                <td><%=rs("Class Name")%></td>
                <td><%=rs("Grade")%></td>
                <td><%=rs("Quarter")%></td>
                <td><%=rs("Year")%></td>
            </tr>
            <%End if%>
            <%rs.MoveNext%>
        <%loop%>
        <%rs.MoveFirst%>
    </table>
</td>
<td width="50%">
    <table border=0 width="100%">
        <%Do until rs.EOF%>
        <%If rs("Quarter")="Summer" and rs("year")=rs2("year") Then%>
            <tr>
                <td><%=rs("Class Number")%></td>
                <td><%=rs("Class Name")%></td>

```

```

        <td><%=rs("Grade")%></td>
        <td><%=rs("Quarter")%></td>
        <td><%=rs("Year")%></td>
    </tr>
    <%End if%>
    <%rs.MoveNext%>
    <%loop%>
    <%rs.MoveFirst%>
</table>
</td>
</tr>
<%rs2.MoveNext%>
<%loop%>
</table>
<center>
<SELECT id=selApproved name=selApproved>
<%
Select Case rs3("Approved")
Case 1
    Response.Write "<Option value='\"'Approved\"'\" selected>Approved</option>"
    Response.Write "<option value='\"'Not\"'\">Not Approved</option>"
Case else
    Response.Write "<Option value='\"'Approved\"'\">Approved</option>"
    Response.Write "<option value='\"'Not\"'\" selected>Not Approved</option>"
end select
%>
</SELECT>
<input type="submit" value="Submit " id=submit1 name=submit1>
</center>

<input type="hidden" name="studentnumber" value="<%Response.Write
request("SelStudent")%>">
<input type="hidden" name="hidTeacher" value="<%Response.Write
request("hidTeacher")%>">
</form>
</BODY>
</HTML>

```

B.2.3 Teacher.asp

```

<%@ Language=VBScript %>
<%

```

```

dim sqlstr
dim strconn

```

```
dim conn
dim tmpRange1
dim tmpRange2
dim strID
dim rs
dim tmpteacher
dim tmpstudentnum
```

```
'Get information from Login.asp(the userID) and places it into a session.contents variable
tmpTeacher=request("hidteacher")
tmpstudentnum=request("studentnumber")
```

```
'open connection
strconn = "Driver={SQL
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D
ATABASE=jblacksd"
set conn = server.createObject("adodb.connection")
set conn2 = server.createObject("adodb.connection")
set rs = server.createObject("adodb.recordset")
set rs2 = server.createObject("adodb.recordset")
conn.open strconn
conn2.open strconn
```

```
if request("selApproved")= "Approved" then
    strUpdate = "update tblStudent set Approved = 1 where [student number] = '" &
tmpstudentnum & "'"
    conn2.Execute strUpdate
```

```
else
    strUpdate = "update tblStudent set Approved = 0 where [student number] = '" &
tmpstudentnum & "'"
    conn2.Execute strUpdate
```

```
end if
```

```
strsql="Select * from tblTeacher where UserName='" & tmpteacher & "'"
rs.Open strsql, conn, 2, 2
If rs.EOF then
Response.Write "string 1: " & strsql & "<BR>"
response.write "request object: " & tmpteacher
else
```

```

tmpRange1=ucase(rs("Student")&"")
'if tmpRange1 = "" Then
                                'Response.Write "EMPTY"
                                'Response.Flush
' End If
tmprange2=lcase(rs("Student"))

    strSQL="Select * from tblStudent where [Last Name] like '[' & tmpRange1 & tmprange2
& ']'%' order by [Last name], [first name]"
    rs2.Open strSQL, conn,2,2
    if rs2.BOF then
        Response.Write("<a href='\"'login.htm\"'>Sorry, there are no students for this
login.</a>")
        session.Contents("sesTeacher")=0
        Response.Write "<BR>"
    else
        Response.Write "<HTML>"
        Response.Write "<HEAD>"
        Response.Write "</HEAD>"
        Response.Write "<BODY>"
        Response.Write "<form action='\"'viewstudentclass.asp\"' method='\"'post\"'
name='\"'frmTeach\"'>"
        Response.Write("<center><img src='\"'images\logo.gif\"'></center><BR>")
        Response.Write("<h3>")
        Response.write("Welcome ")
        response.write rs("TName")
        Response.Write("</h3>")
        Response.Write("Please Pick a Name to view their Degree Plan<BR>")
        Response.Write("<SELECT id=selStudent name=selStudent>")
        rs2.MoveFirst
        do until rs2.eof
            Response.Write("<OPTION value='\"' & rs2('Student Number') & '\"'>" &
rs2("Last Name") & ", " & rs2("first Name") & "</OPTION>")
            rs2.MoveNext
        loop
        Response.Write "</SELECT>"
        Response.Write "<input type='\"'submit\"' value='\"'Enter\"'>"
        Response.Write "<input type='\"'hidden\"' value='\"' & request('hidteacher') & '\"'
name='\"'hidteacher\"'>"
        Response.Write "</form>"
        Response.Write "</BODY>"
        Response.Write "</HTML>"
    end if
end if
%>

```

Appendix C

C.1 Access Front-End Form Code

C.1.1 FrmAddCoopQuarter

```
Private Sub Form_Load()
```

```
On Error GoTo Err_cmdCancel_Click
```

```
Dim conn
```

```
Dim rs
```

```
Dim strconn
```

```
Dim strsql
```

```
Dim tmprs As String
```

```
tmprs = ""
```

```
Set conn = CreateObject("adodb.connection")
```

```
Set rs = CreateObject("adodb.recordset")
```

```
strconn = "Driver={SQL
```

```
Server};Description=UserEntre;SERVER=129.137.100.140;UID=jblack;PWD=jb6378;D  
ATABASE=jblacksd"
```

```
conn.Open strconn
```

```
strsql = "SELECT [Company Name] FROM tblCompanyinfo"
```

```
rs.Open strsql, conn, 2, 2
```

```
Do Until rs.EOF
```

```
    tmprs = tmprs & """" & rs("Company Name") & """";"
```

```
rs.MoveNext
```

```
Loop
```

```
selCompany.RowSource = tmprs
```

```
rs.Close
```

```
conn.Close
```

```
txtstudentnum.Value = Forms.frmcoop.txtstudentnum.Value
```

```
Exit_cmdCancel_Click:
```

```
Exit Sub
```

```
Err_cmdCancel_Click:
```

```
MsgBox Err.Description
```

```
Resume Exit_cmdCancel_Click
```

```
End Sub
```

C.1.2 FrmAllCoop

```
Option Compare Database
```

```
Private Sub cmdAddCoop_Click()
```

```
Call OpenCoop(txtstudentnum.Value)
```

```
End Sub
```

```
Function OpenCoop(tmpValue As String)
```


On Error GoTo Err_cmdAddCoop_Click

Dim stDocName As String

Dim stLinkCriteria As String

stDocName = "frmAddCoopQuarter"

DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_cmdAddCoop_Click:

Exit Function

Err_cmdAddCoop_Click:

MsgBox Err.Description

Resume Exit_cmdAddCoop_Click

frmaddcoopquarter.txtTest.Value = tmpValue

End Function

C.1.3 FrmClass

Option Compare Database

Private Sub cmdDeleteClass_Click()

On Error GoTo Err_cmdDeleteClass_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

Exit_cmdDeleteClass_Click:

Exit Sub

Err_cmdDeleteClass_Click:

MsgBox Err.Description

Resume Exit_cmdDeleteClass_Click

End Sub

Private Sub cmdAddClass_Click()

On Error GoTo Err_cmdAddClass_Click

DoCmd.GoToRecord , , acNewRec

Exit_cmdAddClass_Click:

Exit Sub

Err_cmdAddClass_Click:

MsgBox Err.Description

Resume Exit_cmdAddClass_Click

End Sub

Private Sub cmdNextClass_Click()

On Error GoTo Err_cmdNextClass_Click

DoCmd.GoToRecord , , acNext

Exit_cmdNextClass_Click:

Exit Sub

Err_cmdNextClass_Click:

MsgBox Err.Description

Resume Exit_cmdNextClass_Click

```
End Sub
Private Sub cmdPrevClass_Click()
On Error GoTo Err_cmdPrevClass_Click
```

```
DoCmd.GoToRecord , , acPrevious
```

```
Exit_cmdPrevClass_Click:
Exit Sub
```

```
Err_cmdPrevClass_Click:
MsgBox Err.Description
Resume Exit_cmdPrevClass_Click
```

```
End Sub
```

C.1.4 FrmCompany

Option Compare Database

```
Private Sub cmdAddCompany_Click()
On Error GoTo Err_cmdAddCompany_Click
```

```
DoCmd.GoToRecord , , acNewRec
```

```
Exit_cmdAddCompany_Click:
Exit Sub
```

```
Err_cmdAddCompany_Click:
MsgBox Err.Description
Resume Exit_cmdAddCompany_Click
```

End Sub

Private Sub cmdDelCompany_Click()

On Error GoTo Err_cmdDelCompany_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

Exit_cmdDelCompany_Click:

Exit Sub

Err_cmdDelCompany_Click:

MsgBox Err.Description

Resume Exit_cmdDelCompany_Click

End Sub

Private Sub cmdPreCompany_Click()

On Error GoTo Err_cmdPreCompany_Click

DoCmd.GoToRecord , , acPrevious

Exit_cmdPreCompany_Click:

Exit Sub

Err_cmdPreCompany_Click:

MsgBox Err.Description

Resume Exit_cmdPreCompany_Click

End Sub

Private Sub cmdNextCompany_Click()

On Error GoTo Err_cmdNextCompany_Click

```
DoCmd.GoToRecord , , acNext
```

```
Exit_cmdNextCompany_Click:
```

```
Exit Sub
```

```
Err_cmdNextCompany_Click:
```

```
MsgBox Err.Description
```

```
Resume Exit_cmdNextCompany_Click
```

```
End Sub
```

C.1.5 FrmCoop

```
Option Compare Database
```

```
Private Sub cmdAddCoop_Click()
```

```
Call OpenCoop(txtstudentnum.Value)
```

```
End Sub
```

```
Function OpenCoop(tmpValue As String)
```

```
On Error GoTo Err_cmdAddCoop_Click
```

```
Dim stDocName As String
```

```
Dim stLinkCriteria As String
```

```
stDocName = "frmAddCoopQuarter"
```

```
DoCmd.OpenForm stDocName, , , stLinkCriteria
```

Exit_cmdAddCoop_Click:

Exit Function

Err_cmdAddCoop_Click:

MsgBox Err.Description

Resume Exit_cmdAddCoop_Click

frmaddcoopquarter.txtTest.Value = tmpValue

End Function

C.1.6 FrmMain

Option Compare Database

Private Sub cmdOpenStudent_Click()

On Error GoTo Err_cmdOpenStudent_Click

Dim stDocName As String

Dim stLinkCriteria As String

stDocName = "frmStudent"

DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_cmdOpenStudent_Click:

Exit Sub

Err_cmdOpenStudent_Click:

MsgBox Err.Description

Resume Exit_cmdOpenStudent_Click

End Sub

```
Private Sub cmdOpenClass_Click()
On Error GoTo Err_cmdOpenClass_Click

    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "frmClass"
    DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_cmdOpenClass_Click:
    Exit Sub

Err_cmdOpenClass_Click:
    MsgBox Err.Description
    Resume Exit_cmdOpenClass_Click

End Sub

Private Sub cmdOpenAllCoop_Click()
On Error GoTo Err_cmdOpenAllCoop_Click

    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "frmAllCoop"
    DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_cmdOpenAllCoop_Click:
    Exit Sub

Err_cmdOpenAllCoop_Click:
    MsgBox Err.Description
    Resume Exit_cmdOpenAllCoop_Click
```

```

End Sub

Private Sub cmdDegreePlan_Click()
On Error GoTo Err_cmdDegreePlan_Click

    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "frmDegreePlan"
    DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_cmdDegreePlan_Click:
    Exit Sub

Err_cmdDegreePlan_Click:
    MsgBox Err.Description
    Resume Exit_cmdDegreePlan_Click

End Sub

Private Sub cmdCompany_Click()
On Error GoTo Err_cmdCompany_Click

    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "frmCompany"
    DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_cmdCompany_Click:
    Exit Sub

Err_cmdCompany_Click:
    MsgBox Err.Description
    Resume Exit_cmdCompany_Click

```


End Sub

Private Sub Form_Load()

End Sub

Private Sub cmdOpenReport_Click()

On Error GoTo Err_cmdOpenReport_Click

Dim stDocName As String

Select Case selReport.Value

Case "Quarterly Pay Rates by Major"

stDocName = "PayRatesQuarterly"

DoCmd.OpenReport stDocName, acPreview

Case "Overall Pay Rates by Major"

stDocName = "OverallPayRates"

DoCmd.OpenReport stDocName, acPreview

Case Else

MsgBox "Please Choose a Report to View"

selReport.SetFocus

End Select

Exit_cmdOpenReport_Click:

Exit Sub

Err_cmdOpenReport_Click:

MsgBox Err.Description

Resume Exit_cmdOpenReport_Click

End Sub

Private Sub Form_Unload(Cancel As Integer)

Forms.Application.CloseCurrentDatabase

End Sub

C.1.7 FrmStudent

Option Compare Database

Private Sub cmdAddStudent_Click()

On Error GoTo Err_cmdAddStudent_Click

DoCmd.GoToRecord , , acNewRec

Exit_cmdAddStudent_Click:

Exit Sub

Err_cmdAddStudent_Click:

MsgBox Err.Description

Resume Exit_cmdAddStudent_Click

End Sub

Private Sub cmdDeleteStudent_Click()

On Error GoTo Err_cmdDeleteStudent_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

Exit_cmdDeleteStudent_Click:

Exit Sub

Err_cmdDeleteStudent_Click:

MsgBox Err.Description

Resume Exit_cmdDeleteStudent_Click

End Sub

Private Sub Form_Load()

txtStudentnumber.SetFocus

End Sub

Private Sub cmdPre_Click()

On Error GoTo Err_cmdPre_Click

DoCmd.GoToRecord , , acPrevious

Exit_cmdPre_Click:

Exit Sub

Err_cmdPre_Click:

MsgBox Err.Description

Resume Exit_cmdPre_Click

End Sub

```
Private Sub cmdNext_Click()  
On Error GoTo Err_cmdNext_Click
```

```
DoCmd.GoToRecord , , acNext
```

```
Exit_cmdNext_Click:
```

```
Exit Sub
```

```
Err_cmdNext_Click:
```

```
MsgBox Err.Description
```

```
Resume Exit_cmdNext_Click
```

```
End Sub
```

```
Private Sub cmdOpenCoop_Click()
```

```
On Error GoTo Err_cmdOpenCoop_Click
```

```
Dim stDocName As String
```

```
Dim stLinkCriteria As String
```

```
stDocName = "frmCoop"
```

```
stLinkCriteria = "[Student Number]=" & "" & Me![txtStudentnumber] & ""
```

```
DoCmd.OpenForm stDocName, , , stLinkCriteria
```

```
Exit_cmdOpenCoop_Click:
```

```
Exit Sub
```

```
Err_cmdOpenCoop_Click:
```

```
MsgBox Err.Description
```

```
Resume Exit_cmdOpenCoop_Click
```

```
End Sub
```

References

1. Armstrong, Jonathan. "Co-op Degree Plan". http://www.co-op.coe.ttu.edu/coop_plan.htm March 7, 2000.
2. Inbody, Tiffany. "Texas A&M University Research and Graduate Studies". <http://www.tamu.edu/rgs/GraduateStudies/PLANNING/DEGREEPLAN/plan.html> February 23, 2000.
3. "U of Cincinnati-OCAS" <http://www.ocasppcp.uc.edu/> September 17, 1999
4. Williams Baptist College. "Degree Plan". <http://www.wbcoll.edu/degree.htm>.
March 6, 2000