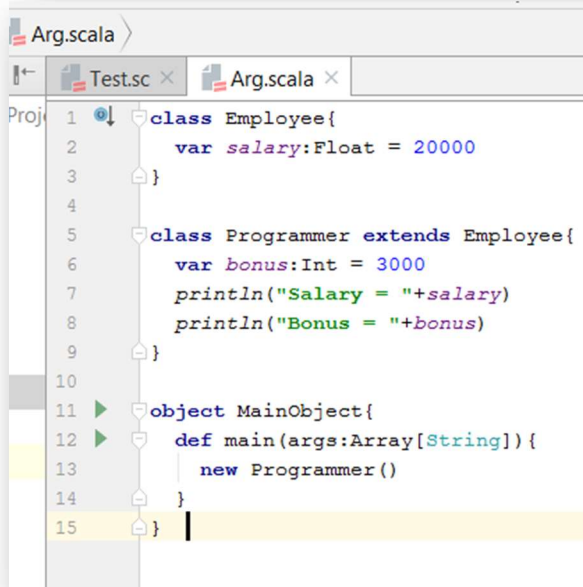


## Assignment15.1

1. Write a simple program to show inheritance in Scala.

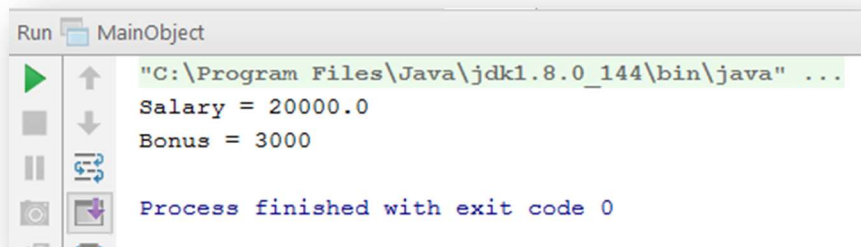
In this program, we are performing an inheritance of a class into another class. In the class **Employee** we have defined **Salary** as float, and inherited **Employee** into the class **Programmer** where we have defined **Bonus** as Int.



```
Arg.scala
Test.sc x Arg.scala x
class Employee{
  var salary:Float = 20000
}
class Programmer extends Employee{
  var bonus:Int = 3000
  println("Salary = "+salary)
  println("Bonus = "+bonus)
}
object MainObject{
  def main(args:Array[String]){
    new Programmer()
  }
}
```

### Output:

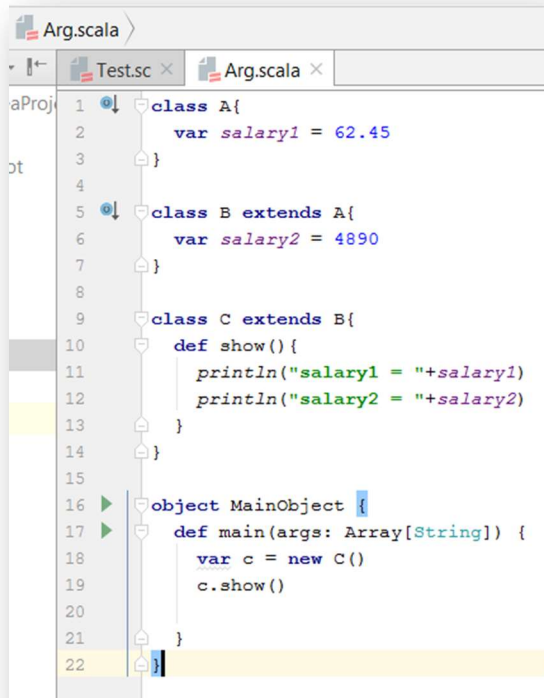
You will notice that Salary gives an output with decimals whereas Bonus gives an output as proper number.



```
Run MainObject
"C:\Program Files\Java\jdk1.8.0_144\bin\java" ...
Salary = 20000.0
Bonus = 3000
Process finished with exit code 0
```

2. Write a simple program to show multiple inheritance in Scala.

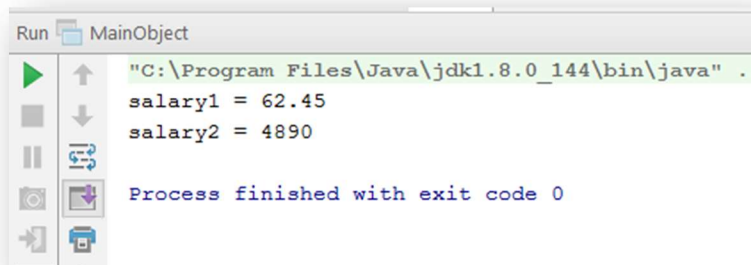
In this program, we are performing multiple inheritance. We are going to inherit **Class A** -> **Class B** and **Class B** -> **Class C**. We haven't specified any object type for the variables **Salary1** and **Salary2**.



```
Arg.scala
Test.sc x Arg.scala x
1 class A{
2     var salary1 = 62.45
3 }
4
5 class B extends A{
6     var salary2 = 4890
7 }
8
9 class C extends B{
10     def show(){
11         println("salary1 = "+salary1)
12         println("salary2 = "+salary2)
13     }
14 }
15
16 object MainObject {
17     def main(args: Array[String]) {
18         var c = new C()
19         c.show()
20     }
21 }
22 }
```

### Output:

You will notice that the output has decimals as well as proper number since we haven't defined object type for the variables.



```
Run MainObject
"C:\Program Files\Java\jdk1.8.0_144\bin\java" ..
salary1 = 62.45
salary2 = 4890
Process finished with exit code 0
```