Mutable Collection - scala.collection.mutable.______

==================

i. val

1.

```
scala> val x = scala.collection.mutable.ArrayBuffer(2,4,6,8)
x: scala.collection.mutable.ArrayBuffer[Int] = ArrayBuffer(2, 4, 6, 8)
```

2.

```
scala> x+=10
res0: x.type = ArrayBuffer(2, 4, 6, 8, 10)
```

NOTE: WE ADDED NEW ELEMENT 10, BECAUSE IT IS MUTABLE

HAD IT BEEN A IMMUTABLE, YOU CAN NOT ADD IT

3.

```
scala> x.map(i=>i+1)
res1: scala.collection.mutable.ArrayBuffer[Int] = ArrayBuffer(3, 5, 7, 9, 11)


scala> x = x.map(i=>i+1)
<console>:8: error: reassignment to val
    x = x.map(i=>i+1)
```

NOTE: YOU CAN NOT REASSIGN TO VAL - VALUE

HAD IT BEEN A VAR, YOU CAN REASSIGN

ii. var

1.

```
scala> var x = scala.collection.mutable.ArrayBuffer(2,4,6,8)
x: scala.collection.mutable.ArrayBuffer[Int] = ArrayBuffer(2, 4, 6, 8)
```

2.

```
scala> x+=10
res0: x.type = ArrayBuffer(2, 4, 6, 8, 10)
```

NOTE: WE ADDED NEW ELEMENT 10, BECAUSE IT IS MUTABLE

HAD IT BEEN A IMMUTABLE, YOU CAN NOT ADD IT

3.

```
scala> x.map(i=>i+1)
res1: scala.collection.mutable.ArrayBuffer[Int] = ArrayBuffer(3, 5, 7, 9, 11)

scala> x = x.map(i=>i+1)
x: scala.collection.mutable.ArrayBuffer[Int] = ArrayBuffer(3, 5, 7, 9, 11)
```

NOTE: YOU CAN REASSIGN TO VAR

Immutable Collection - scala.collection.immutable._____

=================

i. val

1.

```
scala> val x = scala.collection.immutable.List(2,4,6,8)
x: List[Int] = List(2, 4, 6, 8)
```

2.

```
scala> x+=10
<console>:9: error: value += is not a member of List[Int]
         x+=10
          ^
```

NOTE: SINCE IT IS IMMUTABLE, WE CAN NOT ADD ONE MORE ELEMENT

BECAUSE += WHICH IS AN OPERATOR(BEHIND THE SCENES IT IS ACTUALLY A METHOD) TO ADD NEW ELEMENTS IS NOT AVAIALABLE IN SCALA.COLLECTION.IMMUTABLE

+= OPERATOR(OR METHOD) IS DEFINED ONLY IN SCALA.COLLECTION.MUTABLE

2.1.

NOTE: IF YOU WANT TO STILL ADD NEW ELEMENT 10 TO THE LIST, YOU CAN MAKE A NEW LIST AS BELOW

```
scala> val y = x:+10

y: List[Int] = List(2, 4, 6, 8, 10)


scala> x

res5: List[Int] = List(2, 4, 6, 8)


scala> y

res6: List[Int] = List(2, 4, 6, 8, 10)
```

3.

```
scala> x = x.map(i=>i+1)

<console>:8: error: reassignment to val

    x = x.map(i=>i+1)

      ^
```

NOTE: YOU CAN NOT REASSIGN TO VAL - VALUE

HAD IT BEEN A VAR, YOU CAN REASSIGN

ii. var

1.

```
scala> var x = scala.collection.immutable.List(2,4,6,8)
```

```
x: List[Int] = List(2, 4, 6, 8)
```

2.

```
scala> x+=10
<console>:9: error: value += is not a member of List[Int]
              x+=10
               ^
```

NOTE: SINCE IT IS IMMUTABLE, WE CAN NOT ADD ONE MORE ELEMENT

BECAUSE += WHICH IS AN OPERATOR(BEHIND THE SCENES IT IS ACTUALLY A METHOD) TO ADD NEW ELEMENTS IS NOT AVAIALABLE IN SCALA.COLLECTION.IMMUTABLE

+= OPERATOR(OR METHOD) IS DEFINED ONLY IN SCALA.COLLECTION.MUTABLE

2.1

NOTE: IF YOU WANT TO STILL ADD NEW ELEMENT 10 TO THE LIST, YOU CAN MAKE A NEW LIST AS BELOW

```
scala> var y = x:+10
y: List[Int] = List(2, 4, 6, 8, 10)

scala> x
res5: List[Int] = List(2, 4, 6, 8)

scala> y
res6: List[Int] = List(2, 4, 6, 8, 10)
```

2.2

IMP NOTE:you can still do it this way, because x is var and you can completely reassign some new value to the variable of type "var"

```
scala> x = x:+10
x: List[Int] = List(3, 5, 7, 9, 10)
```

3.

```
scala> x = x.map(i=>i+1)

x: List[Int] = List(3, 5, 7, 9)
```

NOTE: YOU CAN REASSIGN TO VAR