

Universidad Nacional Politécnica
Departamento de Ciencia

Ingeniería en Computación
Programación móvil

Nombre: Victor Manuel Leiva

Docente: Ing. Luis

6- Codelabs: Extensions vale 4 puntos

<https://codelabs.developers.google.com/codelabs/kotlin-bootcamp-extensions/>

Answer these questions

Question 1

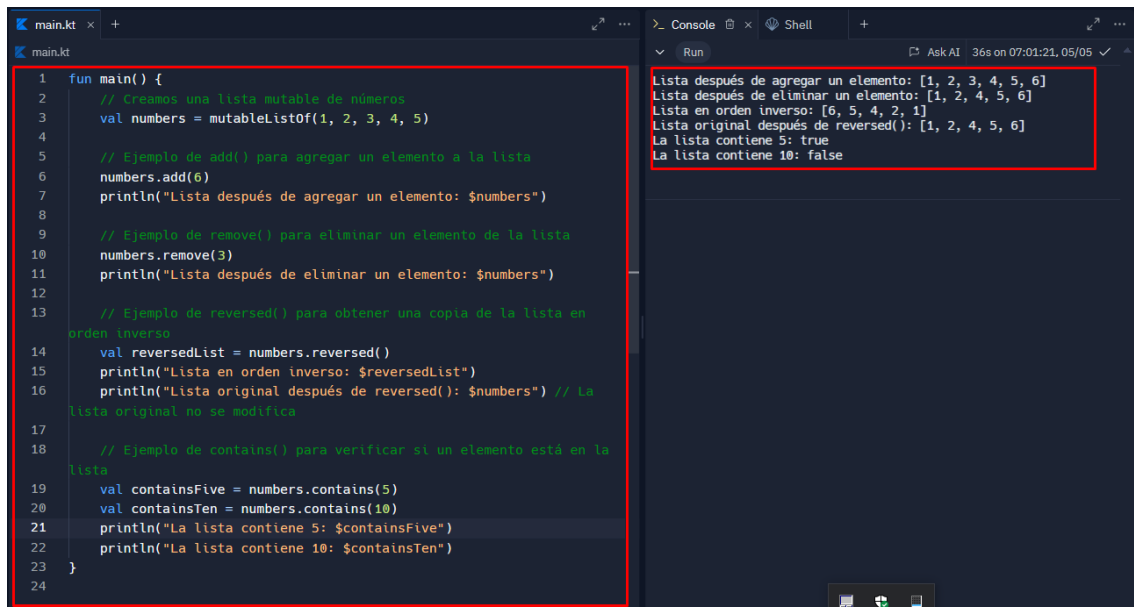
Which one of the following returns a copy of a list?

`add()`

`remove()`

`reversed()`

`contains()`



The screenshot shows a Kotlin IDE with a file named `main.kt`. The code defines a `main` function that demonstrates various list operations on a `MutableList` named `numbers`. The code includes comments in Spanish explaining each step: adding an element, removing an element, reversing the list, and checking for the presence of an element. The output in the console shows the state of the list after each operation, confirming that `reversed()` returns a new list without modifying the original.

```
1 fun main() {
2     // Creamos una lista mutable de números
3     val numbers = mutableListOf(1, 2, 3, 4, 5)
4
5     // Ejemplo de add() para agregar un elemento a la lista
6     numbers.add(6)
7     println("Lista después de agregar un elemento: $numbers")
8
9     // Ejemplo de remove() para eliminar un elemento de la lista
10    numbers.remove(3)
11    println("Lista después de eliminar un elemento: $numbers")
12
13    // Ejemplo de reversed() para obtener una copia de la lista en
orden inverso
14    val reversedList = numbers.reversed()
15    println("Lista en orden inverso: $reversedList")
16    println("Lista original después de reversed(): $numbers") // La
lista original no se modifica
17
18    // Ejemplo de contains() para verificar si un elemento está en la
lista
19    val containsFive = numbers.contains(5)
20    val containsTen = numbers.contains(10)
21    println("La lista contiene 5: $containsFive")
22    println("La lista contiene 10: $containsTen")
23 }
24
```

Console output:

```
Lista después de agregar un elemento: [1, 2, 3, 4, 5, 6]
Lista después de eliminar un elemento: [1, 2, 4, 5, 6]
Lista en orden inverso: [6, 5, 4, 2, 1]
Lista original después de reversed(): [1, 2, 4, 5, 6]
La lista contiene 5: true
La lista contiene 10: false
```

Question 2

Which one of these extension functions on class `AquariumPlant`(`val color: String`, `val size: Int`, `private val cost: Double`, `val leafy: Boolean`) will give a compiler error?

`fun AquariumPlant.isRed() = color == "red"`

`fun AquariumPlant.isBig() = size > 45`

`fun AquariumPlant.isExpensive() = cost > 10.00`

`fun AquariumPlant.isNotLeafy() = leafy == false`



```
1 class AquariumPlant(val color: String, val size: Int, private val
  cost: Double, val leafy: Boolean)
2
3 // Extensión para verificar si una planta es cara
4 fun AquariumPlant.isExpensive() = cost > 10.00 // Esto causará un
  error de compilación
5
6 fun main() {
7     val plant = AquariumPlant("green", 50, 15.0, true)
8
9     println("Is the plant expensive? ${plant.isExpensive()}") //
  Error de compilación aquí
10 }
11
```

main.kt:4:35: error: cannot access 'cost': it is private in 'AquariumPlant'

fun AquariumPlant.isExpensive() = cost > 10.00 // Esto causará un error de compilación

Question 3

Which one of the following is not a place where you can define constants with `const val`?

at the top level of a file

in regular classes

in singleton objects

in companion objects