FPT SOFTWARE

FRESHER ACADEMY

BASIC JAVA

Ngattt (FHO.FA)

04/20/2022

[Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.]

Contents

Objective 2

Business needs 2

Working requirements 2

Technologies..2

Project Descriptions 2

Objective

- Fresher can apply knowledge about Abstract class, Subclasses to code a simple application.
- Create a Student class with instance variables for name, roll number, and grade.
    - o Implement appropriate constructors, getters, and setters for the Student class.
- Create a StudentManagementSystem class that allows users to add new students, remove students, and display the list of all students.
    - o Implement a method to calculate and display the average grade of all students.
- Fresher can create a class, and write some methods
- Fresher can create some instances of class and test all the methods of the class.
- Fresher can using ArrayList: Click here to read about ArrayList

Business needs

- Allow users to add as many students as they want to the system.
- Allow users to remove students from the system based on their roll numbers.
- Display the list of all students with their details (name, roll number, and grade).
- Calculate and display the average grade of all students.
- ***Do not verify information user input***

Working requirements

- Working environment:Eclipse/IntelliJ.

Technologies

- The product implements Java program language with: Abstract class, Sub-classes

Project Descriptions

1. **Step 1: Create a new Java project** (skip this step if you already have YourFullName_JavaSE project)

- Open your preferred Java IDE (such as Eclipse, IntelliJ, or NetBeans).
- Create a new Java project by selecting File > New > Java Project.
- Name the project YourFullName_JavaSE, **for example** "NguyenVanA_JavaSE" and click Finish.

2. **Step 2: Create a package** (skip this step if you already have **lab4** package)
    - In the project explorer, right-click on the src folder and select New > Package.
    - Name the package "**lab4**" and click Finish.
3. **Step 3: Create a Student class**
    - In the Package Explorer panel on the left side of the screen, right-click on the '**lab4**' package to create the class.
    - Select "New" from the context menu, then "Class" from the submenu.
    - In the "New Java Class" dialog box, enter "**Student**" for the class in the "Name" field.
    - Click "Finish" to create the class.
4. **Step 4: Create a StudentManagementSystem class**
    - In the Package Explorer panel on the left side of the screen, right-click on the '**lab4**' package to create the class.
    - Select "New" from the context menu, then "Class" from the submenu.
    - In the "New Java Class" dialog box, enter "**StudentManagementSystem**" for the class in the "Name" field.
    - Click "Finish" to create the class.
5. **Step 5: Create a StudentManagementSystemTest class**
    - In the Package Explorer panel on the left side of the screen, right-click on the '**lab4**' package to create the class.
    - Select "New" from the context menu, then "Class" from the submenu.
    - In the "New Java Class" dialog box, enter "**StudentManagementSystemTest**" for the class in the "Name" field.
    - Click "Finish" to create the class.
6. **Step 6**: Coding for **Student class:**
    - The Student class will have instance variables for name, roll number, and grade.
    - Implement constructors, getters, and setters for the Student class to set and retrieve these attributes.
    - Implement **toString()** method to show information of Student
7. **Step 7**: Coding for **StudentManagementSystem class:**
    - **Step 7.1**: Add students:
        - Users can provide the necessary details for the new student (name, roll number, and grade).
        - The new student object will be created and added to the list of students.
    - **Step 7.2:** Remove students:
        - Provide the roll number of the student to be removed.

  - The student with the matching roll number will be removed from the list of students.
- **Step 7.3:** Display the list of all students:
  - The method will iterate through the list of students and print the details of each student (name, roll number, and grade).
- **Step 7.4:** Calculate and display the average grade of all students:
  - The method will iterate through the list of students, sum up their grades, and then calculate the average grade by dividing the sum by the total number of students.

8. **Step 8**: Coding for **StudentManagementSystemTest** class:
   - In the main() method:
     - Create at least 3 instances for Student class:
     - Test all the methods of **Student** and **StudentManagementSystem** classes.

The End!