Title

Student Management

Background

Write a program to manage information of a student, subject, grading. The program implements the terminology of Object-Oriented Programming (OOP) paradigm. OOP is one of the best choosing ways to design software programs.

In this assignment, we will use an Array List to store a list of students. In fact, an Array List is popularly used to manipulate data.

Program Specifications

Build a Student management program. With the following basic functions

1. Add new student
2. Update Student
    2.1 Update Student
    2.2 Delete Student
3. Add new Subject
4. Update Subject
    4.1 Update Subject
    4.2 Delete Subject
5. Enter Grade
6. Display Student Grade Report
7. Display Subject Grade Report
    Others- Quit Program

Each menu choice should invoke an appropriate function to perform the selected menu item. Your program must display the menu after each task and wait for the user to select another option until the user chooses to quit the program.

Features:

This system contains the following functions:

Display a menu and ask users to select an option.

- **Function 1: Add new student - 50 LOC**
    o   User can require to input a student's information: student id, first name, last name, gender, date of birth, email, phone number.

    o   System can check the valid data with the following conditions:

    – Student id is not allowed to duplicate in the database.
    – Date of birthday must be a real date.
    – first name, last name, gender, date of birth, email, phone number are not allowed null.
    – phone number is number string which has length from 10 to 12.
    – …..

    o   The new student must be appended to the end of the array list.

- o   Application allows adding the student to collection.

- o   Application allows asking to continuous create new student or go back to the main menu.

- ▪ **Function 2: Update student**
  - o   This function requires user enter the student id.

  - o   If student does not exist, the notification "Student does not exist" is shown. Otherwise, user can choose the Update or Delete or back the main menu to choose following functions:

    - ▪ **Function 2.1: Update student information – 50 LOC**
      - –   User can edit student's information. If information is blank, then system does not change old information.

      - –   Application shows the result of the update (success or fail).

      - –   Application asks user to go back to the update menu.

    - ▪ **Function 2.2: Delete student – 50 LOC**
      - –   User can delete the student.

      - –   System must show the confirm message before delete.

      - –   System shows the result of the delete action (success or fail).

      - –   Application asks user to go back to the update menu.

- ▪ **Function 3: Add new subject – 50 LOC**
  - o   This function requires user input new subject information include subject id, subject name, credit.

  - o   Check the valid data with the following conditions:

    - –   Subject id is not allowed to duplicate in the storage.
    - –   Credit must be a positive integer number.
    - –   Subject name is not empty.
    - –   …

  - o   System allows adding subject to collection.

  - o   Application asks user to continuous create new subject or go back to the main menu.

- ▪ **Function 4: Update subject information**
  - o   This function requires user entering the subject id
  - o   If subject does not exist, the notification "Subject does not exist" is shown. Otherwise, user can choose the Update or Delete or Back the main menu function
  - ▪ **Function 4.1: Update subject information – 50 LOC**
    - o   User can edit of the in subject's information. If information is blank, then system does not change old information.

    - o   Application shows the result of the update action (success or fail).

o   Application asks user to go back to the Update menu.

- **Function 4.2: Delete Subject information – 50 LOC**
  o   User can delete the subject.

  o   Application must show the confirm message before delete.

  o   Application shows the result of the delete action (success or fail).

  o   Application asks user to continuous or go back to the Update menu.

- **Function 5: Entering student 's grade – 100 LOC**
  o   This function requires user enter the student id, student id that must be in student list. If student does not exist, the notification "Student does not exist" is shown.

  o   System requires enter the subject id, subject id must be in subject list. If subject does not exist, the notification "Subject does not exist" is shown.

  o   If students have already graded this subject, then system asks user if you want whether overwrite it or not.

  o   Application requires user entering the grade item: Labs, Progress tests, Final exam (Validation is required)

  o   System allows to add the grade item of the student to collection.

  o   System can show the result of the action (success or fail).

  o   Application ask user to continuous or go back to the main menu.

Function 6: Student Grade report – 50 LOC
  o   This function requests user entering the student id, student id must be in student list. If student does not exist, the notification "Student does not exist" is shown. Otherwise, Application will display student's result on the screen. For example:

  -   Student ID: SE150946
  -   Student name: Nguyen Tran Anh Nguyen
      List of subject sort by Subject Name:

++ No ++	+++++++Subject name ++++++++	++ Average mark ++	++ Status ++
1	C#	5.5	Pass
2	Computer Networking	6.5	Pass
3	Introduction Database	4.0	Not Pass

  o   Average mark is calculated following percentage as Labs (30%), Progress tests (30%), Final exam (40%)

  o   Status is pass if average mark more than four. Otherwise, status will be not pass.

  o   Application asks user to go back to the main menu.

Function 7: Subject Grade report – 50 LOC (Ext)

o   This function requests user entering the subject id, subject id must be in subject list. If subject does not exist, the notification "Subject does not exist" is shown. Otherwise, application will display list subject's result on the screen. For example:

- Subject ID: PRJ321
- Student name: Java Web

List of student sort by Student Name:

| ++ Student ID ++ | ++Student name ++ | ++ Average mark ++ | ++ Status ++ |

| SE130580   Le Thanh Dat   5.5   Pass

            SE150865   Nguyen Hoang Lam   6.5   Pass

            SE151316   Tran Vinh An   4.0   Not Pass

o   Application asks user to continuous or go back to the main menu.

- The above specifications are only basic information; you must perform a requirements analysis step and build the application according to real requirements.
- The lecturer will explain the requirement only once on the first slot of the assignment.