Task3:

Step1: Define method sum which take three integer arguments and sum them

def sum(a:Int, b:Int, c:Int) = a + b + c

Step2: Define partial function modifiedSum which refines sum function by making first argument constant value 5 and takes two arguments

def modifiedSum = sum(5, \_:Int, \_:Int)

Step2: Define function modifiedSquare which take a method callback as first parameter and two more integer parameters. It calls callback methd two two integer aguments and square the result

def modifiedSquare(callback : (Int, Int) => Int, x:Int, y:Int):Int = {

val z = callback(x,y)

z \* z

}

Step3: Call modifiedSqaure with modifiedSum as callback method and arguments 7 and 8

val p = modifiedSquare(modifiedSum, 7, 8)

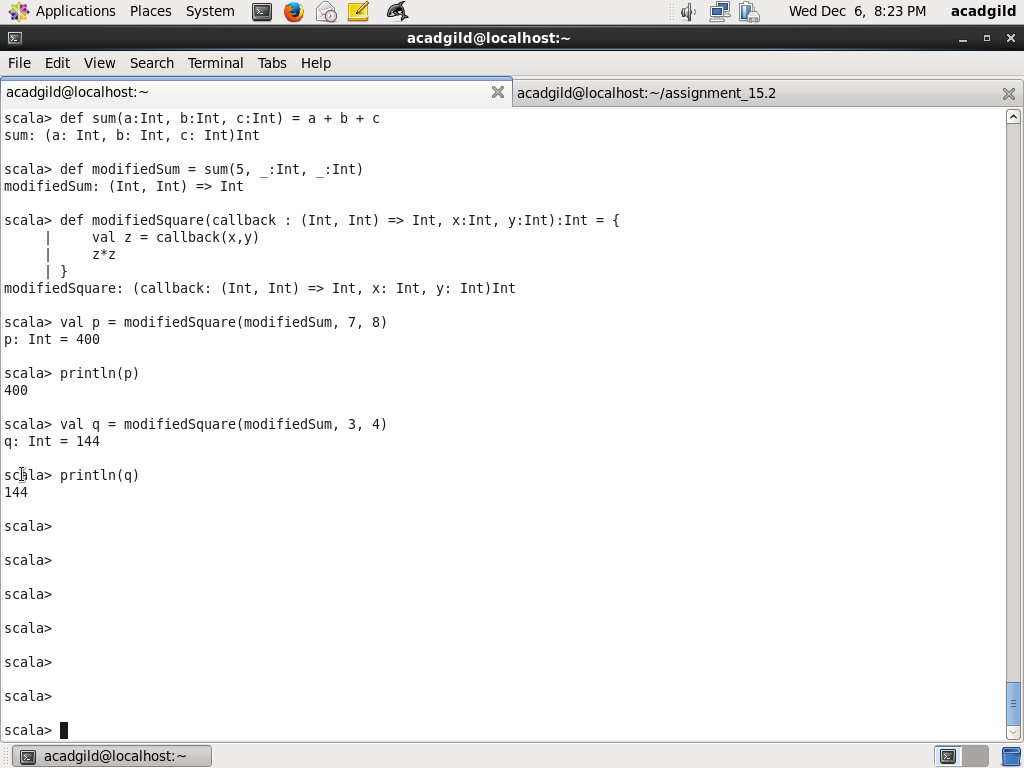
println(p)

Step3: Call modifiedSqaure with modifiedSum as callback method and arguments 3 and 4

val q = modifiedSquare(modifiedSum, 3, 4)

println(q)

Screenshot is as below:



Task4:

Step1: Define a function findPrice which takes subject having type String as argument and returns Int which the price for the subject. Use match clause to get the price of various subjects. Default value is -1, if the subject is other than Android, Big Data Development, Advanced Big Data Development, Spark.

Note: In the question, there are two same subject Big Data Development gives, if I try I am getting warning. So modified second one as Advacned Big Data Development

def findPrice(subject: String):Int = {

val price:Int = subject match {

case "Android" => 12999

case "Big Data Development" => 17999

case "Advanced Big Data Development" => 17999

case "Spark" => 19999

case \_ => -1

}

return price

}

Step2: Call the method with various subjects than Android, Big Data Development, Advanced Big Data Development, Spark, Java and return the corresponding price.

val p = findPrice("Android")

val p = findPrice("Big Data Development")

val p = findPrice("Advanced Big Data Development")

val p = findPrice("Spark")

val p = findPrice("Java")

Screenshot is as below:

