

Object Oriented Programming with Java

Lab Practice:4

1. Design a class named 'Student' to represent a Student information. The class contains:

- A String data field StdName that specifies the student name
- A integer data StdAge that specifies student age
- A double data StdGPA that specifies student GPA

Part1:

Write a test class StudentCreate that creates a student class object then set his age. Use this given partial code:

```
Public class Student {  
    // data members  
    // Define instance variables here  
}  
  
public class StudentCreate{  
    public static void main (String[] args){  
        Student one = new Student();  
        .....  
        .....  
    }  
}
```

Part2:

Suppose in the previous part the student has a GPA -2, which is illegible value for a GPA (GPA has to be positive). We have to avoid this by checking the new GPA value to make sure that is an eligible one; i.e., we should use private attributes (information hiding), setter and getter method.

Now make all the variable private instead of public and add the following setter and getters:

- A method name setName (String studentName) to store the student name.
- A method name setAge (int studentAge) to store the student age.
- A method name setGPA (double studentGPA) to store the student GPA.
- A method name getName() that returns the name of student.
- A method name getAge() that returns the age of student.
- A method name getGPA() that returns the GPA of student.

Use the class you in the previous part and the modify the 'StudentCreate' class to use the methods you implemented in part2 to set new values instead of using the variables directly.

Part 3:

As you may notice in the previous part, we were able to force the user of class Student to use

encapsulation

setters to set new values to the variables and we getters in case he needs to read the value of the variable.

In this part, we Student Name: (Any name) Student Age: (Any integer) Student GPA: -2 are going to check the new value inside the setters before we assign the the new value to the instance variable. For example, when the new value of the GPA is -2 we check first if the new value of the GPA is > 0 . If so, we assign it to the instance variable otherwise we print an error message. In this case, -2 is not > 0 so we will not continue with the assignment. Add an if statement inside the setAge() and setGPA() methods to check if the new value is an acceptable value or not. If yes, assign it to the instance variable, if not, print an error message and terminate the program

Part 4. Student Name or Student Age or GPA validation check.
동일한 방식으로
Null 값 check. → positive check

↓
student Age or GPA Null 값 확인?

Why not? → primitive type은 Null 불가.

WOW!!!



It's good idea
Love U Baby!

1. Announcement with Korean & English.
about C.T

2. Lab assignment develop. Lab 4 -
Lab 5 까지.