



Universidad Nacional Autónoma de México
Facultad de Estudios Superiores Acatlán

Laboratorio de Desarrollo de Aplicaciones Móviles para iOS

Práctica final del curso Online Introducción a Swift



Instrucciones práctica final

En una habitación de hospital, se tienen los siguientes pacientes con su respectiva información.

	Nombre	Apellido	Edad	Dado de Alta
Paciente 1	Juan Carlos	Loeza Peña	31	No
Paciente 2	Karla Ximena	Pérez Martinez	23	Si
Paciente 3	Raúl Santiago	Juárez Nuñez	26	Si
Paciente 4	Miguel Ángel	Álvarez Grapain	28	No

- 1.- Define una estructura para almacenar la información anterior para cada paciente.
 - 2.- Crea 4 variables de tipo estructura anterior para almacenar los 4 pacientes de la tabla anterior.
 - 3.- Declara un arreglo de estructura para almacenar los 4 pacientes anteriores.
 - 4.- Crea 4 funciones diferentes que reciba como parametro un arreglo de struct que definiste en la instrucción 1° que realicen los siguientes procesos:
 - a) Función que imprima nombre completo de los pacientes.
 - b) Función que calcule e imprima la media de las edades de los pacientes.
 - c) Función que imprima:
“El paciente está dado de alta” ó “El paciente no está dado de alta”
Según la información de cada paciente.
 - d) Función que reciba 2 parámetros: un arreglo de struct definida en el ejercicio 1° y un número entero. Deberá imprimir la información completa del paciente que se encuentre en la casilla o posición del número entero que la función ha recibido.
- Advertencia:** Dado el número de pacientes, solo podrá recibir los números: 0,1,2,3 que corresponden a las posiciones de los 4 pacientes.

Por último, deberás llamar a cada función con los parámetros y mostrar el resultado en consola. Para la función del inciso 4.d) deberás llamar a la función 4 veces por cada paciente.

Sugerencias para elaborar la práctica final

Las indicaciones anteriores están relacionadas con los temas anteriores de las sesiones, por lo que puedes consultar el material para guiarte y elaborar la práctica.

Por otro lado, brindamos el siguiente ejemplo para declarar las funciones que reciben como parámetro una estructura:

```
1 import Foundation
2
3 struct perro{                //Declaramos una estructura llamada Perro
4     var nombre: String       //Variable nombre de tipo String
5     var raza: String         //Variable raza de tipo String
6     var edad: Int            //Variable edad de tipo Int
7     var esMacho: Bool        //Variable Bool para saber si es macho
8 }
9 // Instanciamos las variables de los perros de tipo perro.
10 var perro1 = perro(nombre: "Keysi", raza: "Cruzada", edad: 4, esMacho: false)
11 var perro2 = perro(nombre: "Peggy", raza: "Cruzada", edad: 9, esMacho: false)
12 var perro3 = perro(nombre: "Esko", raza: "Siberian Husky", edad: 2, esMacho: true)
13
14 var misPerros = [perro]()    //Inicializamos un arreglo vacío de estructuras perro
15
16 misPerros.append(perro1)     //Añadimos perro1 al arreglo, se coloca en la posición 0
17 misPerros.append(perro2)     //Añadimos perro2 al arreglo, se coloca en la posición 1
18 misPerros.append(perro3)     //Añadimos perro3 al arreglo, se coloca en la posición 2
19
20 //Declaramos una función que recibe un parámetro de arreglo perro llamado "parametroPerro"
21 func imprimeNombres(parametroPerro: [perro]){
22     for i in 0...parametroPerro.count-1{           //Hacemos el ciclo for para recorrer los nombres
23         print("El nombre del \(i+1) perro es: \(parametroPerro[i].nombre)")
24     }
25 }
26 //Llamamos a la función enviándole como parámetro el arreglo o lista misPerros
27 imprimeNombres(parametroPerro: misPerros)
```



```
El nombre del 1 perro es: Keysi
El nombre del 2 perro es: Peggy
El nombre del 3 perro es: Esko
```

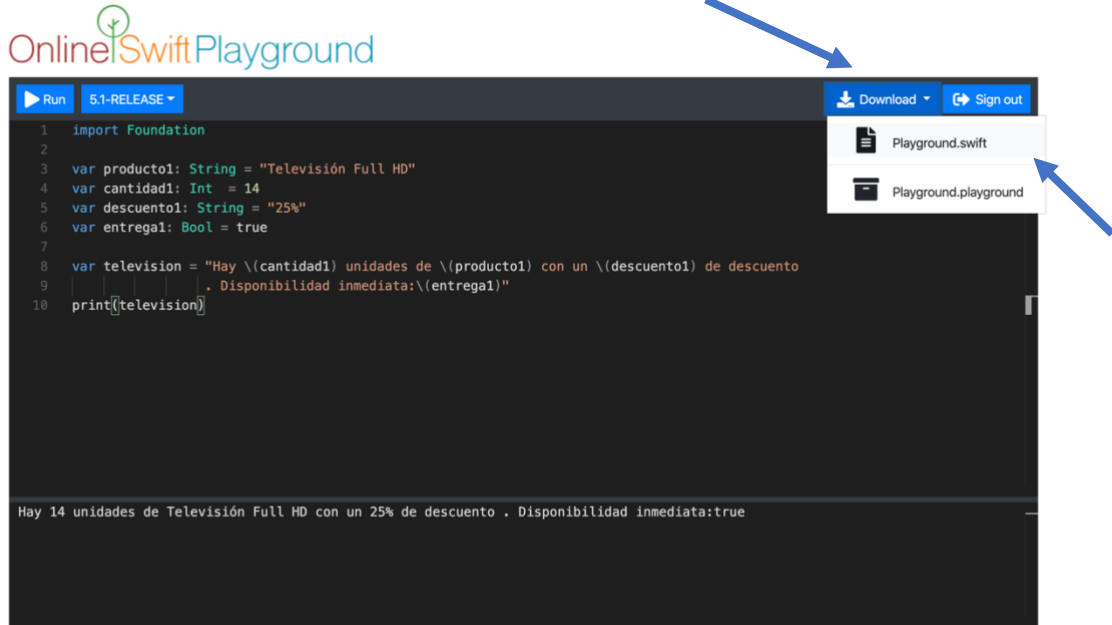
Sintaxis para funciones con 2 o más parámetros:

```
func nombreDeLaFunción( parametro1: TipoDeDato, parametro2: TipoDeDato){
    sentencias
    .
    .
}
```

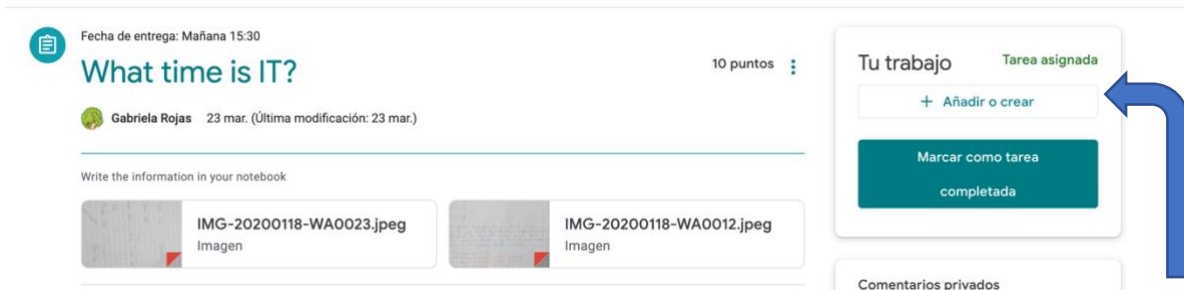
¡Mucho éxito!

Instrucciones para entregar la tarea:

Puedes hacer entrega de tu tarea de actividad en google Classroom, presiona el botón “Download” sobre el compilador. Posteriormente, presiona en descargar el archivo Playground.swift



Después, dirígete al apartado en Google classroom de la lista de actividades y sube tu archivo en “Añadir o crear”



Por último, elige tu archivo .swift y sube tu tarea.



Nota: No olvides confirmar tu entrega al finalizar la selección del archivo.

Si tienes problemas o algún inconveniente con esta forma, también puedes subir la tarea con capturas de pantalla de tu código con los resultados en consola.