Course: JAVA (2) **Semester**: 1st term 2025/2026.	**Lecturer**: Dr. Mohammed AbdelFattah **Assistants**: Engs. / Mostafa El-Mansi, Osama El-Samdouni, Rahgab Hassan, Rana Zakaria, Heba Allah, Maryam Fathy

Assignment 2

Student Information

Name: Abdulrahman Khamis Abdo

Faculty ID: 2323326 Mark: ..... / 10

Write T (True) or F (False):

Q1. A class may have multiple constructors. (T)

Q2. Class does not allow constructors overloading. (F)

Q3. Class does not have any constructor by default. (F)

Q4. The setter method(s) is used to modify the instance variable(s). (T)

Q5. The static method can be accessed by creating an object of the class. (F)

Q6. A constructor in Java is a special method that is used to initialize objects. (T)

Q7. The instance method can be accessed without creating an object of the class. (F)

Q8. The constructor name may not match the class name, and it can have a return type. (F)

Q9. Java enables you to pass a variable number of arguments of the same type to a method. (T)

Q10. The getter method is used to read the instance variable and is known as the accessor method. (T)

Q11. The static keyword refers that the attribute/method belongs to the class, rather than an object. (T)

Write the following program:

Q13. Develop a student class that contains the following data members (id, name, GPA, universityName, and universityAddress) and has the following:

- Make these fields (universityName and universityAddress) as a static attribute.
- Create a “No-Args Constructor” that initializes the student information with default values.
- Create a “Parameterized Constructor” that passes initial values for student objects.
- Create getter and setter methods for these data members (id, name, and GPA).
- Create a static method that prints the university information.
- Create an instance method that prints student information.

Coding:

```
public class Student {


    private int id;
    private String name;
    private double GPA;

    // Static attributes
    static String universityName;
    static String universityAddress;

    // No-Args Constructor
    public Student() {
        this.id = 0;
        this.name = "No Name";
        this.GPA = 0.0;
```

```
        // Default static information
        universityName = "Default University";
        universityAddress = "Default Address";
    }

    // Parameterized Constructor
    public Student(int id, String name, double GPA) {
        this.id = id;
        this.name = name;
        this.GPA = GPA;
    }

    // Getter and Setter for id
    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

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

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

    // Getter and Setter for GPA
    public double getGPA() {
        return GPA;
    }

    public void setGPA(double GPA) {
        this.GPA = GPA;
    }

    // Static method to print university info
    public static void printUniversityInfo() {
        System.out.println("University Name: " + universityName);
        System.out.println("University Address: " + universityAddress);
    }

    // Instance method to print student info
    public void printStudentInfo() {
        System.out.println("ID: " + id);
        System.out.println("Name: " + name);
        System.out.println("GPA: " + GPA);
    }
}
```