EXPERIMENT-8

Give the output for following:

1. println(“BatMaTSatRatIn”.drop(3).take(7).replace(“t”, “s”))

```
println(List(1,2,3).flatMap(x=>List(x,4, x*2)))
```

Ans: OUTPUT:-

Output:

```
MaTSasR
List(1, 4, 2, 2, 4, 4, 3, 4, 6)
```

2. def quadruple(x:Int):Int=x*4

```
val quadrupleCopy=quadruple _
println(quadrupleCopy(-1) + quadruple(2))
```

Ans: OUTPUT:-

Output:

```
4
```

3. val fruits = List("mango", "apple", "pear")

```
val fruits1=fruits.updated(1, "orange")
println(fruits.flatMap(_.toUpperCase))
println(fruits1.filter(_.take(1)==“o”))
```

Ans: OUTPUT:-

Output:

```
List(M, A, N, G, O, A, P, P, L, E, P, E, A, R)
List(orange)
```

4. println("Functionalprogram".take(6).toUpperCase.drop(3))

println(List('x', 'yy', 'zzz').flatMap(I=>List(i, i.length)))

Ans:- OUTPUT:-

Output:

```
CTI
List(x, 1, yy, 2, zzz, 3)
```

5. var rrr= List("ant", "beer", "battered", "cool", "burger")

rrr.filter {(w: String) =>w.take(1) == "b"}.reduceLeft{(a: String, b: String) =>s"$a $b"}

Ans: OUTPUT:-

Output:

```
beer battered burger
```

6. object Whatever{

def speak(something: String)(implicit nice: String) = {println(s"$something $nice")}

}

implicit val nice= "the walrus"

println{Whatever.speak("I am")}

```
println{Whatever.speak("I like")("catfood")}
```

Ans: OUTPUT:-

Output:

```
I am the walrus
()
I like catfood
()
```

7.
```
val s = "Scala programming is fun"

val result = s.split(" ").map(_.reverse) .mkString(" ")

println(result)
```

Ans: OUTPUT:-

Output:

```
alacS gnimmargorp si nuf
```

8.
```
val numbers = List(5, 10, 15, 20)

val result = numbers.reduce((x, y) => x * y)

println(result)
```

Ans: OUTPUT:-

Output:

```
15000
```