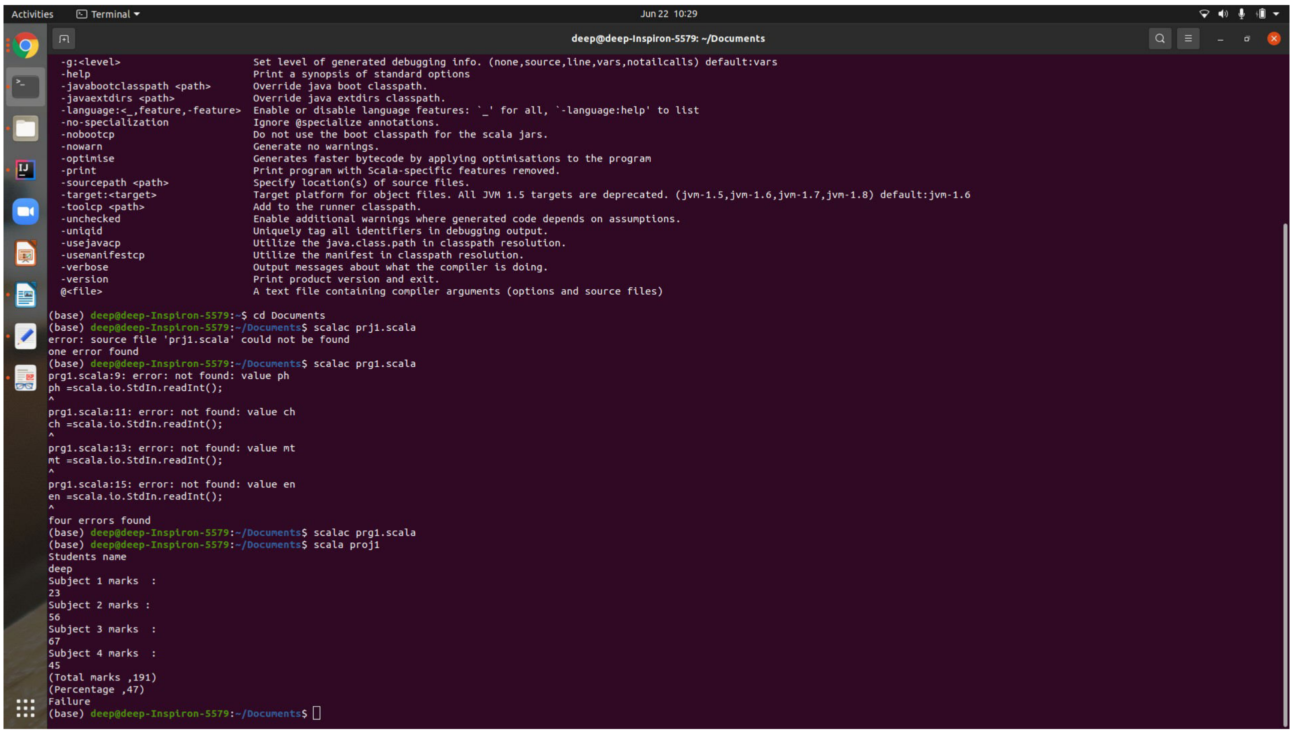
Scala coding assignment -1
Deep bhatt
22 June 2021

Q1 :
ans :

```
object proj1
{
def main(arg: Array[String]){
var gr, nm: String = "";
var s1, s2, s3, s4, tot, per: Int = 0;
println("Students name");
nm =scala.io.StdIn.readLine();
println("Subject 1 marks  : ");
ph =scala.io.StdIn.readInt();
println("Subject 2 marks : ");
ch =scala.io.StdIn.readInt();
println("Subject 3 marks  : ");
mt =scala.io.StdIn.readInt();
println("Subject 4 marks  : ");
en =scala.io.StdIn.readInt();
tot = s1+s2+s3+s4;
per = tot /4;
println("Total marks ",tot);
println("Percentage ",per);
if (per >= 90)
gr ="Grade A";
else
if (per >= 80)
gr = "Grade B";
else
if (per >= 70)
gr = "Grade C";
else
gr = "Failure";
println(gr);
}
}
```

OUTPUT:



```
  -g:<level>                      Set level of generated debugging info. (none,source,line,vars,notailcalls) default:vars
  -help                           Print a synopsis of standard options
  -javabootclasspath <path>       Override java boot classpath.
  -javaextdirs <path>             Override java extdirs classpath.
  -language:<_,feature,-feature>  Enable or disable language features: `_' for all, `-language:help' to list
  -no-specialization              Ignore @specialize annotations.
  -nobootcp                       Do not use the boot classpath for the scala jars.
  -nowarn                         Generate no warnings.
  -optimise                       Generates faster bytecode by applying optimisations to the program
  -print                          Print program with Scala-specific features removed.
  -sourcepath <path>              Specify location(s) of source files.
  -target:<target>                Target platform for object files. All JVM 1.5 targets are deprecated. (jvm-1.5,jvm-1.6,jvm-1.7,jvm-1.8) default:jvm-1.6
  -toolcp <path>                  Add to the runner classpath.
  -unchecked                      Enable additional warnings where generated code depends on assumptions.
  -uniqid                         Uniquely tag all identifiers in debugging output.
  -usejavacp                      Utilize the java.class.path in classpath resolution.
  -usemanifestcp                  Utilize the manifest in classpath resolution.
  -verbose                        Output messages about what the compiler is doing.
  -version                        Print product version and exit.
  @<file>                         A text file containing compiler arguments (options and source files)

(base) deep@deep-Inspiron-5579:~$ cd Documents
(base) deep@deep-Inspiron-5579:~/Documents$ scalac prj1.scala
error: source file 'prj1.scala' could not be found
one error found
(base) deep@deep-Inspiron-5579:~/Documents$ scalac prg1.scala
prg1.scala:9: error: not found: value ph
ph =scala.io.StdIn.readInt();
^
prg1.scala:11: error: not found: value ch
ch =scala.io.StdIn.readInt();
^
prg1.scala:13: error: not found: value mt
mt =scala.io.StdIn.readInt();
^
prg1.scala:15: error: not found: value en
en =scala.io.StdIn.readInt();
^
four errors found
(base) deep@deep-Inspiron-5579:~/Documents$ scalac prg1.scala
(base) deep@deep-Inspiron-5579:~/Documents$ scala proj1
Students name
deep
Subject 1 marks  :
23
Subject 2 marks :
56
Subject 3 marks  :
67
Subject 4 marks  :
45
(Total marks ,191)
(Percentage ,47)
Failure
(base) deep@deep-Inspiron-5579:~/Documents$ 
```

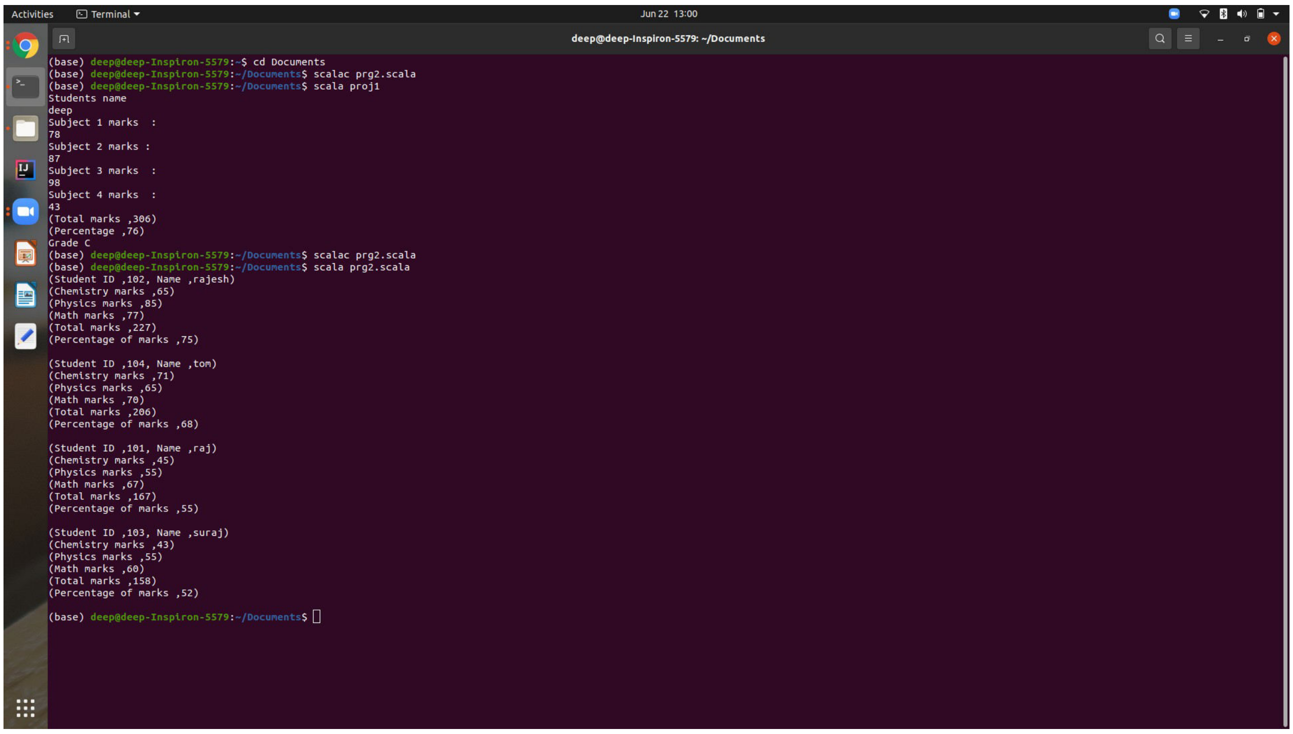
Q2:

Ans :

```
class Students{
 var id, cmarks, pmarks, mmarks, tot, per: Int = 0;
 var name: String = "";
 def SetVal(i:Int, nm:String, cm:Int, pm:Int, mm:Int): Unit = {
 this.id = i;
 this.name = nm;
 this.cmarks = cm;
 this.pmarks = pm;
 this.mmarks = mm;
 this.tot = cm + pm + mm;
 per = this.tot /3;
 }
 def Show(): Unit = {
println("Student ID ",this.id," Name ",this.name);
println("Chemistry marks ",this.cmarks);
println("Physics marks ",this.pmarks);
println("Math marks ",this.mmarks);
println("Total marks ",this.tot);
println("Percentage of marks ",this.per);
 }
}
object SortPercent
{
def main(arg: Array[String]){
 var i,j : Int = 0;
 var stud = Array.fill[Students](4)(new Students());
 stud(0).SetVal(101,"raj",45,55,67);
 stud(1).SetVal(102,"rajesh",65,85,77);
 stud(2).SetVal(103,"suraj",43,55,60);
 stud(3).SetVal(104,"tom",71,65,70);
 var tmp = new Students();

 for(i <- 0 to 3)
 {
 for (j<- i to 3)
 {
 if (stud(i).tot < stud(j).tot)
 {
 tmp = stud(i);
 stud(i) = stud(j);
 stud(j) = tmp;
 }
 }
 }
 for( i <- 0 to 3)
 {
 stud(i).Show();
 println("");
 }
 }
}
```

OUTPUT :



```
(base) deep@deep-Inspiron-5579:~$ cd Documents
(base) deep@deep-Inspiron-5579:~/Documents$ scalac prg2.scala
(base) deep@deep-Inspiron-5579:~/Documents$ scala proj1
Students name
deep
Subject 1 marks  :
78
Subject 2 marks :
87
Subject 3 marks  :
98
Subject 4 marks  :
43
(Total marks ,306)
(Percentage ,76)
Grade C
(base) deep@deep-Inspiron-5579:~/Documents$ scalac prg2.scala
(base) deep@deep-Inspiron-5579:~/Documents$ scala prg2.scala
(Student ID ,102, Name ,rajesh)
(Chemistry marks ,65)
(Physics marks ,85)
(Math marks ,77)
(Total marks ,227)
(Percentage of marks ,75)

(Student ID ,104, Name ,tom)
(Chemistry marks ,71)
(Physics marks ,65)
(Math marks ,70)
(Total marks ,206)
(Percentage of marks ,68)

(Student ID ,101, Name ,raj)
(Chemistry marks ,45)
(Physics marks ,55)
(Math marks ,67)
(Total marks ,167)
(Percentage of marks ,55)

(Student ID ,103, Name ,suraj)
(Chemistry marks ,43)
(Physics marks ,55)
(Math marks ,60)
(Total marks ,158)
(Percentage of marks ,52)

(base) deep@deep-Inspiron-5579:~/Documents$ 
```

Q3:
Ans :

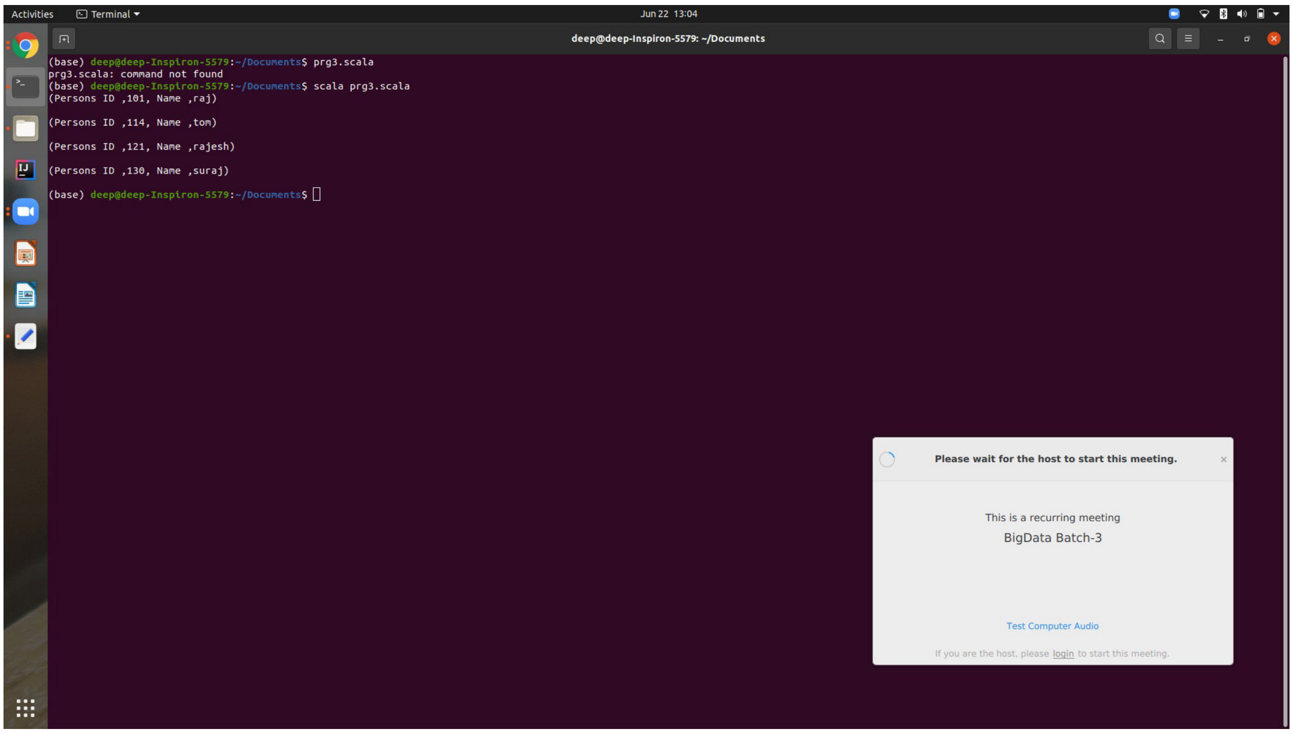
```
        class Person{
 var id: Int = 0;
 var name: String = "";
 def SetVal(i:Int, nm:String): Unit = {
 this.id = i;
 this.name = nm;
 }
 def Show(): Unit = {
println("Persons ID ",this.id," Name ",this.name);
 }
}
object SortObject
{

def main(arg: Array[String]){
 var i,j : Int = 0;
 var pers = Array.fill[Person](4)(new Person());
 pers(0).SetVal(101,"raj");
 pers(1).SetVal(121,"rajesh");
 pers(2).SetVal(130,"suraj");
 pers(3).SetVal(114,"tom");

 var tmp = new Person();

 for(i <- 0 to 3)
 {
 for (j<- i to 3)
 {
 if (pers(i).id > pers(j).id)
 {
 tmp = pers(i);
 pers(i) = pers(j);
 pers(j) = tmp;
 }
 }
 }
 for( i <- 0 to 3)
 {
 pers(i).Show();
 println("");
 }
 }
}
```

OUTPUT :



```
(base) deep@deep-Inspiron-5579:~/Documents$ prg3.scala
prg3.scala: command not found
(base) deep@deep-Inspiron-5579:~/Documents$ scala prg3.scala
(Persons ID ,101, Name ,raj)

(Persons ID ,114, Name ,tom)

(Persons ID ,121, Name ,rajesh)

(Persons ID ,130, Name ,suraj)

(base) deep@deep-Inspiron-5579:~/Documents$
```

Please wait for the host to start this meeting.

This is a recurring meeting

BigData Batch-3

Test Computer Audio

If you are the host, please login to start this meeting.