JAVA ASSIGNMENT -2

1.Create a class called "Car" that has the following properties: make, model, year, color, and price. Include a constructor and getter and setter methods for each property.

```
public class Car {

    private String make;

    private String model;

    private int year;

    private String color;

    private double price;


    // Constructor

    public Car(String make, String model, int year, String color, double price) {

        this.make = make;

        this.model = model;

        this.year = year;

        this.color = color;

        this.price = price;

    }


    // Getter and Setter methods for make

    public String getMake() {

        return make;

    }
```

```
public void setMake(String make) {
    this.make = make;
}

// Getter and Setter methods for model
public String getModel() {
    return model;
}

public void setModel(String model) {
    this.model = model;
}

// Getter and Setter methods for year
public int getYear() {
    return year;
}

public void setYear(int year) {
    this.year = year;
}

// Getter and Setter methods for color
public String getColor() {
    return color;
```

```
}

public void setColor(String color) {
    this.color = color;
}

// Getter and Setter methods for price
public double getPrice() {
    return price;
}

public void setPrice(double price) {
    this.price = price;
}

public static void main(String[] args) {
    // Creating an instance of the Car class
    Car myCar = new Car("Toyota", "Camry", 2022, "Blue", 25000.0);

    // Accessing and printing the properties of the car
    System.out.println("Make: " + myCar.getMake());
    System.out.println("Model: " + myCar.getModel());
    System.out.println("Year: " + myCar.getYear());
    System.out.println("Color: " + myCar.getColor());
    System.out.println("Price: $" + myCar.getPrice());
```

```
    }
}
```

OUTPUT;

Make: Toyota

Model: Camry

Year: 2022

Color: Blue

Price: $25000.0

2.Create a class called "Student" that has the following properties: name, age, gender, grade, and GPA. Include a constructor and getter and setter methods for each property

```
public class Student {
    private String name;
    private int age;
    private String gender;
    private int grade;
    private double GPA;

    // Constructor
    public Student(String name, int age, String gender, int grade, double GPA) {
        this.name = name;
        this.age = age;
        this.gender = gender;
        this.grade = grade;
        this.GPA = GPA;
    }
```

```
// Getter and Setter methods for each property
public String getName() {
    return name;
}

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

public int getAge() {
    return age;
}

public void setAge(int age) {
    this.age = age;
}

public String getGender() {
    return gender;
}

public void setGender(String gender) {
    this.gender = gender;
}
```

```
    public int getGrade() {
        return grade;
    }

    public void setGrade(int grade) {
        this.grade = grade;
    }

    public double getGPA() {
        return GPA;
    }

    public void setGPA(double GPA) {
        this.GPA = GPA;
    }
}public class Main {
    public static void main(String[] args) {
        // Create a new student instance
        Student myStudent = new Student("Alice", 18, "Female", 12, 3.8);

        // Access properties using getter methods
        System.out.println("Student Information:");
        System.out.println("Name: " + myStudent.getName());
        System.out.println("Age: " + myStudent.getAge());
```

```
        System.out.println("Gender: " + myStudent.getGender());
        System.out.println("Grade: " + myStudent.getGrade());
        System.out.println("GPA: " + myStudent.getGPA());

        // Modify properties using setter methods
        myStudent.setGPA(4.0);
        System.out.println("\nUpdated GPA: " + myStudent.getGPA());
    }
}
```

OUTPUT;

Student Information:

Name: Alice

Age: 18

Gender: Female

Grade: 12

GPA: 3.8

Updated GPA: 4.0

3.. Create a class called "Circle" that has the following properties: radius, diameter, and area. Include a constructor and methods to calculate the diameter and area of the circle.

```
public class Circle {
    private double radius;
    private double diameter;
    private double area;
```

```
// Constructor
public Circle(double radius) {
    this.radius = radius;
    this.diameter = calculateDiameter();
    this.area = calculateArea();
}

// Method to calculate diameter
private double calculateDiameter() {
    return 2 * radius;
}

// Method to calculate area
private double calculateArea() {
    return Math.PI * Math.pow(radius, 2);
}

// Getter methods
public double getRadius() {
    return radius;
}

public double getDiameter() {
    return diameter;
}
```

```
    public double getArea() {
        return area;
    }
}public class CircleExample {
    public static void main(String[] args) {
        // Create a circle instance with radius 5.0
        Circle myCircle = new Circle(5.0);


        // Access properties using getter methods
        System.out.println("Radius: " + myCircle.getRadius());
        System.out.println("Diameter: " + myCircle.getDiameter());
        System.out.println("Area: " + myCircle.getArea());
    }
}
```

OUTPUT;

Radius: 5.0

Diameter: 10.0

Area: 78.53981633974483

4.Create a class called "Rectangle" that has the following properties: length, width, and area. Include a constructor and a method to calculate the area of the rectangle.

```
public class Rectangle {
    private double length;
    private double width;
    private double area;
```

```
// Constructor
public Rectangle(double length, double width) {
    this.length = length;
    this.width = width;
    calculateArea();
}

// Method to calculate area
private void calculateArea() {
    this.area = length * width;
}

// Getter methods
public double getLength() {
    return length;
}

public double getWidth() {
    return width;
}

public double getArea() {
    return area;
}
```

```
}public class RectangleExample {
    public static void main(String[] args) {
        // Create a new rectangle instance with length 4 and width 6
        Rectangle myRectangle = new Rectangle(4.0, 6.0);

        // Access properties using getter methods
        System.out.println("Length: " + myRectangle.getLength());
        System.out.println("Width: " + myRectangle.getWidth());
        System.out.println("Area: " + myRectangle.getArea());
    }
}
```

OUTPUT;

Length: 4.0

Width: 6.0

Area: 24.0

5.Create a class called "BankAccount" that has the following properties: account number, account balance, account holder name, and account type. Include a constructor and methods to deposit and withdraw money from the account.

```
public class BankAccount {
    private String accountNumber;
    private double accountBalance;
    private String accountHolderName;
    private String accountType;

    // Constructor
```

```
    public BankAccount(String accountNumber, double accountBalance, String accountHolderName, String accountType) {
        this.accountNumber = accountNumber;
        this.accountBalance = accountBalance;
        this.accountHolderName = accountHolderName;
        this.accountType = accountType;
    }

    // Method to deposit money
    public void deposit(double amount) {
        accountBalance += amount;
        System.out.println("Deposit of $" + amount + " successful. New balance: $" + accountBalance);
    }

    // Method to withdraw money
    public void withdraw(double amount) {
        if (amount <= accountBalance) {
            accountBalance -= amount;
            System.out.println("Withdrawal of $" + amount + " successful. New balance: $" + accountBalance);
        } else {
            System.out.println("Insufficient funds. Withdrawal unsuccessful.");
        }
    }
```

```
    // Getter methods
    public String getAccountNumber() {
        return accountNumber;
    }

    public double getAccountBalance() {
        return accountBalance;
    }

    public String getAccountHolderName() {
        return accountHolderName;
    }

    public String getAccountType() {
        return accountType;
    }
}public class Main {
    public static void main(String[] args) {
        // BankAccount example
        BankAccount myAccount = new BankAccount("123456789", 1000.0, "John Doe", "Savings");
        System.out.println("Account Holder: " + myAccount.getAccountHolderName());
        System.out.println("Initial Balance: $" + myAccount.getAccountBalance());
        myAccount.deposit(500.0);
        myAccount.withdraw(200.0);
```

```
 System.out.println();
   }}
```

OUTPUT;

Account Holder: John Doe

Initial Balance: $1000.0

Deposit of $500.0 successful. New balance: $1500.0

Withdrawal of $200.0 successful. New balance: $1300**.0**

6.Create a class called "Person" that has the following properties: name, age, address, phone number, and email address. Include a constructor and getter and setter methods for each property.

```
public class Person {
   private String name;
   private int age;
   private String address;
   private String phoneNumber;
   private String emailAddress;

   // Constructor
   public Person(String name, int age, String address, String phoneNumber, String emailAddress) {
      this.name = name;
      this.age = age;
      this.address = address;
      this.phoneNumber = phoneNumber;
      this.emailAddress = emailAddress;
```

```
}

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

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

public int getAge() {
    return age;
}

public void setAge(int age) {
    this.age = age;
}

public String getAddress() {
    return address;
}

public void setAddress(String address) {
    this.address = address;
```

```
    }

    public String getPhoneNumber() {
        return phoneNumber;
    }

    public void setPhoneNumber(String phoneNumber) {
        this.phoneNumber = phoneNumber;
    }

    public String getEmailAddress() {
        return emailAddress;
    }

    public void setEmailAddress(String emailAddress) {
        this.emailAddress = emailAddress;
    }}public class Main {
    public static void main(String[] args) {
  // Person example
        Person person = new Person("Alice", 25, "123 Main St", "555-1234",
"alice@example.com");
        System.out.println("Person Information:");
        System.out.println("Name: " + person.getName());
        System.out.println("Age: " + person.getAge());
        System.out.println("Address: " + person.getAddress());
```

```
        System.out.println("Phone Number: " + person.getPhoneNumber());
        System.out.println("Email Address: " + person.getEmailAddress());

        System.out.println();
```

OUTPUT:

Person Information:

Name: Alice

Age: 25

Address: 123 Main St

Phone Number: 555-1234

Email Address: alice@example.com

7.Create a class called "Animal" that has the following properties: name, species, age, and weight. Include a constructor and getter and setter methods for each property.

```
public class Animal {
    private String name;
    private String species;
    private int age;
    private double weight;

    // Constructor
    public Animal(String name, String species, int age, double weight) {
        this.name = name;
        this.species = species;
```

```
    this.age = age;
    this.weight = weight;
}

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

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

public String getSpecies() {
    return species;
}

public void setSpecies(String species) {
    this.species = species;
}

public int getAge() {
    return age;
}
```

```
    public void setAge(int age) {
        this.age = age;
    }

    public double getWeight() {
        return weight;
    }

    public void setWeight(double weight) {
        this.weight = weight;
    }}public class Main {
    public static void main(String[] args) {
        Animal lion = new Animal("Leo", "Lion", 5, 150.0);
        System.out.println("Animal Information:");
        System.out.println("Name: " + lion.getName());
        System.out.println("Species: " + lion.getSpecies());
        System.out.println("Age: " + lion.getAge());
        System.out.println("Weight: " + lion.getWeight() + " kg");
    }
}
```

OUTPUT:

Animal Information:

Name: Leo

Species: Lion

Age: 5

Weight: 150.0 kg

8.Create a class called "Triangle" that has the following properties: base, height, and area. Include a constructor and a method to calculate the area of the triangle.

```
public class Triangle {
    private double base;
    private double height;
    private double area;

    // Constructor
    public Triangle(double base, double height) {
        this.base = base;
        this.height = height;
        calculateArea();
    }

    // Method to calculate area
    private void calculateArea() {
        this.area = 0.5 * base * height;
    }

    // Getter methods
    public double getBase() {
        return base;
    }
```

```
    public double getHeight() {
        return height;
    }

    public double getArea() {
        return area;
    }
}public class TriangleExample {
    public static void main(String[] args) {
        // Create a new triangle instance with base 5 and height 8
        Triangle myTriangle = new Triangle(5.0, 8.0);

        // Access properties using getter methods
        System.out.println("Base: " + myTriangle.getBase());
        System.out.println("Height: " + myTriangle.getHeight());
        System.out.println("Area: " + myTriangle.getArea());
    }
}
```

OUTPUT;

Base: 5.0

Height: 8.0

Area: 20.0

9.. Create a class called "Employee" that has the following properties: name, employee ID, department, job title, and salary. Include a constructor and getter and setter methods for each property

```
public class Employee {
```

```
    private String name;
    private int employeeID;
    private String department;
    private String jobTitle;
    private double salary;

    // Constructor
    public Employee(String name, int employeeID, String department, String
jobTitle, double salary) {
        this.name = name;
        this.employeeID = employeeID;
        this.department = department;
        this.jobTitle = jobTitle;
        this.salary = salary;
    }

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

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

```
public int getEmployeeID() {
    return employeeID;
}

public void setEmployeeID(int employeeID) {
    this.employeeID = employeeID;
}

public String getDepartment() {
    return department;
}

public void setDepartment(String department) {
    this.department = department;
}

public String getJobTitle() {
    return jobTitle;
}

public void setJobTitle(String jobTitle) {
    this.jobTitle = jobTitle;
}

public double getSalary() {
```

```
    return salary;
  }

  public void setSalary(double salary) {
    this.salary = salary;
  }
}public class EmployeeExample {
  public static void main(String[] args) {
    // Create a new employee instance
    Employee employee = new Employee("John Doe", 12345, "IT", "Software
Engineer", 75000.0);

    // Access properties using getter methods
    System.out.println("Name: " + employee.getName());
    System.out.println("Employee ID: " + employee.getEmployeeID());
    System.out.println("Department: " + employee.getDepartment());
    System.out.println("Job Title: " + employee.getJobTitle());
    System.out.println("Salary: $" + employee.getSalary());
  }
```

OUTPUT;

Name: John Doe

Employee ID: 12345

Department: IT

Job Title: Software Engineer

Salary: $75000.0

10.Create a class called "Address" that has the following properties: street, city, state, zip code, and country. Include a constructor and getter and setter methods for each property.

```
public class Address {
    private String street;
    private String city;
    private String state;
    private String zipCode;
    private String country;

    // Constructor
    public Address(String street, String city, String state, String zipCode, String country) {
        this.street = street;
        this.city = city;
        this.state = state;
        this.zipCode = zipCode;
        this.country = country;
    }

    // Getter and Setter methods for each property
    public String getStreet() {
        return street;
    }

    public void setStreet(String street) {
```

```
    this.street = street;
}

public String getCity() {
    return city;
}

public void setCity(String city) {
    this.city = city;
}

public String getState() {
    return state;
}

public void setState(String state) {
    this.state = state;
}

public String getZipCode() {
    return zipCode;
}

public void setZipCode(String zipCode) {
    this.zipCode = zipCode;
```

```
    }

    public String getCountry() {
        return country;
    }

    public void setCountry(String country) {
        this.country = country;
    }
}
public class AddressExample {
    public static void main(String[] args) {
        // Create a new address instance
        Address myAddress = new Address("123 Main St", "Cityville", "Stateville",
"12345", "Countryland");

        // Access properties using getter methods
        System.out.println("Street: " + myAddress.getStreet());
        System.out.println("City: " + myAddress.getCity());
        System.out.println("State: " + myAddress.getState());
        System.out.println("Zip Code: " + myAddress.getZipCode());
        System.out.println("Country: " + myAddress.getCountry());
    }
}
```

OUTPUT;

Street: 123 Main St

City: Cityville

State: Stateville

Zip Code: 12345

Country: Countryland