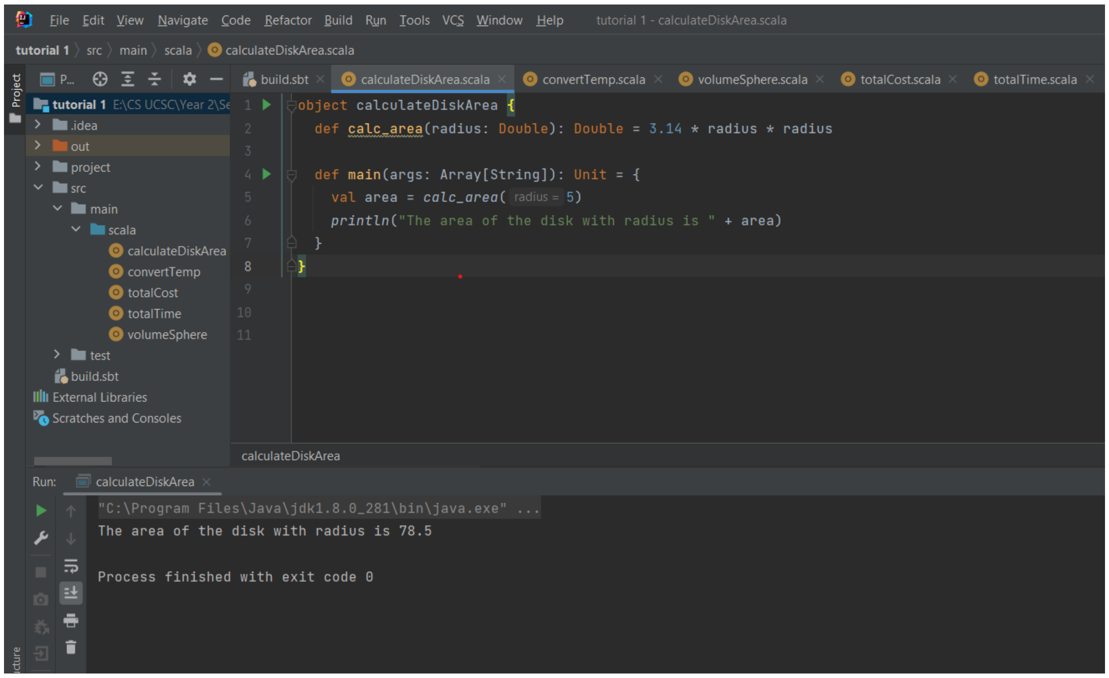
Functional Programming – Tutorial 1

Question 1



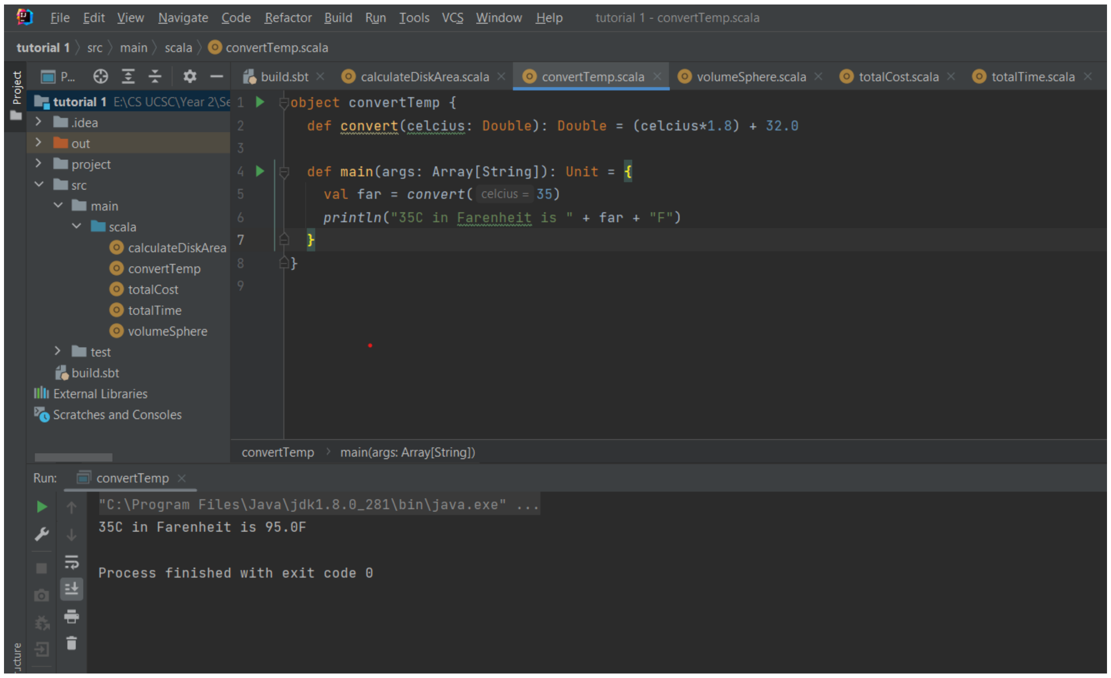
```scala
object calculateDiskArea {
  def calc_area(radius: Double): Double = 3.14 * radius * radius

  def main(args: Array[String]): Unit = {
    val area = calc_area( radius = 5)
    println("The area of the disk with radius is " + area)
  }
}
```

```
"C:\Program Files\Java\jdk1.8.0_281\bin\java.exe" ...
The area of the disk with radius is 78.5

Process finished with exit code 0
```

Question 2



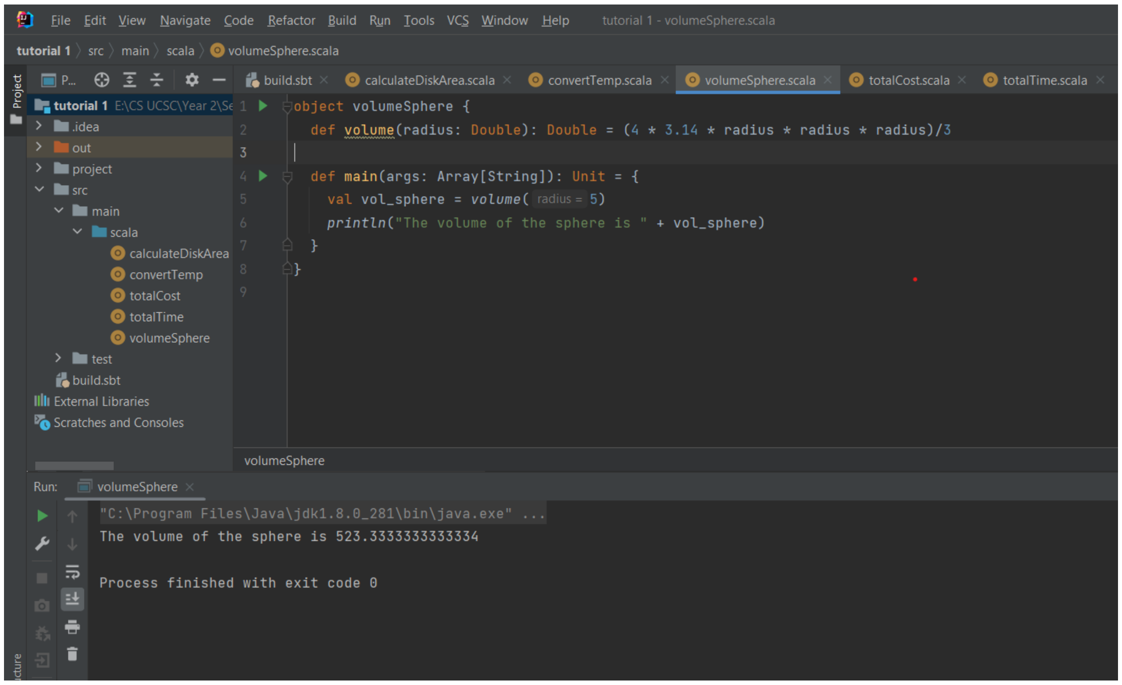
```scala
object convertTemp {
  def convert(celcius: Double): Double = (celcius*1.8) + 32.0

  def main(args: Array[String]): Unit = {
    val far = convert( celcius = 35)
    println("35C in Farenheit is " + far + "F")
  }
}
```

```
"C:\Program Files\Java\jdk1.8.0_281\bin\java.exe" ...
35C in Farenheit is 95.0F

Process finished with exit code 0
```

Question 3



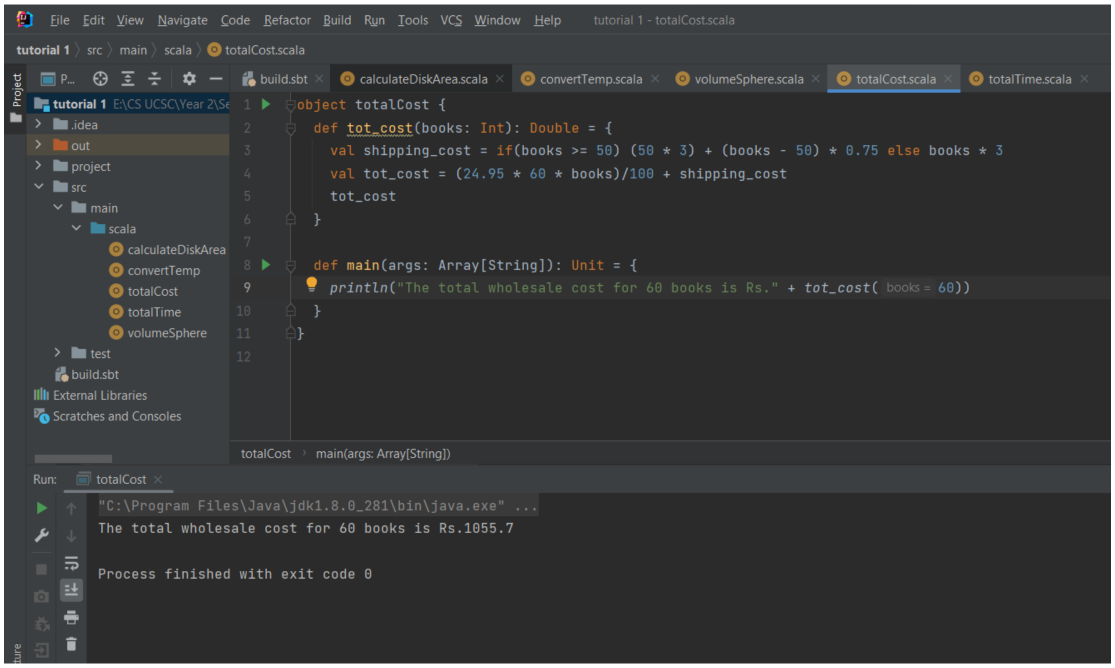
```scala
object volumeSphere {
  def volume(radius: Double): Double = (4 * 3.14 * radius * radius * radius)/3

  def main(args: Array[String]): Unit = {
    val vol_sphere = volume( radius = 5)
    println("The volume of the sphere is " + vol_sphere)
  }
}
```

```
"C:\Program Files\Java\jdk1.8.0_281\bin\java.exe" ...
The volume of the sphere is 523.3333333333334

Process finished with exit code 0
```

Question 4



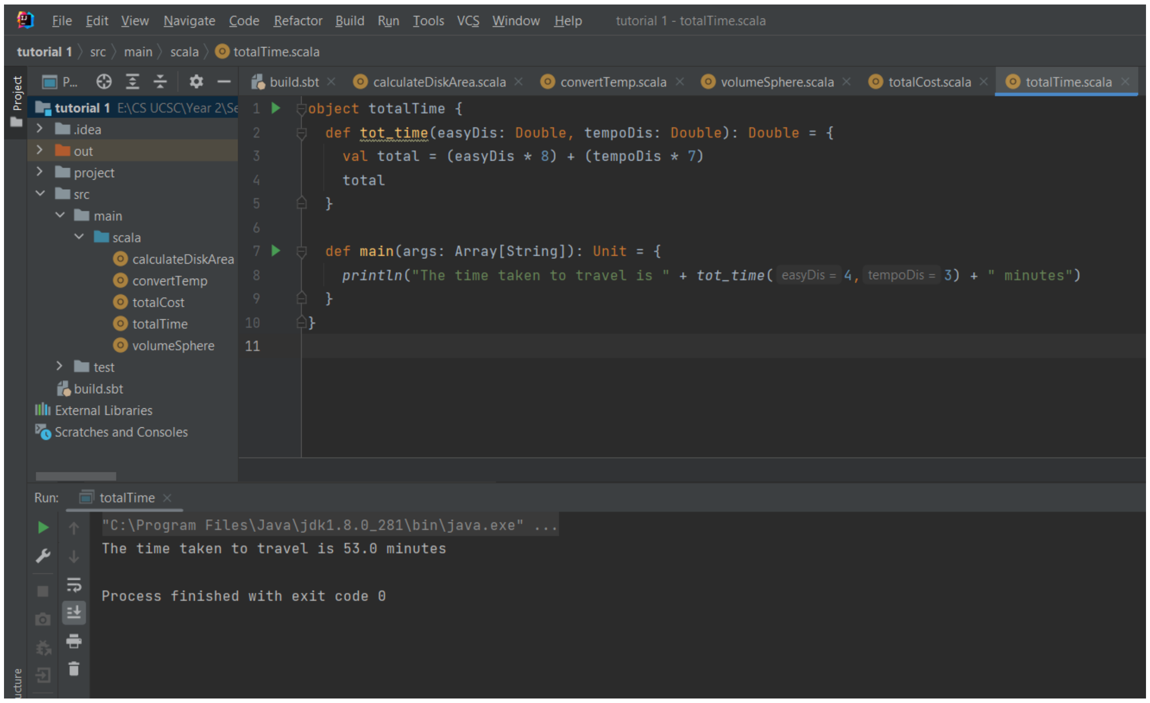
```scala
object totalCost {
  def tot_cost(books: Int): Double = {
    val shipping_cost = if(books >= 50) (50 * 3) + (books - 50) * 0.75 else books * 3
    val tot_cost = (24.95 * 60 * books)/100 + shipping_cost
    tot_cost
  }

  def main(args: Array[String]): Unit = {
    println("The total wholesale cost for 60 books is Rs." + tot_cost( books = 60))
  }
}
```

```
"C:\Program Files\Java\jdk1.8.0_281\bin\java.exe" ...
The total wholesale cost for 60 books is Rs.1055.7

Process finished with exit code 0
```

Question 5


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help tutorial 1 - totalTime.scala
tutorial 1 src main scala totalTime.scala
build.sbt calculateDiskArea.scala convertTemp.scala volumeSphere.scala totalCost.scala totalTime.scala
tutorial 1 E:\CS UCSC\Year 2\Se
.idea
out
project
src
main
scala
calculateDiskArea
convertTemp
totalCost
totalTime
volumeSphere
test
build.sbt
External Libraries
Scratches and Consoles
Run: totalTime

```scala
object totalTime {
  def tot_time(easyDis: Double, tempoDis: Double): Double = {
    val total = (easyDis * 8) + (tempoDis * 7)
    total
  }

  def main(args: Array[String]): Unit = {
    println("The time taken to travel is " + tot_time( easyDis = 4, tempoDis = 3) + " minutes")
  }
}
```

```
"C:\Program Files\Java\jdk1.8.0_281\bin\java.exe" ...
The time taken to travel is 53.0 minutes

Process finished with exit code 0
```