**Assignment – 8**

1. What is the difference between interface **and** *abstract* class?(10 points)

2. Write a Java program to create an interface **Shape** with the getArea() method. Create three classes Rectangle, Circle, and Triangle that implement the Shape interface. Implement the getArea() method for each of the three classes.(10 points)

3. Write a Java program to create a Animal interface **with** a method called bark() that takes no arguments and returns void. Create a Dog class **that** *implements* *Animal* and overrides speak() to print "*Dog* is barking".

4.write a program to reverse a *String* using recursion?(10 points)

5.*Write* a program using cloneable interface?

**Answers:**

**1. Ans:**

**2. Ans:**

**2.** interface **Shape** {

*double* getArea()**;**

}

class **Rectangle** *implements* *Shape* {

*private* *double* length**;**

*private* *double* width**;**

*public* Rectangle(*double* **length,** *double* **width**) {

*this***.***length* **=** length**;**

*this***.***width* **=** width**;**

    }

    @**Override**

*public* *double* getArea() {

**return** length **\*** width**;**

    }

}

class **Circle** *implements* *Shape* {

*private* *double* radius**;**

*public* Circle(*double* **radius**) {

*this***.***radius* **=** radius**;**

    }

    @**Override**

*public* *double* getArea() {

**return** Math**.***PI* **\*** radius **\*** radius**;**

    }

}

class **Triangle** *implements* *Shape* {

*private* *double* base**;**

*private* *double* height**;**

*public* Triangle(*double* **base,** *double* **height**) {

*this***.***base* **=** base**;**

*this***.***height* **=** height**;**

    }

    @**Override**

*public* *double* getArea() {

**return** 0.5 **\*** base **\*** height**;**

    }

}

*public* class **ShapeCalculator** {

*public* *static* *void* main(String[] **args**) {

        Shape rectangle **=** **new** Rectangle(10**,** 5)**;**

        Shape circle **=** **new** Circle(7)**;**

        Shape triangle **=** **new** Triangle(10**,** 5)**;**

        System**.***out***.**println("Rectangle Area: " **+** rectangle**.**getArea())**;**

        System**.***out***.**println("Circle Area: " **+** circle**.**getArea())**;**

        System**.***out***.**println("Triangle Area: " **+** triangle**.**getArea())**;**

    }

}

**Output:**

**A computer screen with text on it

Description automatically generated**

**3. Ans:**

interface **Animal** {

*void* bark()**;**

}

*public* class **Dog** *implements* *Animal* {

    @**Override**

*public* *void* bark() {

        System**.***out***.**println("Dog is barking")**;**

    }

*public* *static* *void* main(String[] **args**) {

        Dog dog **=** **new** Dog()**;**

        dog**.**bark()**;**

    }

}

**Output:**

**A computer screen with text

Description automatically generated with medium confidence**

**4.Ans:**

*public* class **StringReversal** {

*public* *static* String reverse(String **str**) {

**if** (str**.**isEmpty()) {

**return** str**;**

        }

**return** reverse(str**.**substring(1)) **+** str**.**charAt(0)**;**

    }

*public* *static* *void* main(String[] **args**) {

        String original **=** "Hello, World!"**;**

        String reversed **=** reverse(original)**;**

        System**.***out***.**println("Original: " **+** original)**;**

        System**.***out***.**println("Reversed: " **+** reversed)**;**

    }

}

**Output:**

**A black screen with white text

Description automatically generated**

**5. Ans:**

*public* class **Person** *implements* *Cloneable* {

*private* String name**;**

*private* *int* age**;**

*public* Person(String **name,** *int* **age**) {

*this***.***name* **=** name**;**

*this***.***age* **=** age**;**

    }

    @**Override**

*protected* Object clone() *throws* CloneNotSupportedException {

**return** *super***.**clone()**;**

    }

    @**Override**

*public* String toString() {

**return** "Person{name='" **+** name **+** "', age=" **+** age **+** "}"**;**

    }

*public* *static* *void* main(String[] **args**) {

**try** {

            Person person1 **=** **new** Person("John Doe"**,** 30)**;**

            Person person2 **=** (Person) person1**.**clone()**;**

            System**.***out***.**println("Original: " **+** person1)**;**

            System**.***out***.**println("Clone: " **+** person2)**;**

        } **catch** (CloneNotSupportedException **e**) {

            e**.**printStackTrace()**;**

        }

    }

}

**Output:**

**A computer screen shot of text

Description automatically generated**