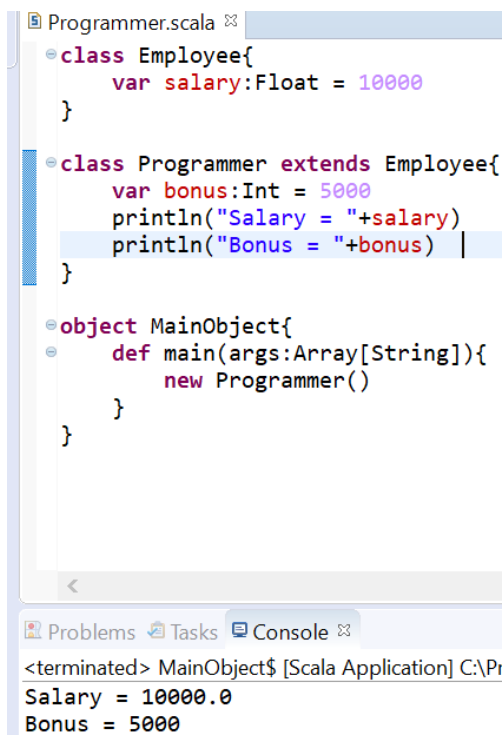


Task 1 : show inheritance in scala

```
class Employee{
    var salary:Float = 10000
}

class Programmer extends Employee{
    var bonus:Int = 5000
    println("Salary = "+salary)
    println("Bonus = "+bonus)
}

object MainObject{
    def main(args:Array[String]){
        new Programmer()
    }
}
```



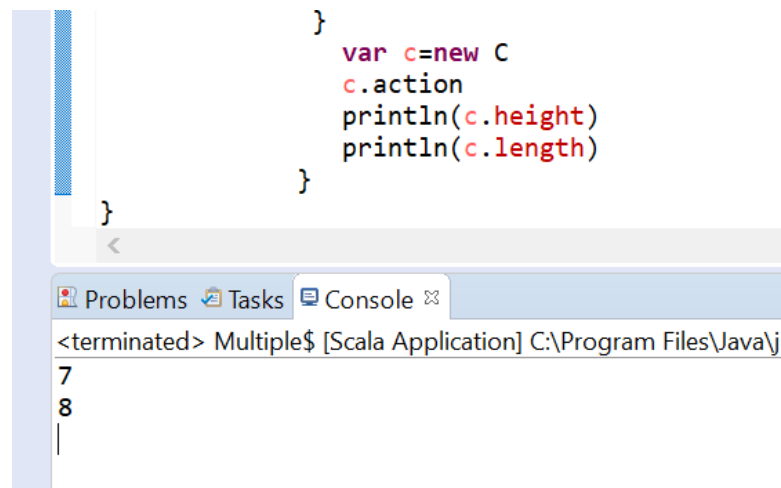
The screenshot shows an IDE window titled "Programmer.scala". The code is the same as in the previous block. The "Console" tab at the bottom shows the output of the program:

```
<terminated> MainObject$ [Scala Application] C:\Pr
Salary = 10000.0
Bonus = 5000
```

Task 2 : Show multiple inheritance

```
object Multiple{
  def main(args: Array[String]):Unit={
    trait A{
      var length:Int= _
      def action={
        length=length+5
      }
    }
    trait B{
      var height:Int = _
      def action={
        height = height + 1
      }
    }
    class C extends A with B{
      length=3;
      height+=6;
      override def action={
        super[A].action
        super[B].action
      }
    }
    var c=new C
    c.action
    println(c.height)
    println(c.length)
  }
}
```

```
    }
    var c=new C
    c.action
    println(c.height)
    println(c.length)
  }
}
```



The screenshot shows an IDE window with a Scala file. The code defines a Scala object named 'Multiple' with a 'main' method. Inside 'main', it defines two traits, 'A' and 'B'. Trait 'A' has a 'length' variable and an 'action' method that increments 'length' by 5. Trait 'B' has a 'height' variable and an 'action' method that increments 'height' by 1. A class 'C' is defined that extends both 'A' and 'B'. 'C' initializes 'length' to 3 and 'height' to 6. It overrides the 'action' method to call both 'super[A].action' and 'super[B].action'. In the 'main' method, an instance 'c' of class 'C' is created, 'c.action' is called, and the values of 'c.height' and 'c.length' are printed. The IDE's console shows the output of the program: '7' followed by '8' on the next line. The console title bar indicates the application is 'Multiple\$ [Scala Application]' running in 'C:\Program Files\Java\j'.

<terminated> Multiple\$ [Scala Application] C:\Program Files\Java\j
7
8
|

Task 4 : Print prices of courses

```
class MatchExample {  
  def printCourseInfo(course: String) =  
  {  
    course match {  
      case "Android App Development" => println("14,999 INR");  
      case "Data Science" => println("49,999 INR");  
      case "Big Data Hadoop & Spark Developer" => println("24,999 INR");  
      case "Blockchain Certification" => println("49,999 INR");  
      case _ => println("we are sorry, As of now we dont have this course");  
    }  
  }  
}  
  
object Main{  
  def main(args:Array[String]){  
    val matchObject=new MatchExample();  
    matchObject.printCourseInfo("Big Data Hadoop & Spark Developer");  
    matchObject.printCourseInfo("Data structures");  
  }  
}
```

```
object Main{  
  def main(args:Array[String]){  
    val matchObject=new MatchExample();  
    matchObject.printCourseInfo("Big Data Hadoop & Spark Developer");  
    matchObject.printCourseInfo("Data structures");  
  }  
}
```

Problems Tasks Console

<terminated> Main\$ [Scala Application] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (30-Jan-2019, 1

24,999 INR

we are sorry, As of now we dont have this course