

Program structure:

When you first log in, there will be a prompt saying “it seems like this is your first time to log in the system.”

```
It seems like this is your first time to log in the system.  
Please enter your USERNAME:
```

When you log in afterwards, it will ask you to choose your identity (left picture).

For admin, after entering your username and password, you can choose among two sections (Course Management and Reports)(right picture)

```
Welcome  
Please tell us your identity:  
(1) Admin  
(2) Student  
(0) Terminate the program  
Please enter the number:
```

```
Please enter the number: 1  
Please enter your USERNAME:Admin  
Please enter your PASSWORD:Admin001  
Following are the options  
  
(1) Course Management  
(2) Reports  
  
(9) Return to user menu  
(0) Terminate program  
Please enter the number:
```

For course management (left picture), and reports (right picture)

```
Following are the options, please enter the number  
  
(1) Create a course  
(2) Delete a course  
(3) Edit a course  
(4) Display a course's information  
(5) Register a student  
  
(0) Terminate program  
(9) Go back to the last menu  
Please enter the number:
```

```
(1) View all courses  
(2) View full courses  
(3) Save a file for full courses  
(4) View the names of the students that are registered in a specific course  
(5) View the list of courses that a given student is registered in  
(6) Sort the courses based on the current number of students registered  
  
(0) Terminate program  
(9) Go back to the last menu  
Please enter the number:
```

For student, after logging in, you will have,

```
Following are the options  
  
(1) View all courses  
(2) View available courses  
(3) Register in a course  
(4) Withdraw from a course  
(5) View enrolled courses  
  
(0) Terminate program  
(9) Return to the previous menu.  
Please enter the number:
```

Method overriding:

- I. Student Class
 - a. Student class **overrides** the method viewCourse() inherited from StudentInterface interface.
 - b. Student class **overrides** the method viewAvailableCourse() inherited from StudentInterface interface.
- II. Admin Class
 - a. Admin class **overrides** the method viewFullCourse() inherited from AdminInterface interface
 - b. Admin class **overrides** the method createStudent() inherited from AdminInterface interface

Abstract class:

- I. User is the abstract class

Inheritance:

- I. Student Class **inherits** from User class, and implements StudentInterface
- II. Admin Class **inherits** from User class, and implements AdminInterface

Encapsulation:

- I. All attributes of Student class are **private**
- II. All attributes of Course class are **private**
- III. All attributes of Admin class are **private**

Abstract data type:

- I. The program uses ArrayList<Student>, ArrayList<Course> to store Students, Courses
- II. Within each Course object, the course uses ArrayList<String>to save the student information.

