



Universidad Nacional Politécnica

Programacion Movil

Estudiante:

- Adrian Eduardo Reynosa Flores

2010455

Docente:

- Ing. Luis Guido Calderon.

“Acumulado para el primer parcial”

3- Codelabs: Kotlin Basics vale 5 puntos

<https://developer.android.com/codelabs/kotlin-bootcamp-basics>

Responde estas preguntas

Pregunta 1

¿Cuál de las siguientes declara una lista inmutable de cadenas?

- ☐ `val school = arrayOf("shark", "salmon", "minnow")`
- ☐ `var school = arrayOf("shark", "salmon", "minnow")`
- ☒ `val school = listOf("shark", "salmon", "minnow")`
- ☐ `val school = mutableListOf("shark", "salmon", "minnow")`

Pregunta 2

¿Cuál será el resultado del siguiente código? `for (i in 3..8 step 2) print(i)`

- ☐ 345678
- ☐ 468
- ☐ 38
- ☒ 357

Pregunta 3

¿Cuál es el propósito del signo de interrogación en este código? `var rocks: Int? = 3`

- ☐ El tipo de variable `rocks` no es fijo.
- ☒ La variable `rocks` se puede establecer en nula.
- ☐ La variable `rocks` no se puede establecer en nula.
- ☐ La variable `rocks` no debe inicializarse de inmediato.

```
1. import java.util.*
2.
3. fun main() {
4.     // Declaración de La lista immutable
5.     val school = listOf("shark", "salmon", "minnow")
6.
7.     // Iterar sobre los elementos de la lista e imprimirlos
8.     for (fish in school) {
9.         println(fish)
10.    }
11. }
12.
```

Éxito #stdin #stdout 0.13s 46008KB

[comments \(?\)](#)

 stdin

[copy](#)

Standard input is empty

 stdout

[copy](#)

```
shark
salmon
minnow
```

[editar](#) [fork](#) [descargar](#)

[copy](#)

```
1. import java.util.*
2.
3. fun main() {
4.     for (i in 3..8 step 2) {
5.         print(i)
6.     }
7. }
8.
```

Éxito #stdin #stdout 0.12s 38284KB

[comments \(?\)](#)

 stdin

[copy](#)

Standard input is empty

 stdout

[copy](#)

357

4- Codelabs: Functions vale 4 puntos

<https://developer.android.com/codelabs/kotlin-bootcamp-functions>

Pregunta 1

La `contains(element: String)` función regresa `true` si la cadena `element` está contenida en la cadena a la que se llama. ¿Cuál será el resultado del siguiente código?

```
val decorations = listOf ("rock", "pagoda", "plastic plant", "alligator", "flowerpot")

println(decorations.filter {it.contains('p')}))
```

- ☐ [pagoda, plastic, plant]
- ☐ [pagoda, plastic plant]
- ☒ [pagoda, plastic plant, flowerpot]
- ☐ [rock, alligator]

Pregunta 2

En la siguiente definición de función, ¿cuál de los parámetros se requiere? `fun shouldChangeWater (day: String, temperature: Int = 22, dirty: Int = 20, numDecorations: Int = 0): Boolean {...}`

- ☐ numDecorations
- ☐ dirty
- ☒ day
- ☐ temperature

Pregunta 3

Puede pasar una función con nombre normal (no el resultado de llamarla) a otra función. ¿Cómo pasarías `increaseDirty(start: Int) = start + 1` a `updateDirty(dirty: Int, operation: (Int) -> Int)`?

- ☐ updateDirty(15, &increaseDirty())
- ☐ updateDirty(15, increaseDirty())
- ☐ updateDirty(15, ("increaseDirty()"))
- ☒ updateDirty(15, ::increaseDirty)

```
1. import java.util.*
2.
3. fun increaseDirty(start: Int): Int {
4.     return start + 1
5. }
6.
7. fun updateDirty(dirty: Int, operation: (Int) -> Int): Int {
8.     return operation(dirty)
9. }
10.
11. fun main() {
12.     val initialDirty = 10
13.     val newDirty = updateDirty(initialDirty, ::increaseDirty)
14.     println("El valor de dirty después de la actualización es: $newDirty")
15. }
16.
```

Éxito #stdin #stdout 0.11s 38272KB

[comments \(?\)](#)

stdin

[copy](#)

Standard input is empty

stdout

[copy](#)

El valor de dirty después de la actualización es: 11

5- Codelabs: Classes and objects vale 4 puntos

<https://developer.android.com/codelabs/kotlin-bootcamp-classes>

Pregunta 1

Las clases tienen un método especial que sirve como modelo para crear objetos de esa clase. ¿Cómo se llama el método?

- ☐ Un constructor
- ☐ Un instanciador
- ☒ Un constructor
- ☐ Un plano

Pregunta 2

¿Cuál de las siguientes afirmaciones sobre interfaces y clases abstractas NO es correcta?

- ☐ Las clases abstractas pueden tener constructores.
- ☐ Las interfaces no pueden tener constructores.
- ☒ Se pueden crear instancias de interfaces y clases abstractas directamente.
- ☐ Las propiedades abstractas deben ser implementadas por subclases de la clase abstracta.

Pregunta 3

¿Cuál de los siguientes NO es un modificador de visibilidad de Kotlin para propiedades, métodos, etc.?

- ☐ `internal`
- ☒ `nosubclass`
- ☐ `protected`
- ☐ `private`

```

1 internal open class InternalClass {
2     internal val internalProperty: Int = 5
3
4     internal fun internalFunction() {
5         println("This is an internal function.")
6     }
7 }
8
9 class Parent {
10     protected val protectedProperty: Int = 10
11 }
12
13 class Child : Parent() {
14     fun accessProtectedProperty() {
15         println("The value of protected property in Child class is: $protectedProperty")
16     }
17 }
18
19 class MyClass {
20     private val privateProperty: String = "Private Property"
21
22     private fun privateFunction() {
23         println("This is a private function.")
24     }
25
26     fun accessPrivateMembers() {
27         println(privateProperty)
28         privateFunction()
29     }
30 }
31
32 fun main() {
33     // Uso de elementos con diferentes modificadores de visibilidad
34     val internalObj = InternalClass()
35     internalObj.internalFunction()
36     println("The value of internal property is: ${internalObj.internalProperty}")
37
38     val child = Child()
39     child.accessProtectedProperty()
40
41     val myObj = MyClass()
42     myObj.accessPrivateMembers()
43 }

```

Pregunta 4

Considere esta clase de datos: `data class Fish(val name: String, val species:String, val colors:String)`. ¿Cuál de los siguientes NO es código válido para crear y desestructurar un `Fish` objeto?

- ☐ `val (name1, species1, colors1) = Fish("Pat", "Plecostomus", "gold")`
- ☐ `val (name2, _, colors2) = Fish("Bitey", "shark", "gray")`
- ☒ `val (name3, species3, _) = Fish("Amy", "angelfish", "blue and black stripes")`
- ☐ `val (name4, species4, colors4) = Fish("Harry", "halibut")`

Pregunta 5

Digamos que eres dueño de un zoológico con muchos animales que necesitan ser cuidados. ¿Cuál de las siguientes opciones NO sería parte de la implementación del cuidado?

- ☒ Una `interface` para diferentes tipos de alimentos que comen los animales.
- ☐ Una `abstract Caretaker` clase a partir de la cual puedes crear diferentes tipos de cuidadores.
- ☐ Un `interface` para dar agua limpia a un animal.
- ☐ Una `data` clase para una entrada en un horario de alimentación.

6- Codelabs: Extensions vale 4 puntos

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

Responde estas preguntas

Pregunta 1

¿Cuál de los siguientes devuelve una copia de una lista?

☐ `add()`

☐ `remove()`

☒ `reversed()`

☐ `contains()`

Pregunta 2

¿Cuál de estas funciones de extensión `class AquariumPlant(val color: String, val size: Int, private val cost: Double, val leafy: Boolean)` dará un error de compilación?

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

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

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

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

Pregunta 3

¿Cuál de los siguientes no es un lugar donde puedas definir constantes `const val`?

☐ en el nivel superior de un archivo

☒ en clases regulares

☐ en objetos singleton

☐ en objetos complementarios

```

3. fun main() {
4.     val originalList = mutableListOf(1, 2, 3, 4, 5)
5.
6.     // Agregar un elemento a La Lista
7.     originalList.add(6)
8.
9.     // Remover un elemento de La Lista
10.    originalList.remove(3)
11.
12.    // Verificar si La Lista contiene un elemento
13.    val containsElement = originalList.contains(4)
14.
15.    // Obtener una copia invertida de La Lista
16.    val reversedList = originalList.reversed()
17.
18.    // Imprimir resultados
19.    println("Lista original: $originalList") // Lista original después de agregar y eliminar: [1, 2, 4, 5,
20.    6]
21.    println("¿La lista original contiene el elemento 4? $containsElement") // Imprime: ¿La Lista original c
22.    ontiene el elemento 4? true
23.    println("Lista invertida: $reversedList") // Imprime: Lista invertida: [6, 5, 4, 2, 1]
24. }

```

Éxito #stdin #stdout 0.1s 38612KB

[comments \(?\)](#)

 stdin

[copy](#)

Standard input is empty

 stdout

[copy](#)

Lista original: [1, 2, 4, 5, 6]
 ¿La lista original contiene el elemento 4? true
 Lista invertida: [6, 5, 4, 2, 1]