Wednesday, October 9, 2024 12:18 PM

Assignment on encapsulation

Here's a basic assignment to help you understand encapsulation in Java:

Assignment: Encapsulating a Student Class

Problem:

1. Create a class called Student that encapsulates the following private data members:
- name (String)
- age (int)
- grade (String)
2. Provide public getter and setter methods for each of these data members to access and modify the values.
3. Add validation inside the setter methods:
- The age should be greater than or equal to 5.
- The grade should only accept valid values such as "A", "B", "C", "D", or "F". Any other input should be rejected, and an error message should be displayed.

Task:

1. Define the Student class with private fields and implement the appropriate getters and setters.
2. Create an object of the Student class and use the setter methods to set values.
3. Use the getter methods to print the values.
4. Demonstrate the validation logic in the setter methods by attempting to set invalid data.

Code Template:

```
class Student {
   // Private data members
   private String name;
   private int age;
   private String grade;

   // Getter for name
   public String getName() {
      return name;
   }

   // Setter for name
   public void setName(String name) {
      this.name = name;
   }

   // Getter for age
   public int getAge() {
      return age;
   }

   // Setter for age with validation
   public void setAge(int age) {
      if (age >= 5) {
         this.age = age;
      } else {
         System.out.println("Age must be at least 5.");
      }
   }
```

```
    // Getter for grade
    public String getGrade() {
        return grade;
    }

    // Setter for grade with validation
    public void setGrade(String grade) {
        if (grade.equals("A") || grade.equals("B") || grade.equals("C") ||
            grade.equals("D") || grade.equals("F")) {
            this.grade = grade;
        } else {
            System.out.println("Invalid grade. Please enter A, B, C, D, or F.");
        }
    }
}

public class Main {
    public static void main(String[] args) {
        // Create an instance of Student
        Student student = new Student();

        // Setting values using setter methods
        student.setName("John Doe");
        student.setAge(10);  // Valid age
        student.setGrade("A");  // Valid grade

        // Accessing values using getter methods
        System.out.println("Student Name: " + student.getName());
        System.out.println("Student Age: " + student.getAge());
        System.out.println("Student Grade: " + student.getGrade());

        // Demonstrating validation
        student.setAge(3);  // Invalid age
        student.setGrade("E");  // Invalid grade
    }
}
```

Expected Output:

```
Student Name: John Doe
Student Age: 10
Student Grade: A
Age must be at least 5.
Invalid grade. Please enter A, B, C, D, or F.
```

Key Concepts:

- Encapsulation: The Student class has private data members and only allows access to these members through public getter and setter methods.
- Data Validation: The setter methods contain logic to ensure the age and grade values are valid, demonstrating how encapsulation controls access to an object's internal state.