```
//Creating a list of tuples, where the 1st element of the tuple is an
//int and the second element is a string.

//SOLUTION
val tup = (
  (1,"alpha"),
  (2,"beta"),
  (3,"gamma"),
  (4,"zeta"),
  (5,"omega")
)

//Output
tup: ((Int, String), (Int, String), (Int, String), (Int, String),
(Int, String)) = ((1,alpha),(2,beta),(3,gamma),(4,zeta),(5,omega))



//   <<<<<------ TASK 1 ----->>>>>
//for the above list, print the numbers where the corresponding
//string length is 4

//SOLUTION 1

(Below statement prints Number as well as String)

val tup4 = tup.productIterator.map{case(k,v) =>
if(v.toString.length==4) (k,v) else null}.filter(x => x!=null)
tup4.foreach(println)

//Output
tup4: Iterator[(Any, Any)] = non-empty iterator
(2,beta)
(4,zeta)
res0: Unit = ()

(Below statement prints Number only)

var tup44 = tup.productIterator.map{case(k,v) =>
if(v.toString.length==4) k else null}.filter(x => x!=null)
tup44.foreach(println)

//Output
tup44: Iterator[Any] = non-empty iterator
2
4
res1: Unit = ()
```

```
//SOLUTION 2

(Below statement prints Number as well as String)

tup.productIterator.map{case(k,v) => if(v.toString.length==4) (k,v)
else null}.filter(x => x!=null).foreach(println)

//Output
(2,beta)
(4,zeta)
res2: Unit = ()

(Below statement prints Number only)

tup.productIterator.map{case(k,v) => if(v.toString.length==4) k else
null}.filter(x => x!=null).foreach(println)

//Output
2
4
res3: Unit = ()

// *********************************



//   <<<<------ TASK 2 ----->>>>>
//find the average of all numbers, where the corresponding string
//contains alphabet 'm' or alphabet 'z'

//SOLUTION 1
var tup9 = tup.productIterator.map{case(k,v) =>
if(v.toString.matches(".*m.*")||v.toString.matches(".*z.*")) k else
null}.filter(x => x!=null).toList.map(_.toString).map(_.toInt)
tup9.foreach(println)

tup9.reduce(_ + _)./(tup9.length)

//Output
tup9: List[Int] = List(3, 4, 5)
3
4
5
res4: Unit = ()

res5: Int = 4
```

```
//SOLUTION 2
tup.productIterator.map{case(k,v) =>
if(v.toString.matches(".*m.*")||v.toString.matches(".*z.*")) k else
null}.filter(x =>
x!=null).toList.map(_.toString).map(_.toInt).reduce(_ +
_)./(tup9.length)

//Output
res6: Int = 4

// **********************************
```