



Actividad | 1 | Instalación de XCode/Aplicación 1.

Redacción avanzada.

Ingeniería en Desarrollo de
Software.



academiaglobal

TUTOR: Sandra Luz Lara Dévora

ALUMNO: Carlos Ariel Nicolini

FECHA: 21/09/2025

Índice

Introducción	3
Descripción	4
Justificación	5
Desarrollo.....	6
• Codificación	6
• Prueba de la aplicación.....	8
Conclusión.....	12
Referencias.....	13

Introducción

El lenguaje de programación Swift es el utilizado actualmente para los desarrollos digitales de apps para entornos IOS y macOS. Es decir, que es el lenguaje ideal a la hora de programar una app que deba ser compatible con los dispositivos de la marca Apple.

La presentación de Swift tenía su explicación en la búsqueda de un lenguaje de programación que fuera mas asequible que las opciones existentes en aquel momento, con Objective-C a la cabeza, además de ofrecer mejoras en cualquier desarrollo.

El éxito fue inmediato. Las características de Swift han motivado a que se haya estandarizado como lenguaje de programación propio de IOS. En la actualidad, todo profesional que se precie, debe dominarlo, para poder trabajar en el desarrollo de cualquier app compatible.

Las características de Swift son las que han respaldado su éxito. De forma general, este lenguaje de programación destaca en comparación a otros desarrollos anteriores, por ser más seguro, más rápido en su desarrollo y también que permite mejores resultados.

Los puntos fuertes de este lenguaje son:

- Sin errores: Al basarse en un código más limpio, con una estructura de variables menos propensa a incorrecciones y con gestiones automáticas, la existencia de errores o problemas es menor.
- Velocidad de desarrollo: Su existencia deriva de la necesidad de mejorar o incluso sustituir lenguajes como Objective-C (al cual no solo supera), sino que también se le considera mejor que usar C o C++.
- Constante evolución: Se presenta como una misma evolución del resto de lenguajes de programación previos a él, sigue evolucionando, aprovechando así los mejores avances en las nuevas tecnologías y permitiendo cada vez desarrollos mas complejos y funcionales para los usuarios.

Descripción

Contextualización:

Swift es un lenguaje de programación moderno que brinda mayor seguridad y es intuitivo. Por lo anterior, con este se pueden crear múltiples y variadas aplicaciones. Para lograrlo, es necesario contar con un amplio conocimiento de su funcionamiento para poder comprender como se deben diseñar y realizar dichas aplicaciones.

Se necesita crear una aplicación donde sea posible el ingresar un numero y que diga si es número par o impar.

Actividad:

Realizar instalación de XCode o utilizar los compiladores online (según la preferencia) y crear la aplicación solicitada.

En este ejercicio además de lo solicitado, me tome la libertad de agregarle otras pequeñas funcionalidades. Le agregue un bucle para que al realizar una consulta y tener el resultado, nos solicite nuevamente ingresar otro número. Esto continuara hasta que ingresemos la palabra salir, tras lo cual después de un mensaje de despedida el programa terminara.

Además, agregue una validación, para que si agregamos un carácter que no sea un numero (el carácter ingresado si no puede ser convertido en numérico) indicara que es incorrecto y que debemos ingresar un número.

Justificación

En esta ocasión nos empezaremos a familiarizar con el lenguaje de programación Swift el cual está basado en varios lenguajes preexistentes y realizaremos una aplicación que al ingresarle un numero nos indicara si es par o impar.

Este tipo de ejercicio ya lo hemos realizado con otros lenguajes de programación como C++, pero en este caso utilizaremos como ya indicamos anteriormente el lenguaje Swift y lo haremos online en la pagina Replit. Hemos revisado en clases este lenguaje el cual es muy intuitivo de usar, autocompleta muchos de los comandos que se ingresan para así evitar errores de escritura. Este ejercicio nos sirve para empezar a utilizarlo y ayudarnos a conocerlo, ya que es un lenguaje de programación con mucho potencial, con mucho futuro, muy fácil de usar, pero si que necesita tener mucha práctica con él, me he tenido que documentar mucho, leer y mirar videos de explicaciones para entender su funcionamiento, aunque sus principios son iguales o parecidos a los de otros lenguajes de programación, con sus cambios.

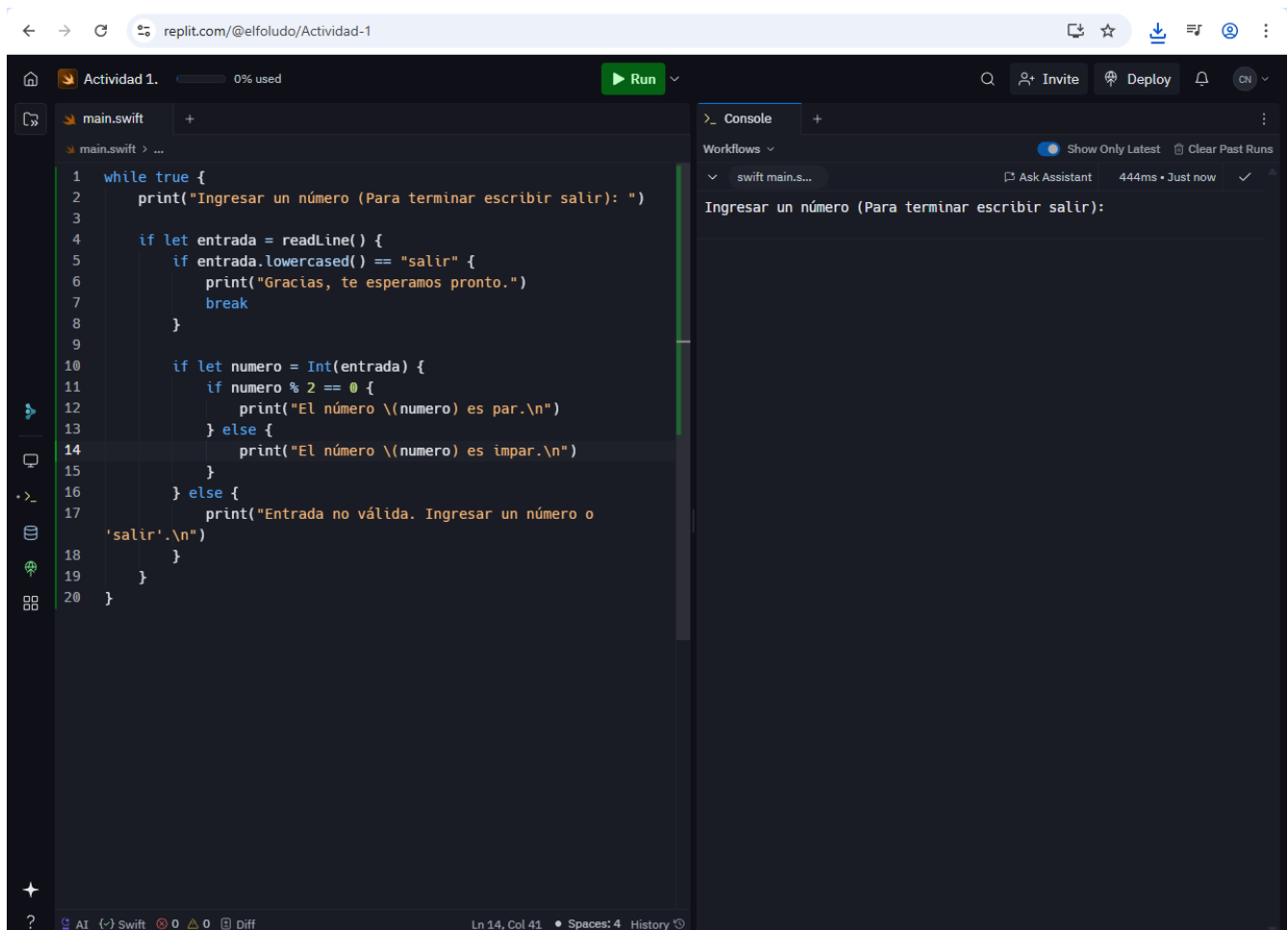
Este trabajo fue subido al siguiente enlace de GitHub

<https://github.com/CarlosNico/Desarrollo-de-Aplicaciones-M-viles-III>

Desarrollo

Codificación

En esta parte del ejercicio realizamos la codificación de la aplicación que al ingresar un numero nos indique en pantalla si es un numero par o impar. En mi caso agregue un bucle para que al consultar un número y revisar si es par o impar, se pueda realizar otra consulta, además de que se ingresa un carácter que no sea un numero nos dirá que es incorrecto, que se agregue un numero para validar, también en caso de querer salir de la aplicación solo tendremos que poner la palabra salir y se terminara el programa.



```
1 while true {
2     print("Ingresar un número (Para terminar escribir salir): ")
3
4     if let entrada = readLine() {
5         if entrada.lowercased() == "salir" {
6             print("Gracias, te esperamos pronto.")
7             break
8         }
9
10        if let numero = Int(entrada) {
11            if numero % 2 == 0 {
12                print("El número \(numero) es par.\n")
13            } else {
14                print("El número \(numero) es impar.\n")
15            }
16        } else {
17            print("Entrada no válida. Ingresar un número o
18            'salir'.\n")
19        }
20    }
```

The screenshot shows a Swift Playground interface. The left pane displays the Swift code for a program that repeatedly prompts the user to enter a number. If the user enters 'salir', the program prints a goodbye message and exits. If the user enters a valid integer, the program checks if it is even or odd and prints the result. If the input is not a valid integer, it prints an error message. The right pane shows the console output, which currently displays the prompt 'Ingresar un número (Para terminar escribir salir):'. The top bar of the playground shows 'Actividad 1.' and '0% used'.

A continuación, explicaremos lo que realiza el código:

- `While true {` - Declaramos el bucle infinito, lo que significa que el código dentro de las llaves se va a ejecutar una y otra vez hasta que sea detenido de manera manual.
- `Print(" Por favor ingresar un numero (o escribe "Salir" para terminar):")` – Envía el mensaje en la consola para pedirle al usuario que escriba un numero o la palabra salir si quiere terminar el programa.
- `If let entrada = readLine () }` –En esta línea intenta leer lo que el usuario escribió en la consola., verifica si que esa entrada no sea null, y si no lo es, la guarda en la variable entrada.
- `If entrada.lowercased() == "salir" {` - Convierte la entrada del usuario a minúsculas y la compara con Salir. Si el usuario escribió salir, entra en este bloque del código.
- `Print("Programa finalizado.")` y `break` – Imprime el mensaje de salida y sale del bucle con el break, lo cual termina la ejecución del programa.
- `If let numero = Int(entrada) {` - Si el usuario no escribió salir, intenta convertir la entrada en un numero entero, si la conversiones es exitosa, guarda el valor en la variable numero, si la conversión falla (por que no se escribió un numero), se va al bloque else.
- `If numero %2 == 0 {` - Comprueba si el numero es divisible por 2 (si lo es, entonces es número par).
- `Print("El número \n(numero) es par. \n")` – Imprime que el numero es par (incluyendo el valor de la variable numero).
- `else { print("El número \n(numero) es impar. \n") }` – Si el numero no es divisible por 2, entonces es impar. Imprime en la consola que el número es impar (incluyendo el valor de la variable numero).

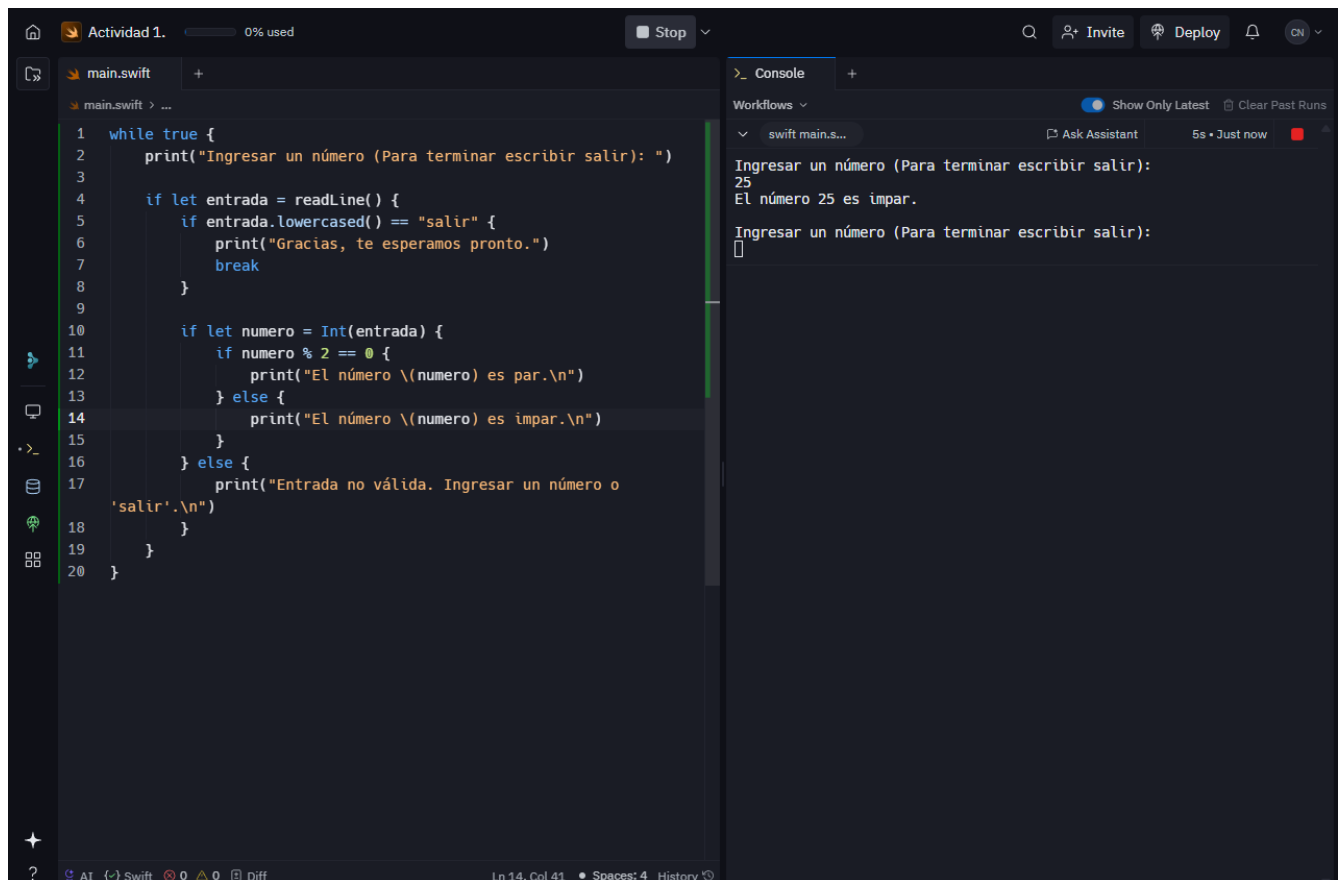
- Else { print(“Entrada no valida .Ingresar un numero o salir’. \n”) – Si la conversión a numero no fue exitosa, este bloque muestra un mensaje de error indicando que se ingrese un numero o la palabra salir para terminar el programa.

Después de esto, el ciclo while true vuelve a empezar, pidiendo al usuario ingresar un número o la palabra salir.

Prueba de la aplicación

A continuación, realizaremos las pruebas del aplicativo

Al ingresar el numero 25 nos da de resultado que es impar.



The screenshot shows a Swift code editor with a file named `main.swift`. The code implements a `while true` loop that prompts the user to enter a number. It checks if the input is "salir" (to exit) or if it's a valid integer. If it's an even number, it prints "El número \n(numero) es par.\n". If it's an odd number, it prints "El número \n(numero) es impar.\n". If the input is not a valid number, it prints "Entrada no válida. Ingresar un número o 'salir'. \n".

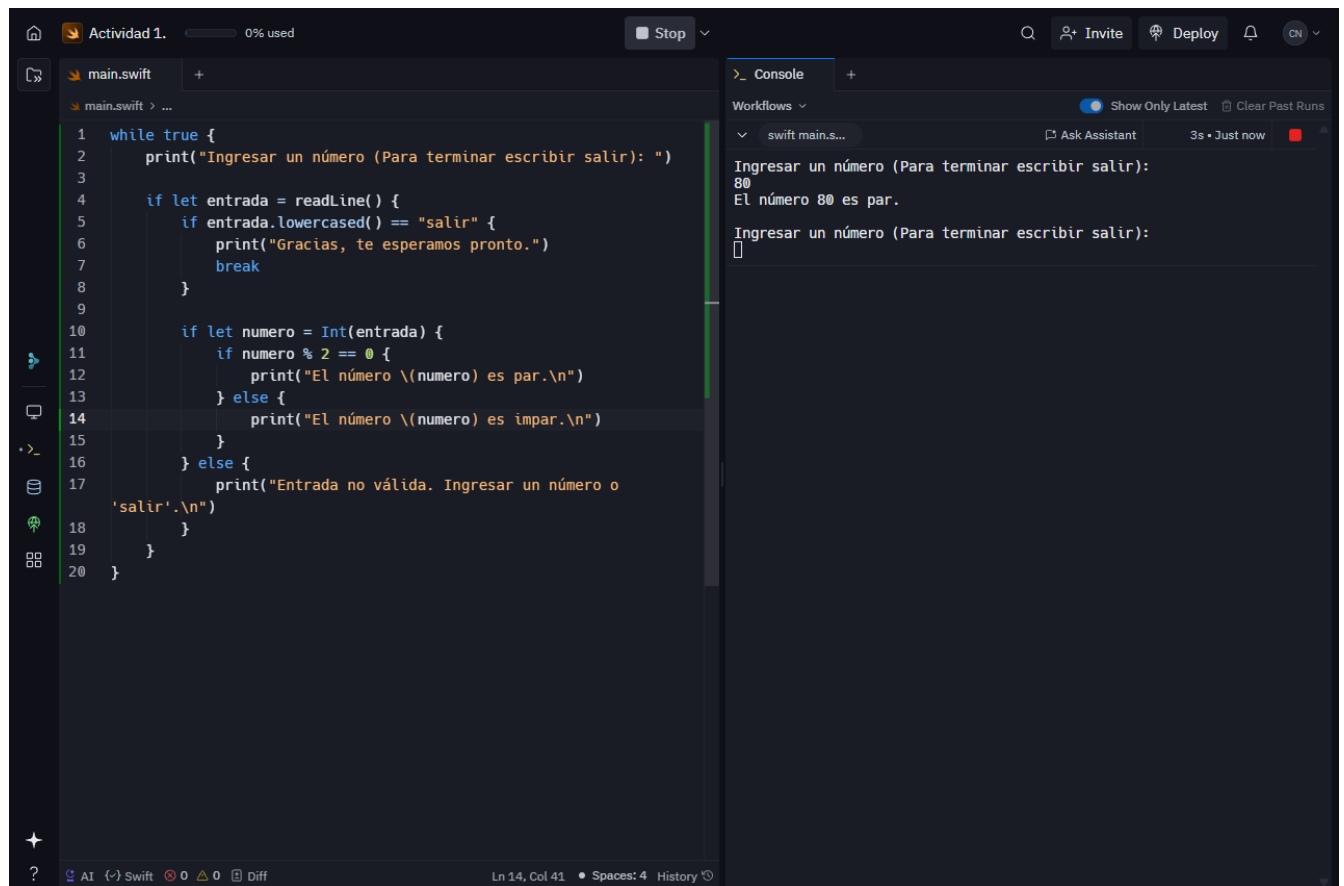
```
1 while true {
2     print("Ingresar un número (Para terminar escribir salir): ")
3
4     if let entrada = readLine() {
5         if entrada.lowercased() == "salir" {
6             print("Gracias, te esperamos pronto.")
7             break
8         }
9
10        if let numero = Int(entrada) {
11            if numero % 2 == 0 {
12                print("El número \n(numero) es par.\n")
13            } else {
14                print("El número \n(numero) es impar.\n")
15            }
16        } else {
17            print("Entrada no válida. Ingresar un número o
18            'salir'. \n")
19        }
20    }
}
```

The console output shows the program running. It prompts for input, receives "25", and outputs "El número 25 es impar." followed by another prompt.

```
> Console
Workflows
  swift main.s... 5s • Just now
Ingresar un número (Para terminar escribir salir):
25
El número 25 es impar.
Ingresar un número (Para terminar escribir salir):

```


Al ingresar el numero 80 nos indica que es par

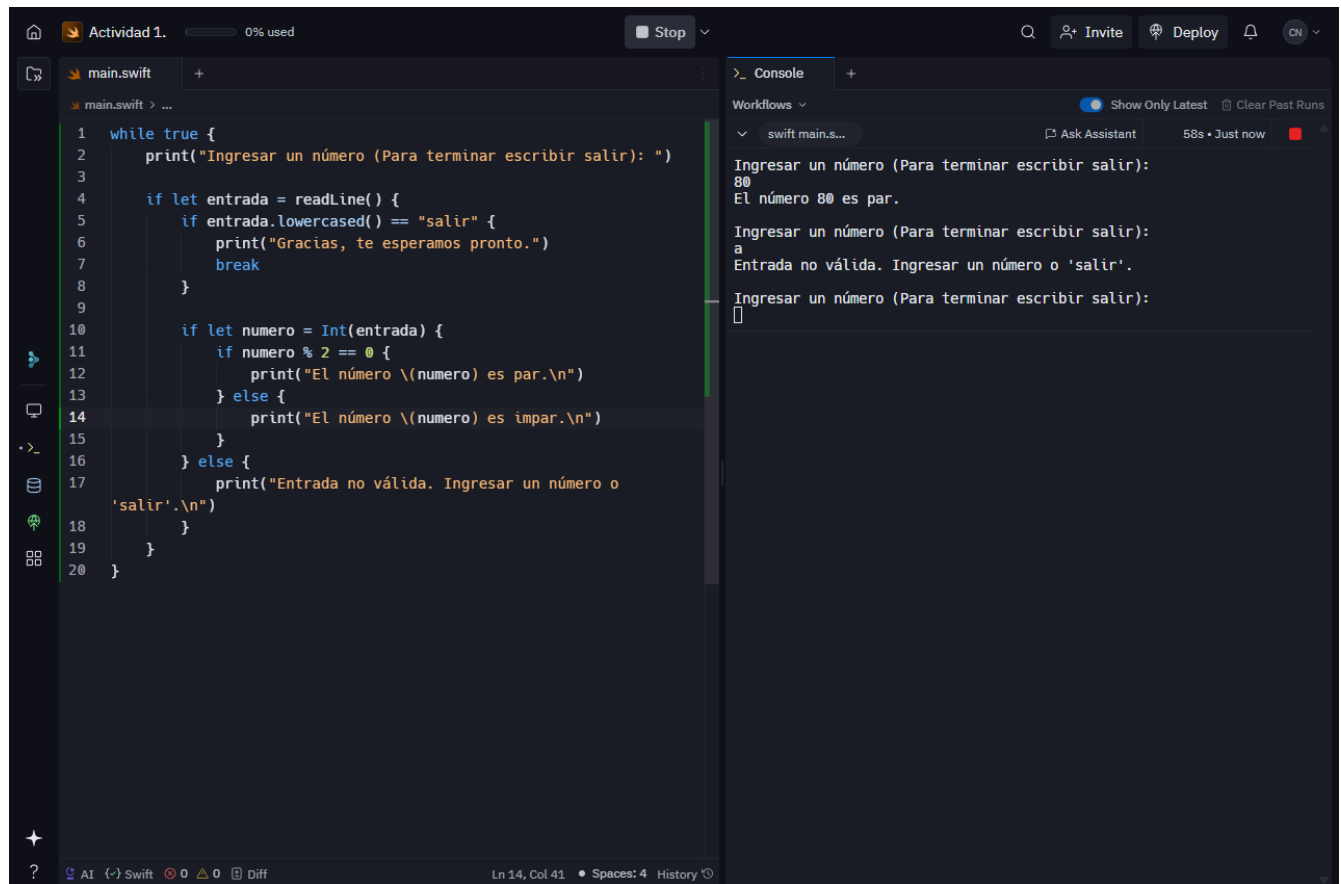


```
1 while true {
2     print("Ingresar un número (Para terminar escribir salir): ")
3
4     if let entrada = readLine() {
5         if entrada.lowercased() == "salir" {
6             print("Gracias, te esperamos pronto.")
7             break
8         }
9
10        if let numero = Int(entrada) {
11            if numero % 2 == 0 {
12                print("El número \(numero) es par.\n")
13            } else {
14                print("El número \(numero) es impar.\n")
15            }
16        } else {
17            print("Entrada no válida. Ingresar un número o
18            'salir'.\n")
19        }
20    }
```

Console Output:

```
Ingresar un número (Para terminar escribir salir):
80
El número 80 es par.
Ingresar un número (Para terminar escribir salir):
```

Al ingresar una letra nos indica que es una entrada no valida (además se muestra que el bucle funciona y sigue pidiendo ingresar un numero hasta que pongamos la palabra salir y se cierre).



The screenshot shows a Swift code editor with a file named `main.swift`. The code implements a `while true` loop that prompts the user to enter a number. If the user enters the word "salir", the program prints a thank-you message and breaks the loop. If the user enters a number, it checks if it is even or odd. If the user enters a letter, it prints an error message and continues the loop.

```
1 while true {
2     print("Ingresar un número (Para terminar escribir salir): ")
3
4     if let entrada = readLine() {
5         if entrada.lowercased() == "salir" {
6             print("Gracias, te esperamos pronto.")
7             break
8         }
9
10        if let numero = Int(entrada) {
11            if numero % 2 == 0 {
12                print("El número \(numero) es par.\n")
13            } else {
14                print("El número \(numero) es impar.\n")
15            }
16        } else {
17            print("Entrada no válida. Ingresar un número o
18            'salir'.\n")
19        }
20    }
```

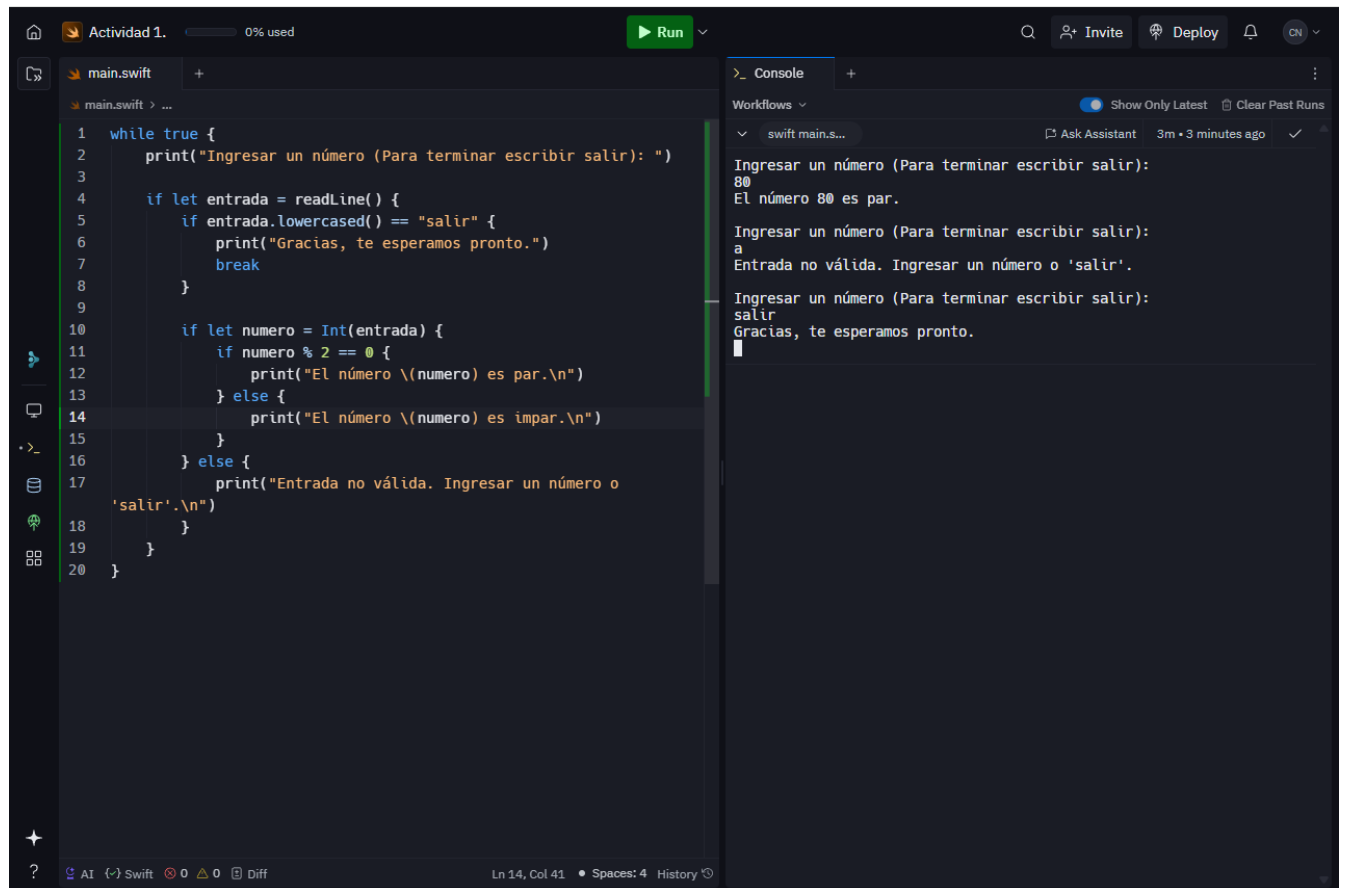
The console output shows the program's execution:

```
Ingresar un número (Para terminar escribir salir):
80
El número 80 es par.

Ingresar un número (Para terminar escribir salir):
a
Entrada no válida. Ingresar un número o 'salir'.

Ingresar un número (Para terminar escribir salir):
```

Al ingresar la palabra salir nos da la despedida y termina.



```
1 while true {
2     print("Ingresar un número (Para terminar escribir salir): ")
3
4     if let entrada = readLine() {
5         if entrada.lowercased() == "salir" {
6             print("Gracias, te esperamos pronto.")
7             break
8         }
9
10        if let numero = Int(entrada) {
11            if numero % 2 == 0 {
12                print("El número \(numero) es par.\n")
13            } else {
14                print("El número \(numero) es impar.\n")
15            }
16        } else {
17            print("Entrada no válida. Ingresar un número o
18            'salir'. \n")
19        }
20    }
```

Console Output:

```
Ingresar un número (Para terminar escribir salir):
80
El número 80 es par.

Ingresar un número (Para terminar escribir salir):
a
Entrada no válida. Ingresar un número o 'salir'.

Ingresar un número (Para terminar escribir salir):
salir
Gracias, te esperamos pronto.
```

Conclusión

El aprendizaje de los distintos lenguajes de programación, nuevos y viejos es indispensable para todo programador o persona que quiera ingresar en el mundo de la programación, lo cual es una herramienta muy necesaria en el ámbito laboral. El lenguaje de programación Swift es un lenguaje moderno, muy potente y fácil de aprender el cual nos va a ser de mucha utilidad para las plataformas IOS, watchOS, macOS y tvOS. Este programa tiene una sintaxis muy sencilla y clara, con una ayuda para que sea más segura lo cual nos va a ayudar muchísimo en aprender más rápido.

En este ejercicio no se si es porque ya hemos hecho muchos trabajos con varios lenguajes o porque este lenguaje es de muy rápido aprendizaje (puede que sean los dos combinados), que me divertí muchísimo y fue una experiencia muy agradable.

Espero que el trabajo realizado cumpla con lo solicitado y espero poder seguir aprendiendo sobre este lenguaje el cual me va a ayudar muchísimo en mi crecimiento como profesional y que me sirve de experiencia para los próximos retos que se me van a presentar tanto en la universidad como en mi vida laboral.

<https://github.com/CarlosNico/Desarrollo-de-Aplicaciones-M-viles-III>

Referencias

¿Qué es Swift? El gran lenguaje de programación. (n.d.). Tokio School. Retrieved August 9, 2025, from <https://www.tokioschool.com/formaciones/cursos-programacion/swift/que-es/>

Swift - Apple (MX). (n.d.). Apple. Retrieved August 9, 2025, from <https://www.apple.com/mx/swift/>

Programación en Swift: idescubre sus ventajas y desventajas! (n.d.). Tokio School. Retrieved August 9, 2025, from <https://www.tokioschool.com/noticias/swift-ventajas-desventajas/>