



Swift





Introducción

Bienvenidos al módulo de Swift. Durante su desarrollo, nos adentraremos a Swift y a sus principales funcionalidades para, luego, trabajar con iOS. Para ello, buscaremos hacer foco en conceptos claves como “constante”, “variable”, “tipos de datos”, “operadores”, “funciones”, entre otros. Buscamos que con este recorrido puedas adquirir una noción general sobre todo lo que se puede realizar con esta herramienta y que comiences a implementarlo en tu día a día.



Tema 1. Introducción a Xcode, comentarios y print

Objetivo

Luego de navegar esta cápsula, deberás podés hacer lo siguiente:

- conocer el origen y uso de Swift;
- descargar y abrir Xcode;
- crear un nuevo Playground de Xcode;
- interactuar e identificar las secciones del Playground;
- escribir comentarios; e
- imprimir valores en la consola usando `print`.

Swift y Xcode

Swift es un lenguaje de programación multiparadigma creado por Apple para el desarrollo de aplicaciones para iOS, Mac, Apple TV y Apple Watch.

Inicialmente, el desarrollo de aplicaciones para este entorno se realizaba con Objective-C, pero desde 2014, con la inclusión de Swift, se ha empezado un proceso de migración de Objective-C a Swift, lo que ha hecho que incluso hoy en día haya aplicaciones en las que convivan ambos lenguajes.

Para este tipo de aplicaciones, se utiliza Xcode como IDE, que es un entorno de desarrollo integrado; esto esencialmente significa que Xcode incluye muchas herramientas adicionales para el desarrollo, como depurador, control de código fuente, administrador de dispositivos, simulador, y herramientas de creación de perfiles,

Interface Builder, documentación y más. Está pensado principalmente para compilar Swift y Objective-C, pero soporta otros lenguajes como C / C++, AppleScript, Java y más.

Xcode solo se ejecuta en macOS: es necesario tener una Mac para crear aplicaciones de iOS con Xcode (no se ejecutará en iPad). Hay algunas alternativas disponibles para Windows/PC, pero están lejos de ser ideales.

Requisitos mínimos del sistema para Xcode 12:

- una Mac con macOS Catalina o macOS Big Sur;
- al menos 50 GB de espacio de almacenamiento libre;
- 8 GB de RAM pero 16 GB para más comodidad; y
- al menos una CPU equivalente a Intel i5 o i7.

Descarga del Xcode

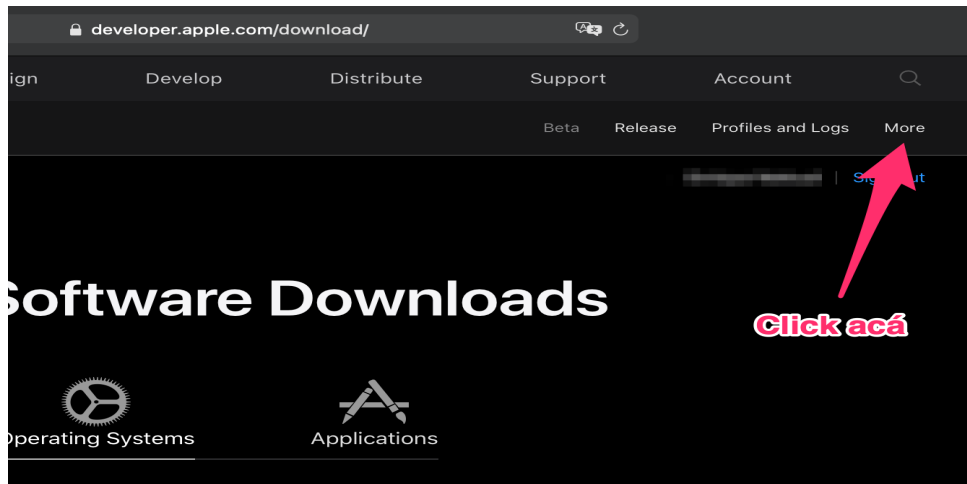
La forma más sencilla para descargar Xcode es a través del [App Store](#).

Para este curso, usaremos la versión 12.4, que deberás descargar desde la [página de Apple](#):

Ten paciencia, pues la instalación puede demorar si la conexión no es buena.

1. Obtén una cuenta gratuita de desarrollador con tu ID de Apple. En este *post* se indican los pasos para crear la cuenta.
2. Ingresa a la página de Apple e inicia sesión.
3. Haz clic en «More» en la parte superior derecha de la página

Figura 1: Descargar Xcode



Fuente: adaptación propia con base en captura de pantalla de web de Apple (developer.apple.com/download/)

4. Utiliza el campo de búsqueda para encontrar la versión 12.4 de Xcode.

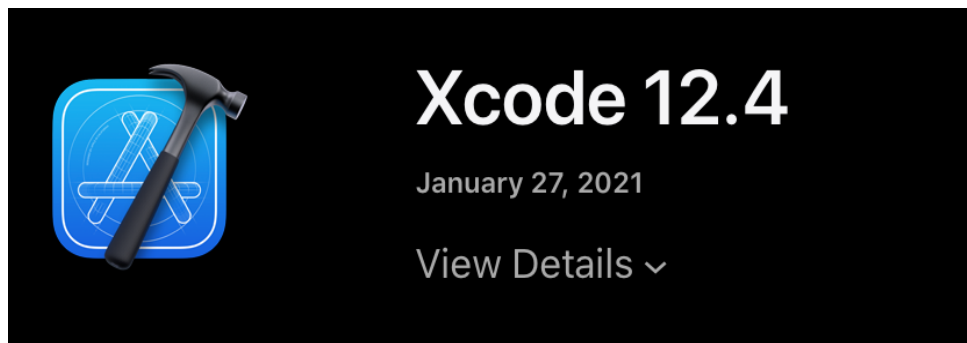
Figura 2: Encontrar versión 12.4 de Xcode



Fuente: captura de pantalla de web de Apple (developer.apple.com/download/)

5. Selecciona el .dmg para iniciar la descarga

Figura 3: Selección del .dmg para iniciar la descarga



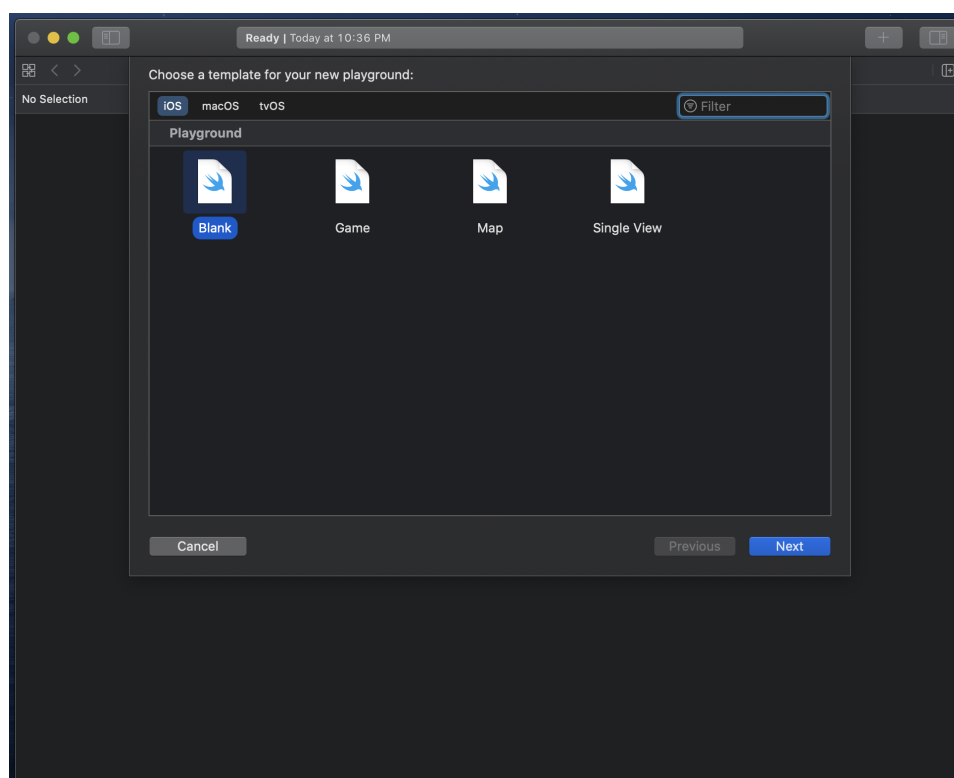
Fuente: captura de pantalla de web de Apple (developer.apple.com/download/)

Playgrounds

Xcode permite crear diferentes tipos de proyectos según la necesidad de cada equipo; entre estos, se incluyen los Playgrounds. Los Playgrounds permiten escribir y probar código de manera rápida sin necesidad de crear una aplicación completa.

Para crear un Playground, debes abrir Xcode y, en la barra de herramientas, seleccionar «File» > «New» > «Playground». Se mostrará una pantalla para elegir el *template* del Playground. Selecciona «Blank» y luego elige el nombre y ubicación para guardar el archivo.

Figura 4: Crear un Playground



Fuente: captura de pantalla de Xcode (Apple, Inc.)

Al finalizar la creación, se mostrará el Playground con un código de ejemplo.

Figura 5: Playground con un código de ejemplo



Fuente: adaptación propia con base en captura de pantalla de Xcode (Apple, Inc.)

1. Editor de código: en esta zona es donde se escribe el código.
2. Zona de resultados: aquí se muestran los resultados del código. Sirve para ir revisando que el código funcione como se espera.
3. Botón de ejecución: los Playgrounds pueden ejecutarse ya sea automática o manualmente. Para cambiar entre estas opciones, debes dejar presionado el botón «Play» de la parte inferior y seleccionar la opción deseada. En caso de que esto se haga manualmente, deberás presionar cualquiera de los dos botones de «Play» para que el código se ejecute.
4. Visor de actividades: aquí se muestra el estado del Playground y se indica si está en proceso de ejecución (mostrando un *spinner*), si se terminó la ejecución correctamente o si hay algún error.
5. Área *debug*: en esta zona se mostrarán los mensajes que se imprimen usando `print()`.

Comentarios en Swift

Un comentario es un texto aclaratorio que el compilador ignora completamente y está destinado a ser leído por los programadores. Los comentarios en Swift no son muy distintos de otros lenguajes. Su sintaxis es bastante clásica:

Los comentarios de una sola línea empiezan con “//”.

```
// Esto es un comentario
```

Los comentarios de múltiples líneas empiezan con “/*” y terminan con “*/”.

```
/*  
    Esto es un comentario  
    de varias líneas  
*/
```

Print

La función `Print` es muy similar a equivalentes de otros lenguajes como C, Java, C#.

Es muy usada para *debugging* durante el desarrollo de un producto en iOS/macOS; permite mostrar información durante las pruebas, pero no deben dejarse en el código final.

En su forma más simple, por ejemplo, se mostrará en consola el texto “Hola Mundo!”:

```
print("Hola Mundo!")
```

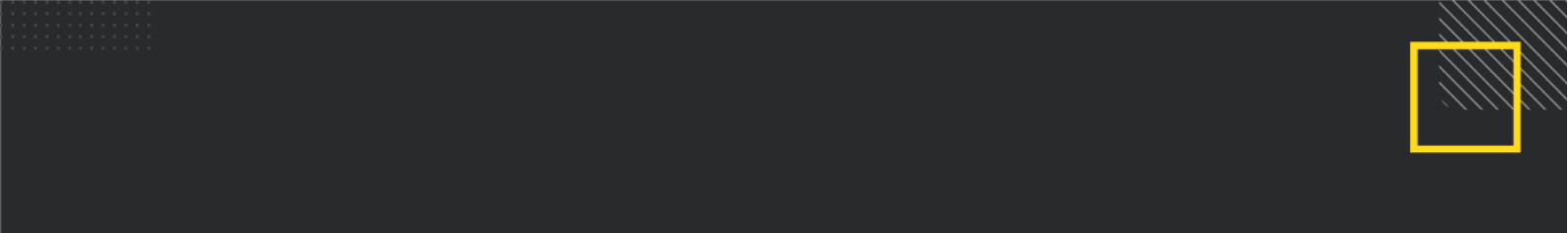
Si bien “Hola Mundo” es un *string*, podemos usar `print` para otros tipos de datos como `Int` y `Float`.

```
print(1.52) // 1.54  
print(150) // 150  
print(true) // true
```

Veremos los tipos de datos en las cápsulas “Variables” y “Tipos de datos I”.

Actividad de repaso

1. Crea un Playground siguiendo los pasos indicados anteriormente.
2. Escribe un comentario de una línea que diga “Mi primer Playground”.
3. Escribe un comentario de varias líneas que diga tu nombre, edad, y la ciudad donde vives.



4. Utilizando `print`, muestra en consola un mensaje con tu nombre y luego otro con tu edad.