

DEEBAK KUMAR K
192324064
CLASS PROBLEMS

Question 1:

write java program number less than 10 and greater than 50 exception throws out of range and it should square number use scanner class (valid range 10to50)

Code:

```
import java.util.Scanner;

public class NumberRangeValidator {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        try {
            // Prompt user for a number
            System.out.print("Enter a number between 10 and 50: ");
            int number = scanner.nextInt();

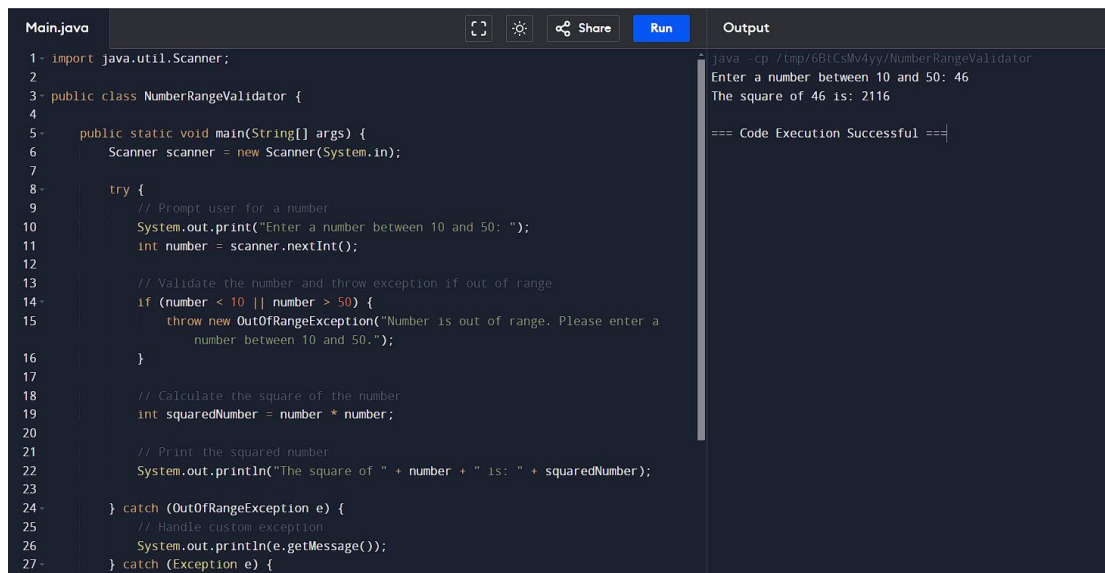
            // Validate the number and throw exception if out of range
            if (number < 10 || number > 50) {
                throw new OutOfRangeException("Number is out of range. Please
enter a number between 10 and 50.");
            }

            // Calculate the square of the number
            int squaredNumber = number * number;

            // Print the squared number
            System.out.println("The square of " + number + " is: " +
squaredNumber);

        } catch (OutOfRangeException e) {
            // Handle custom exception
            System.out.println(e.getMessage());
        } catch (Exception e) {
            // Handle any other exceptions
            System.out.println("Invalid input. Please enter an integer.");
        } finally {
            // Close the scanner
            scanner.close();
        }
    }
}
```

```
// Custom exception class for out of range errors
class OutOfRangeException extends Exception {
    public OutOfRangeException(String message) {
        super(message);
    }
}
```



```
Main.java
1- import java.util.Scanner;
2
3- public class NumberRangeValidator {
4
5-     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7
8         try {
9             // Prompt user for a number
10            System.out.print("Enter a number between 10 and 50: ");
11            int number = scanner.nextInt();
12
13            // Validate the number and throw exception if out of range
14            if (number < 10 || number > 50) {
15                throw new OutOfRangeException("Number is out of range. Please enter a
                    number between 10 and 50.");
16            }
17
18            // Calculate the square of the number
19            int squaredNumber = number * number;
20
21            // Print the squared number
22            System.out.println("The square of " + number + " is: " + squaredNumber);
23
24        } catch (OutOfRangeException e) {
25            // Handle custom exception
26            System.out.println(e.getMessage());
27        } catch (Exception e) {
28
29        }
30    }
31}
```

```
Output
java -cp /tmp/6BtCsMv4yy/NumberRangeValidator
Enter a number between 10 and 50: 46
The square of 46 is: 2116

=== Code Execution Successful ===
```

Question 2:

create box class with data members width,height,depth.create two subclasses boxweight with weight as data member and ship man with cost data members

Code:

```
// Base class
class Box {
    // Data members for Box class
    double width;
    double height;
    double depth;

    // Constructor for Box class
    public Box(double width, double height, double depth) {
        this.width = width;
        this.height = height;
        this.depth = depth;
    }

    // Method to compute volume of the box
    public double volume() {
        return width * height * depth;
    }
}
```

```

    }

    // Method to display box dimensions
    public void displayDimensions() {
        System.out.println("Width: " + width);
        System.out.println("Height: " + height);
        System.out.println("Depth: " + depth);
    }
}

// Subclass of Box that includes weight
class BoxWeight extends Box {
    // Data member for weight
    double weight;

    // Constructor for BoxWeight class
    public BoxWeight(double width, double height, double depth, double
weight) {
        super(width, height, depth); // Call the constructor of the base class
        this.weight = weight;
    }

    // Method to display weight
    public void displayWeight() {
        System.out.println("Weight: " + weight);
    }
}

// Subclass of Box that includes shipping cost
class ShipMan extends Box {
    // Data member for cost
    double cost;

    // Constructor for ShipMan class
    public ShipMan(double width, double height, double depth, double cost) {
        super(width, height, depth); // Call the constructor of the base class
        this.cost = cost;
    }

    // Method to display shipping cost
    public void displayCost() {
        System.out.println("Shipping Cost: " + cost);
    }
}

// Main class to test the functionality
public class BoxTest {
    public static void main(String[] args) {
        // Create an instance of BoxWeight
        BoxWeight boxWeight = new BoxWeight(10, 20, 15, 25);
        System.out.println("BoxWeight Details:");
    }
}

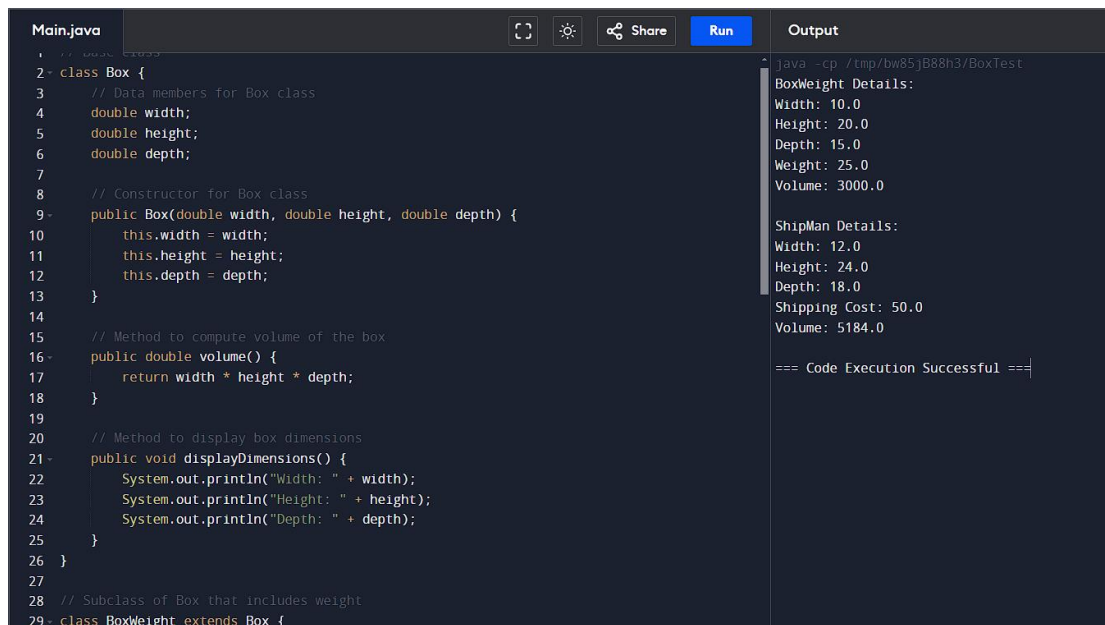
```

```

        boxWeight.displayDimensions();
        boxWeight.displayWeight();
        System.out.println("Volume: " + boxWeight.volume());

        // Create an instance of ShipMan
        ShipMan shipMan = new ShipMan(12, 24, 18, 50);
        System.out.println("\nShipMan Details:");
        shipMan.displayDimensions();
        shipMan.displayCost();
        System.out.println("Volume: " + shipMan.volume());
    }
}

```



The screenshot shows a Java IDE with a file named `Main.java`. The code defines a `Box` class with attributes `width`, `height`, and `depth`, and methods `volume()` and `displayDimensions()`. A subclass `BoxWeight` extends `Box` and adds a `weight` attribute and a `displayCost()` method. The program creates a `BoxWeight` object and a `ShipMan` object, and prints their details.

```

Main.java
1 // Main class
2 class Box {
3     // Data members for Box class
4     double width;
5     double height;
6     double depth;
7
8     // Constructor for Box class
9     public Box(double width, double height, double depth) {
10         this.width = width;
11         this.height = height;
12         this.depth = depth;
13     }
14
15     // Method to compute volume of the box
16     public double volume() {
17         return width * height * depth;
18     }
19
20     // Method to display box dimensions
21     public void displayDimensions() {
22         System.out.println("Width: " + width);
23         System.out.println("Height: " + height);
24         System.out.println("Depth: " + depth);
25     }
26 }
27
28 // Subclass of Box that includes weight
29 class BoxWeight extends Box {

```

Output

```

java -cp ./tmp/bw85jB88h3/BoxTest
BoxWeight Details:
Width: 10.0
Height: 20.0
Depth: 15.0
Weight: 25.0
Volume: 3000.0

ShipMan Details:
Width: 12.0
Height: 24.0
Depth: 18.0
Shipping Cost: 50.0
Volume: 5184.0

=== Code Execution Successful ===

```