**Abstraction**

1. Create an abstract class Shape
2. The Shape class has two abstract methods calculateArea() and calculatePerimeter(). Both the methods have a return type of void.
3. Create a class Quadrilateral which extends the abstract class Shape.
4. Implement all the abstract method of the parent class
5. Create an abstract class named Vehicle which consist of two methods: wheel and door. Both the methods have void return type and no parameters. The method wheel has no implementation.
6. Create a class name Bus and extend the Vehicle class.

A screenshot of a computer program

Description automatically generatedA computer screen shot of text

Description automatically generatedA screenshot of a computer program

Description automatically generated

**Interface**

1. Create an interface Animal. The Animal interface has two methods eat() and walk()
2. Create another interface Printable. The Printable interface has a method called display();
3. Create a class Cow that implements the Animal and Printable interfaces
4. Create an interface LivingBeing
5. Create an method void specialFeature()

A black screen with text on it

Description automatically generated

A screen shot of a computer

Description automatically generated

**Classes**

1. Create 2 classes Fish and Bird that implements LivingBeing
2. The specialFeature should display special feature of the respective class animal.

A black rectangle with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

**Exception**

1. In the following program, which exception will be generated

public class Demo{

public static void main(String[] args) {

System.out.println(10/0);

}

}

Handle the exception above by using try-catch.

**A screen shot of a computer program

Description automatically generated**

1. In the following program, which exception will be generated

public class Demo{

public static void main(String[] args) {

int[] age = {10,20,25,24,28,27,30,31,32};

System.out.println(age[9]);

}

}

Handle the exception by using throws keyword.

A screenshot of a computer program

Description automatically generated

**Regular Expressions**

1. Write a Java program to check whether a string contains only a certain set of characters (in this case a-z, A-Z and 0-9).

A computer screen with text on it

Description automatically generated

1. Write a Java program to find the sequence of one upper case letter followed by lower case letters. Z

A screen shot of a computer program

Description automatically generated

1. Develop a Java program to check if a given string represents a file with a ".txt" extension.

A screenshot of a computer program

Description automatically generated

1. Write a Java program that validates usernames based on the following criteria**:**

* Should start with a letter.
* Can include letters, numbers, and underscores.
* Should be between 3 and 16 characters in length.

**A screenshot of a computer program

Description automatically generated**