

# Scala programming

Name: K shreeshanth.

Usn: 22BTRADO17.

Date:11-09-2023.

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

Code;

```
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("Andrey Ira", 35, "France")
    println("Original Person:")
    println(s"Name: ${person.getName}")
    println(s"Age: ${person.getAge}")
    println(s"Country: ${person.getCountry}")
    person.setName("Lior Daniela")
    person.setAge(30)
    person.setCountry("Canada")
    println("\nUpdated Person:")
    println(s"Name: ${person.getName}")
    println(s"Age: ${person.getAge}")
    println(s"Country: ${person.getCountry}")
  }
}
```

Output:

The screenshot shows the Scala Online Compiler interface. The code editor contains the following Scala code:

```
1 class Person(var name: String, var age: Int, var country: String) {
2   def getName: String = name
3   def setName(newName: String): Unit = {
4     name = newName
5   }
6   def getAge: Int = age
7   def setAge(newAge: Int): Unit = {
8     age = newAge
9   }
10  def getCountry: String = country
11  def setCountry(newCountry: String): Unit = {
12    country = newCountry
13  }
14 }
15 object PersonApp {
16   def main(args: Array[String]): Unit = {
17     val person = new Person("Andrey Ira", 35, "France")
18     println("Original Person:")
19     println(s"Name: ${person.getName}")
20     println(s"Age: ${person.getAge}")
21     println(s"Country: ${person.getCountry}")
22     person.setName("k shreeshanth")
23     person.setAge(19)
24     person.setCountry("Bharath")
25   }
26 }
```

The output panel shows the following output:

```
STDIN
Input for the program (Optional)

Output:
Original Person:
Name: Andrey Ira
Age: 35
Country: France

Updated Person:
Name: k shreeshanth
Age: 19
Country: Bharath
```

The interface includes a header with the OneCompiler logo, navigation links (Q&A, POSTS, CHALLENGES, MORE), and a LOGIN button. The footer contains a prompt to write, run, and share Scala code online using OneCompiler's Scala online compiler for free, along with a link to Adobe Stock and a button to get 10 free images.

The screenshot shows the Scala Online Compiler interface. The code editor contains the following Scala code:

```
1 class Person(var name: String, var age: Int, var country: String) {
2   def getName: String = name
3   def setName(newName: String): Unit = {
4     name = newName
5   }
6   def getAge: Int = age
7   def setAge(newAge: Int): Unit = {
8     age = newAge
9   }
10  def getCountry: String = country
11  def setCountry(newCountry: String): Unit = {
12    country = newCountry
13  }
14 }
15 object PersonApp {
16   def main(args: Array[String]): Unit = {
17     val person = new Person("Andrey Ira", 35, "France")
18     println("Original Person:")
19     println(s"Name: ${person.getName}")
20     println(s"Age: ${person.getAge}")
21     println(s"Country: ${person.getCountry}")
22     person.setName("k shreeshanth")
23     person.setAge(19)
24     person.setCountry("Bharath")
25     println("\nUpdated Person:")
26     println(s"Name: ${person.getName}")
27     println(s"Age: ${person.getAge}")
28     println(s"Country: ${person.getCountry}")
29   }
30 }
```

The output panel shows the following output:

```
STDIN
Input for the program (Optional)

Output:
Original Person:
Name: Andrey Ira
Age: 35
Country: France

Updated Person:
Name: k shreeshanth
Age: 19
Country: Bharath
```

The interface includes a header with the OneCompiler logo, navigation links (Q&A, POSTS, CHALLENGES, MORE), and a LOGIN button. The footer contains a prompt to write, run, and share Scala code online using OneCompiler's Scala online compiler for free, along with a link to Adobe Stock and a button to get 10 free images.