NAME – TAPABRATA BANERJEE

USN- 22BTRAD030

SCALA PROGRAMMING

Q. Write a Scala program that creates an abstract class Shape with an abstract method area. Implement subclasses Rectangle and Circle that override the area method.

```scala
abstract class Shape {
  def area: Double
}
class Rectangle(width: Double, height: Double) extends Shape {
  def area: Double = width * height
}
class Circle(radius: Double) extends Shape {
   def area: Double = math.Pi * radius * radius
}
object ShapeApp {
  def main(args: Array[String]): Unit = {
  val rectangle = new Rectangle(7, 5)
  println(s"Rectangle Area: ${rectangle.area}")
  val circle = new Circle(4.5)
  println(s"Circle Area: ${circle.area}")
  }
}
```

Output:

Rectangle Area: 35.0
Circle Area: 63.61725123519331

Q. Write a Scala program that creates a class called Person with properties like name, age and country. Implement methods to get and set properties.

```scala
class Person(var name: String, var age: Int, var country: String) {
  def getName: String = name
  def setName(newName: String): Unit = {
  name = newName
  }
  def getAge: Int = age
  def setAge(newAge: Int): Unit = {
  age = newAge
  }
  def getCountry: String = country
  def setCountry(newCountry: String): Unit = {
  country = newCountry
  }
}
object PersonApp {
  def main(args: Array[String]): Unit = {
  val person = new Person("Tapabrata Banerjee", 20, "India")
  println("Original Person:")
  println(s"Name: ${person.getName}")
  println(s"Age: ${person.getAge}")
  println(s"Country: ${person.getCountry}")
  person.setName("Osama bin Laden")
  person.setAge(54)
  person.setCountry("PAKISTAN")
  println("\nUpdated Person:")
  println(s"Name: ${person.getName}")
  println(s"Age: ${person.getAge}")
  println(s"Country: ${person.getCountry}")
  }
}
```

Output:

Original Person:
Name: Tapabrata Banerjee
Age: 20
Country: India

Updated Person:
Name: Osama bin Laden
Age: 54
Country: PAKISTAN