Vanessa Ulloa – CST 338 Final Project

# Simple Student Registration System

## Main Class – FinalProject.java

* Variables/Objects:
  + FPGUI projectMain
    - This object is an instantiation of the FGUI class (see below)
    - Set object to visible (true).
* Methods:
  + N/A

## FPGUI.java

* This class is responsible for the GUI aspects of the program. It will extend JFrame will create the mainPanel which will itself hold the 3 panels that information will be displayed on.
* Variables/Objects:
  + Static final int WIDTH,HEIGHT
  + Static final String FNAME,LNAME
  + JPanel mainPanel,userPanel,schedulePanel,viewPanel
  + JButton registerBtn,dropBtn
  + JLabel userLbl,registerLbl,viewLbl,registerStatusLbl,viewStatusLbl
  + JLabel fnameLbl,lnameLbl,fnameUserLbl,lnameuserLbl
  + JList scheduleList, studentList
  + ScheduleOfClasses theSchedule
    - An object of the ScheduleOfClasses Class
    - This object creates an ArrayList of ClassName objects in String form in order to add to a JList that is viewable on the schedulePanel
  + StudentSchedule newStudentSchedule
    - An object of the StudentSchedule Class
    - This class works with the ScheduleOfClasses objects in the JList and adds them to an ArrayList that is that student’s schedule.
    - Displayed in JList on the viewPanel
* Methods:
  + FPGUI – default constructor
    - Sets basic details of the mainPanel (size, title, background,etc.)
    - Instantiates the JPanels, JButtons and JLabels used as the “titles” of the JPanels.
    - Sets layouts in each panel, using GridLayout on mainPanel, GridBagLayout on userPanel and FlowLayout in the remaining 2 panels.
    - Calls createUserPanel(), createSchedulePanel(), createViewSchedulePanel()
  + createUserPanel()
    - userPanel displays JLabels with the current student information (fnameLbl,lnameLbl,fnameUserLbl,lnameUserLbl)
    - Method adds components to the panel using GridBagConstraints
  + createSchedulePanel()
    - schedulePanel displays JList with master schedule of classes created from the ScheduleOfClasses class.
    - registerBtn uses ActionListener to add class to StudentSchedule object and display in JList in viewPanel
  + createViewSchedulePanel()
    - viewPanel displays JList with “registered” classes from the JList in the schdulePanel
    - Classes can be “dropped” (removed) from the JList int his panel.
    - dropBtn uses ActionListener to “drop” the class and also calls updateViewPanel() to update the display
  + updateViewPanel()
    - resets the Listdata in the JList in the viewPanel
    - calls repaint() to update display

## ClassName.java

* Variables/Objects:
  + Private static string classScheduleName
    - A place where the randomly created class name will be stored and used to populate ArrayLists in ScheduleOfClasses class
  + String[] classID
  + String[] scheduleDays
  + String[] scheduleTimes
    - Above 3 String arrays are the pool that will be used to generate random class names
* Methods:
  + ClassName() – default constructor
    - Populates classScheduleName with results of other class methods
  + getClassScheduleName – accessor for classScheduleName variable
  + generateClassID()
    - uses Random to return random value from the classID array
  + generateScheduleDays()
    - uses Random to return random value from the scheduleDays array
  + generateScheduleItems()
    - uses Random to return random value from the scheduleTimes array

## StudentSchedule.java

* Variables/Objects:
  + ArrayList<string> studentClassSchedule
    - ArrayList that will hold the entire class schedule
    - ArrayList is an easier way to manage multiple entries without having a set array
  + Boolean flag
    - Used to determine if a class has already been registered or not.
* Methods:
  + addClassToSchedule()
    - method that adds a ClassName.toString() value to the ArrayList (that student’s schedule
    - uses method to determine if class already exists within the ArrayList (already registered)
    - if successful, class is added to the ArrayList and status message appears in the schedulePanel. Else, status message also appears and class is not added.
  + dropClassFromSchedule()
    - method to “drop” or remove a class.
    - Uses ArrayList method to remove class from the list
    - Updates status message in the viewPanel
  + isValidRegister(String)
    - method used to check if class already exists in the list (already registered)
    - returns Boolean value of true or false

## ScheduleOfClasses.java

* Variables/Objects:
  + Public static final int MAXSCHEDULECLASSES
  + ArrayList<String> scheduleOfClasses
* Methods:
  + ScheduleOfClasses() – default constructor
    - Calls generateClassList() method
  + generateClassList()
    - uses MAXSCHEDULECLASSES to determine ArrayList size
    - loops through to fill entries with ClassName.toString() values
    - those values are displayed in the schedulePanel

## Testing

* Testing for this program consisted of:
  + Registering for the same class more than once
  + Dropping a class when the student schedule is empty
  + Adding and dropping classes in different order every time.