ASSIGNMENT - 2

Gollamandala Harika

1.Create a class Person with attributes like name, age, and methods to display these attributes.

Program:

```
public class Person
        {
   private String name;
   private int age;
   public Person(String name, int age)
                {
      this.name = name;
      this.age = age;
      }
                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 void displayPerson()
                {
      System.out.println("Name: " + name);
      System.out.println("Age: " + age);
      }
      public static void main(String[] args)
                {
      Person person = new Person("Harika", 20);
      person.displayPerson();
      person.setName("Stency");
      person.setAge(19);
```

```
        person.displayPerson();
    }
}
```

2. Implement a BankAccount class with methods for deposit, withdrawal, and displaying the account balance.

Program:

```
public class BankAccount {
    private double balance;
    public BankAccount(double initialBalance) {
        if (initialBalance < 0) {
            throw new IllegalArgumentException("Initial balance cannot be negative.");
        }
        this.balance = initialBalance;
    }
    public void deposit(double amount)
                    {
        if (amount <= 0) {
            throw new IllegalArgumentException("Deposit amount must be positive.");
        }
        balance += amount;
    }
    public void withdraw(double amount)
                    {
        if (amount <= 0)
                            {
            throw new IllegalArgumentException("Withdrawal amount must be positive.");
        }
        if (amount > balance)
                            {
            throw new IllegalArgumentException("Insufficient funds.");
        }
        balance -= amount;
    }
    public void displayBalance()
            {
        System.out.printf("Current balance: $%.2f%n", balance);
    }
    public static void main(String[] args)
                    {
        BankAccount account = new BankAccount(1000.00);
        account.displayBalance();
        account.deposit(800.00);
        account.displayBalance();
```

```
        account.withdraw(400.00);
        account.displayBalance();
        try
                {
            account.withdraw(1500.00);
        }
                catch (IllegalArgumentException e)
                        {
            System.out.println(e.getMessage());
        }
        account.displayBalance();
    }
}
```

3.Write a program that uses constructors to initialize objects and demonstrates method overloading.

Program:

```
public class Rectangle {
    private double width;
    private double height;

    public Rectangle(double width, double height)
                {
        this.width = width;
        this.height = height;
    }

    public Rectangle(double side) {
        this(side, side);
    }
    public Rectangle() {
        this.width = 3;
        this.height = 3;
    }
    public double calculateArea()
                {
        return width * height;
    }
    public double calculateArea(double side)
                {
        return side * side;
    }
    public double calculatePerimeter()
                {
```

```
        return 2 * (width + height);
    }
    public double calculatePerimeter(double side)
   {
        return 4 * side;
    }
    public void display() {
        System.out.printf("Rectangle width: %.2f, height: %.2f%n", width, height);
        System.out.printf("Area: %.2f%n", calculateArea());
        System.out.printf("Perimeter: %.2f%n", calculatePerimeter());
    }

    public static void main(String[] args) {

        Rectangle rect1 = new Rectangle(7, 5);
        rect1.display();


        Rectangle square = new Rectangle(8);
        square.display();


        Rectangle defaultRect = new Rectangle();
        defaultRect.display();

        System.out.printf("Area of square with side 6.0: %.2f%n", square.calculateArea(6));
        System.out.printf("Perimeter of square with side 6.0: %.2f%n",
square.calculatePerimeter(6));
    }
}
```