

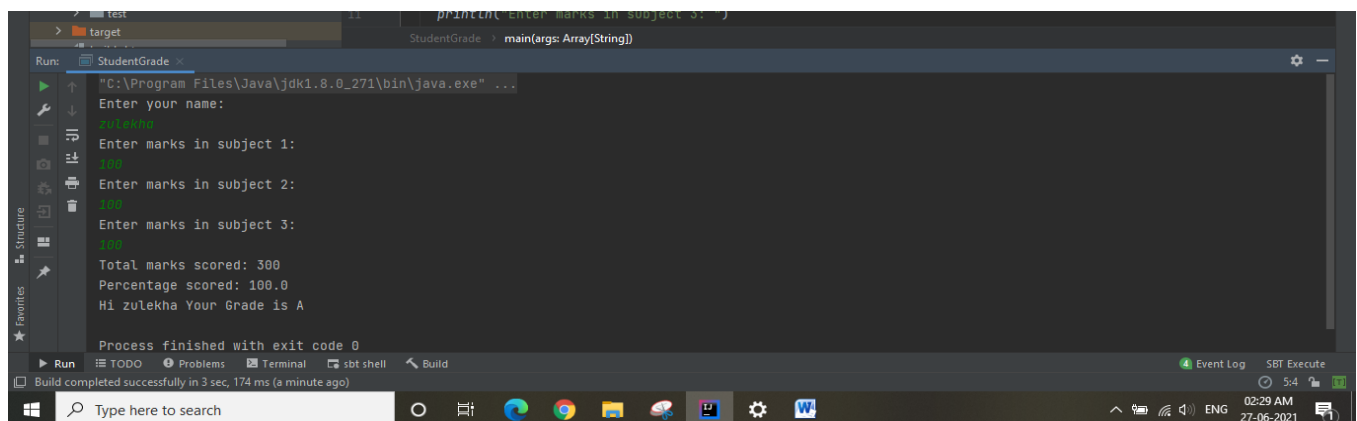
Scala coding assignment-1

1. WAP to calculate student grade based on below rules
 - a. 90 to 100 -> Grade A
 - b. 80 to 90 -> Grade B
 - c. 70 to 80 -> Grade C
 - d. Failure.

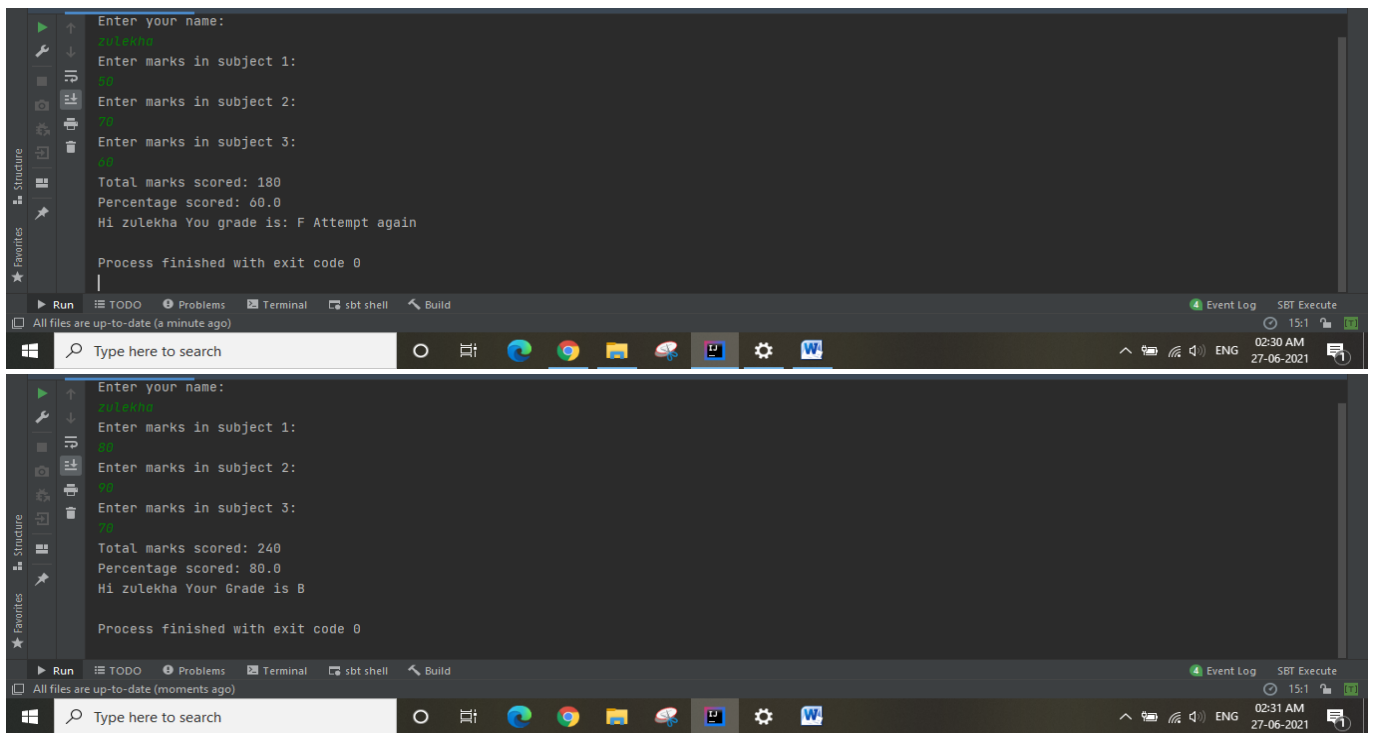
Program:

```
object StudentGrade {  
  def main(args: Array[String]): Unit = {  
  
    println("Enter your name: ")  
    val name = scala.io.StdIn.readLine()  
    println("Enter marks in subject 1: ")  
    var m1 = scala.io.StdIn.readInt()  
    println("Enter marks in subject 2: ")  
    var m2 = scala.io.StdIn.readInt()  
    println("Enter marks in subject 3: ")  
    var m3 = scala.io.StdIn.readInt()  
  
    var sum: Int = m1 + m2 + m3;  
    val per: Float = sum/3  
    var Grade: String = ""  
    println("Total marks scored: "+sum)  
    println("Percentage scored: "+per)  
  
    if(per >= 90) {  
      Grade = "A"  
      println("Hi "+name+" Your Grade is " + Grade)  
    }  
    else if (per >= 80 & per < 90) {  
      Grade = "B"  
      println("Hi "+name+" Your Grade is " + Grade)  
    }  
  
    else if (per >= 70 & per < 80) {  
      Grade = "C"  
      println("Hi "+name+" Your Grade is " + Grade)  
    }  
  
    else {  
      Grade = "F"  
      println("Hi "+name+" You grade is: " + Grade + " Attempt again")  
    }  
  }  
}
```

Output:



```
Run: StudentGrade  
"C:\Program Files\Java\jdk1.8.0_271\bin\java.exe" ...  
Enter your name:  
zulekha  
Enter marks in subject 1:  
100  
Enter marks in subject 2:  
100  
Enter marks in subject 3:  
100  
Total marks scored: 300  
Percentage scored: 100.0  
Hi zulekha Your Grade is A  
Process finished with exit code 0  
Build completed successfully in 3 sec, 174 ms (a minute ago)
```



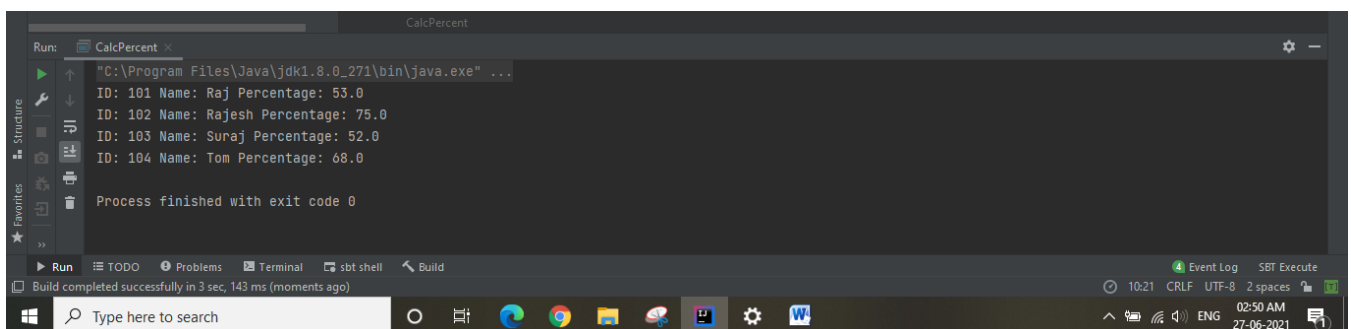
- WAP to calculate maximum % scored student report from below data.
 {id:101,name:raj,cmarks:45,pmarks:55,mmarks:67}
 {id:102,name:rajesh,cmarks:65,pmarks:85,mmarks:77}
 {id:103,name:suraj,cmarks:43,pmarks:55,mmarks:60}
 {id:104,name:tom,cmarks:71,pmarks:65,mmarks:70}

Program:

```
class Percentage
{
    def calculate(id:Int, name: String, cmarks: Int, pmarks: Int, mmarks: Int ): Unit =
    {
        var per: Float = (cmarks+pmarks+mmarks)/3
        println("ID: "+id+" Name: "+name+" Percentage: "+per)
    }
}

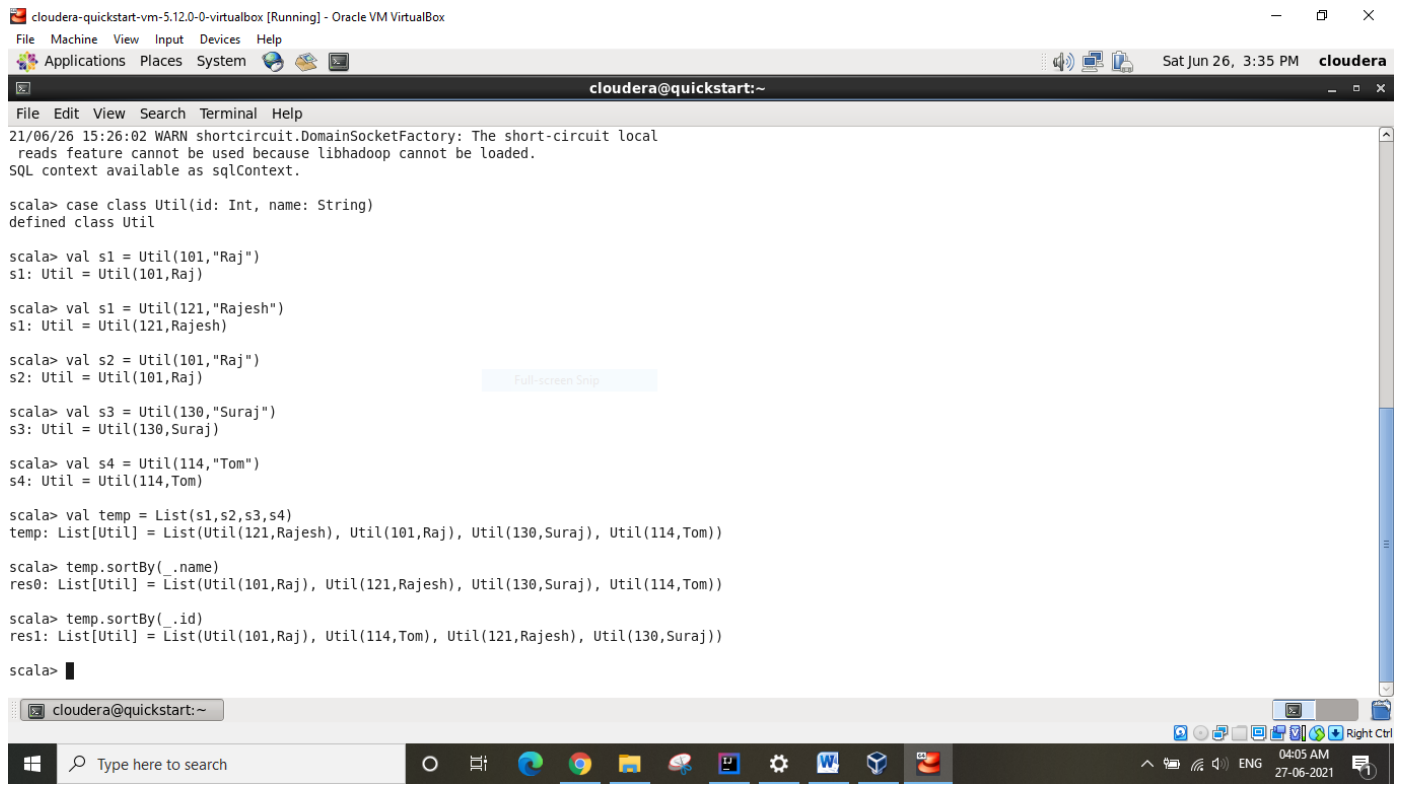
object CalcPercent {
    def main(args: Array[String]): Unit = {
        var P1=new Percentage()
        P1.calculate(101,"Raj",45,55,60)
        P1.calculate(102, "Rajesh",65,85,77)
        P1.calculate(103,"Suraj",43,55,60)
        P1.calculate(104,"Tom", 75,61,70)
    }
}
```

Output:



3. WAP to perform sorting of below data based on id and name(create class, object and a method for sorting in util class)

```
{id:101,name:raj}  
{id:121,name:rajesh}  
{id:130,name:suraj}  
{id:114,name:tom}
```



The screenshot shows a terminal window titled "cloudera-quickstart-vm-5.12.0-0-virtualbox [Running] - Oracle VM VirtualBox". The terminal output includes a warning message about libhadoop not being loaded, followed by Scala code that defines a Util class, creates four objects (s1, s2, s3, s4), and sorts a list of these objects by name and then by id. The results of the sorting are displayed as List[Util] objects.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
21/06/26 15:26:02 WARN shortcircuit.DomainSocketFactory: The short-circuit local  
reads feature cannot be used because libhadoop cannot be loaded.  
SQL context available as sqlContext.  
  
scala> case class Util(id: Int, name: String)  
defined class Util  
  
scala> val s1 = Util(101,"Raj")  
s1: Util = Util(101,Raj)  
  
scala> val s2 = Util(121,"Rajesh")  
s2: Util = Util(121,Rajesh)  
  
scala> val s3 = Util(130,"Suraj")  
s3: Util = Util(130,Suraj)  
  
scala> val s4 = Util(114,"Tom")  
s4: Util = Util(114,Tom)  
  
scala> val temp = List(s1,s2,s3,s4)  
temp: List[Util] = List(Util(121,Rajesh), Util(101,Raj), Util(130,Suraj), Util(114,Tom))  
  
scala> temp.sortBy(_.name)  
res0: List[Util] = List(Util(101,Raj), Util(121,Rajesh), Util(130,Suraj), Util(114,Tom))  
  
scala> temp.sortBy(_.id)  
res1: List[Util] = List(Util(101,Raj), Util(114,Tom), Util(121,Rajesh), Util(130,Suraj))  
  
scala> █
```