

Ingeniería en Desarrollo de Software

Actividad: Número 3.

Nombre de la Actividad: Programa 2 (parte 2).

Banco mexicano / Saldo y salir.

Nombre del Curso: Desarrollo de aplicaciones móviles IV.

Tutor: Marco Alonso Rodríguez Tapia.

Alumno: Omar Juárez Carmona.

Fecha: 27 - Noviembre – 2023.

INDICE

Contextualización y actividad.....	3
Introducción.....	4
Descripción.....	6
Justificación.....	7
Investigación.....	9
Desarrollo.....	10
- Codificación.....	10
- Prueba del programa.....	15
Conclusión.....	18
Referencias y link.....	19

CONTEXTUALIZACION Y ACTIVIDAD

Contextualización:

El Banco Mexicano necesita un programa para su banca en línea. En este se podrán realizar diversas acciones, como depositar, retirar, consultar saldo o salir. La app deberá estar creada con el lenguaje Swift.

Actividad:

En la actividad anterior, se creó la interfaz del menú principal del programa de Banco Mexicano. Además, se crearon las opciones “Depósito” y “Retiro”. Ahora, se deberá terminar de darle funcionalidad a las opciones faltantes: “Saldo” y “Salir”. En caso de que el usuario ingrese la opción Saldo, el programa deberá mostrar en pantalla:

- El saldo del usuario
- Preguntar si desea realizar otra operación.

En caso de que el usuario ingrese la opción Salir, el programa deberá mostrar en pantalla que se ha cerrado la sesión.

INTRODUCCION

Swift es un lenguaje de programación poderoso e intuitivo creado por Apple para desarrollar aplicaciones de iOS, Mac, Apple TV y Apple Watch. Es fácil de usar y de código abierto, lo que lo hace ideal para que cualquier persona con una idea pueda hacerla realidad. Los desarrolladores pueden programar de una forma más segura y confiable, ahorrar tiempo y enriquecer la experiencia con las aplicaciones. Swift es rápido y eficiente, proporciona información en tiempo real y puede incorporarse fácilmente al código de Objective-C existente. Los desarrolladores están haciendo cosas asombrosas con Swift, como crear aplicaciones para Lyft, Khan Academy, Clear Sky Guide, LinkedIn, Kickstarter, Eventbrite y Hipmunk. Además, las aplicaciones en Swift son pura velocidad. Por ejemplo, un algoritmo común de búsqueda se completa mucho más rápido en Swift que en Objective-C o Python.

2.7. Numerosas universidades e instituciones académicas de todo el mundo están enseñando a programar con Swift y Xcode en el Mac, lo que permite a los estudiantes contar con las mejores herramientas para crear aplicaciones increíbles. ¡Y lo mejor de todo es que Swift es gratis y de código abierto!

Swift es un lenguaje de programación de alto nivel desarrollado por Apple. Es un lenguaje de programación multiparadigma que admite programación orientada a objetos, programación imperativa y programación funcional.

Para programar funciones avanzadas en Swift, primero debes tener una comprensión sólida de las funciones básicas. Las funciones son bloques de código que realizan una tarea específica y se pueden llamar en cualquier lugar de tu programa. En Swift, las funciones son ciudadanos de primera clase, lo que significa que se pueden pasar como argumentos a otras funciones y se pueden devolver como valores.

Una vez que tengas una comprensión sólida de las funciones básicas, puedes comenzar a explorar las funciones avanzadas en Swift. Algunas de las características avanzadas de las funciones en Swift incluyen:

- ✓ Funciones anidadas
- ✓ Funciones genéricas
- ✓ Funciones con parámetros variádicos
- ✓ Funciones con valores predeterminados
- ✓ Funciones con parámetros de entrada y salida

Replit es una plataforma en línea que actúa como un entorno de desarrollo integrado (IDE) basado en la nube. Es compatible con múltiples lenguajes de programación, incluidos JavaScript, Python y HTML. Replit es accesible desde cualquier lugar y no requiere instalación. Además, tiene una comunidad activa y recursos educativos.

Sin más preámbulo, continuemos con nuestra actividad final conociendo la programación con funciones avanzadas en el lenguaje de programación Swift.

DESCRIPCION

El Banco Mexicano necesita un programa para su banca en línea. En este se podrán realizar diversas acciones, como depositar, retirar, consultar saldo o salir. La app deberá estar creada con el lenguaje Swift.

En esta actividad final continuaremos codificando en el lenguaje de programación Swift en nuestra herramienta de trabajo en línea de Replit.

En la actividad anterior, se creó la interfaz del menú principal del programa de Banco Mexicano. Además, se crearon las opciones “Depósito” y “Retiro”. Ahora, se deberá terminar de darle funcionalidad a las opciones faltantes: “Saldo” y “Salir”.

En caso de que el usuario ingrese la opción Saldo, el programa deberá mostrar en pantalla:

- El saldo del usuario
- Preguntar si desea realizar otra operación.

En caso de que el usuario ingrese la opción Salir, el programa deberá mostrar en pantalla que se ha cerrado la sesión.

En esta última actividad de nuestra materia, culminaremos nuestra aplicación bancaria en programación Swift con la herramienta de trabajo Replit.

Es continuación de la aplicación de banco mencionada en la actividad número 2, en donde probaremos los otros 2 menús, que sería consultar el saldo del usuario, preguntar si desea realizar otra operación y la opción de salir de la app.

Vamos a revisar nuestro código fuente y comenzaremos a ejecutarlo para así poder verificar si realmente cuenta con dichas especificaciones antes mencionadas en esta descripción de esta actividad a realizar.

Debo mencionar que estos códigos ya los escribí en la actividad número 2, pero aun así vamos a ejecutar nuestra app para verificar que todo funciona sin ningún problema alguno. Sin más pre ambulo, manos a la obra.

JUSTIFICACION

Swift es un lenguaje de programación de alto nivel desarrollado por Apple. Fue diseñado para ser seguro, rápido y fácil de usar. Swift es un lenguaje de programación moderno que se utiliza para desarrollar aplicaciones para iOS, macOS, watchOS y tvOS. Es un lenguaje de programación de código abierto, lo que significa que cualquier persona puede contribuir a su desarrollo y mejora. Swift es un lenguaje de programación muy popular y se utiliza en muchas aplicaciones populares en la App Store.

La importancia de programar en Swift radica en su facilidad de uso, seguridad y velocidad. Swift es fácil de aprender y usar, lo que lo hace ideal para principiantes y expertos por igual. Además, Swift es un lenguaje de programación seguro que ayuda a prevenir errores comunes en el código. También es muy rápido, lo que significa que las aplicaciones desarrolladas en Swift son muy eficientes.

En resumen, Swift es un lenguaje de programación moderno, seguro y rápido que se utiliza para desarrollar aplicaciones para iOS, macOS, watchOS y tvOS. La facilidad de uso, seguridad y velocidad son algunas de las razones por las que Swift es tan popular entre los desarrolladores.

Ventajas de programar en lenguaje de programación :

- Proceso de desarrollo más rápido.
- Rendimiento y seguridad.
- Gestión de memoria simplificada.
- Soporte multidispositivo.
- Comunidad de código abierto y una alta gama de recursos.

Por estas razones es recomendable el poder escribir programación en el lenguaje de Swift y es por eso que se le recomienda al público programador a utilizar esta herramienta de trabajo para programadores.

Replit es una plataforma en línea que ofrece un entorno de desarrollo integrado (IDE) para múltiples lenguajes de programación, incluido Swift. Hay varias razones por las que alguien podría optar por utilizar Replit:

- Acceso en línea: No necesitas instalar nada en tu computadora. Puedes acceder a Replit desde cualquier lugar con una conexión a internet y un navegador web, lo que lo hace conveniente para trabajar en proyectos desde diferentes dispositivos.
- Colaboración sencilla: Replit permite compartir proyectos fácilmente con otros, lo que es genial para colaborar en tiempo real o mostrar tu código a otras personas para recibir retroalimentación.
- Configuración simplificada: Configurar el entorno de desarrollo puede ser complicado a veces, pero Replit se encarga de la mayor parte de la configuración, lo que te permite comenzar a programar de inmediato.
- Herramientas integradas: Ofrece una variedad de herramientas integradas, como control de versiones, consola, pre visualización de aplicaciones web, entre otras, lo que facilita el desarrollo y la depuración de código.
- Facilidad para probar y compartir proyectos: Puedes ejecutar y probar tu código directamente en la plataforma sin configurar entornos locales.

Además, compartir tu trabajo es tan simple como compartir un enlace.

En el caso específico de Swift, Replit proporciona un entorno en línea donde puedes escribir, ejecutar y probar código Swift sin tener que preocuparte por la instalación local del entorno de desarrollo.

Sin embargo, la elección de la plataforma de desarrollo depende de tus necesidades y preferencias. Algunos programadores prefieren entornos de desarrollo locales más personalizados, mientras que otros encuentran que las plataformas en línea como Replit son más accesibles y prácticas, especialmente para proyectos pequeños o para aprender un nuevo lenguaje de programación.

INVESTIGACION

Que es el lenguaje de programación de Swift.

Swift es un lenguaje de programación creado por Apple para desarrollar aplicaciones para iOS, Mac, Apple TV y Apple Watch. Es un lenguaje poderoso e intuitivo que ofrece a los desarrolladores más libertad que nunca. Swift es fácil de usar y de código abierto, lo que lo hace ideal para que cualquier persona con una idea pueda hacerla realidad. Los desarrolladores están haciendo cosas asombrosas con Swift. Swift es un lenguaje rápido y eficiente que proporciona información en tiempo real y puede incorporarse fácilmente al código de Objective-C existente. Swift es gratis y de código abierto, y está disponible para desarrolladores, educadores y estudiantes bajo la licencia de código abierto Apache 2.0. Numerosas universidades e instituciones académicas de todo el mundo están enseñando a programar con Swift y Xcode en el Mac, lo que permite a los estudiantes contar con las mejores herramientas para crear aplicaciones increíbles. Además, con el curso gratuito “Desarrollo en Swift” de Apple, dar el paso de la programación amateur a la profesional es más fácil que nunca.

Que es la herramienta de trabajo Replit.

Replit es una plataforma en línea que actúa como un entorno de desarrollo integrado (IDE) basado en la nube. Es compatible con múltiples lenguajes de programación, incluidos JavaScript, Python y HTML. Replit es accesible desde cualquier lugar y no requiere instalación. Además, tiene una comunidad activa y recursos educativos.

DESARROLLO

Codificación

```
import Foundation

class Banco {
    var saldo: Double = 0.0

    //Funcion realizar un deposito

    func realizarDeposito() {
        print("Ingrese la cantidad a depositar:")
        if let cantidad = Double(readLine() ?? "0") {
            saldo += cantidad
            print("Depósito realizado. Saldo actual: \(saldo)")

            // Preguntar si desea realizar otro depósito

            print("¿Desea realizar otro depósito? (Si/No)")
            if let respuesta = readLine(), respuesta.lowercased() == "no" {
                preguntarOtraOperacion()
            } else {
                realizarDeposito()
            }
        } else {
            print("Cantidad no válida. Inténtelo de nuevo.")
            realizarDeposito()
        }
    }
}
```

```

    }
}

// Funcion realizar un retiro

func realizarRetiro() {
    if saldo == 0 {
        print("No cuenta con saldo para realizar un retiro.")
    } else {
        print("Ingrese la cantidad a retirar:")
        if let cantidad = Double(readLine() ?? "0") {
            if cantidad <= saldo {
                saldo -= cantidad
                print("Retiro realizado. Saldo actual: \(saldo)")

                // Preguntar si desea realizar otro retiro

                print("¿Desea realizar otro retiro? (Si/No)")
                if let respuesta = readLine(), respuesta.lowercased() == "no" {
                    preguntarOtraOperacion()
                } else {
                    realizarRetiro()
                }
            } else {
                print("Saldo insuficiente. No se puede realizar el retiro.")
                realizarRetiro()
            }
        } else {

```

```

        print("Cantidad no válida. Inténtelo de nuevo.")

        realizarRetiro()

    }

}

}

//Funcion consultar saldo

func consultarSaldo() {
    print("Saldo actual: \$(saldo)")
    preguntarOtraOperacion()
}

//Funcion salir

func salir() {
    print("Gracias por utilizar nuestro servicio. ¡Hasta pronto!")
    exit(0)
}

//Funcion realizar otra operacion

func preguntarOtraOperacion() {
    print("¿Desea realizar otra operación? (Si/No)")
    if let respuesta = readLine(), respuesta.lowercased() == "si" {
        mostrarMenu()
    } else {

```

```

        salir()
    }
}

//Funcion mostrar menu

func mostrarMenu() {
    print("Menú:")
    print("1. Depósito")
    print("2. Retiro")
    print("3. Saldo")
    print("4. Salir")

    print("Ingrese la opción deseada:")
    if let opcion = Int(readLine() ?? "0") {
        switch opcion {
            case 1:
                realizarDeposito()
            case 2:
                realizarRetiro()
            case 3:
                consultarSaldo()
            case 4:
                salir()
            default:
                print("Opción no válida. Inténtelo de nuevo.")
                mostrarMenu()
        }
    }
}

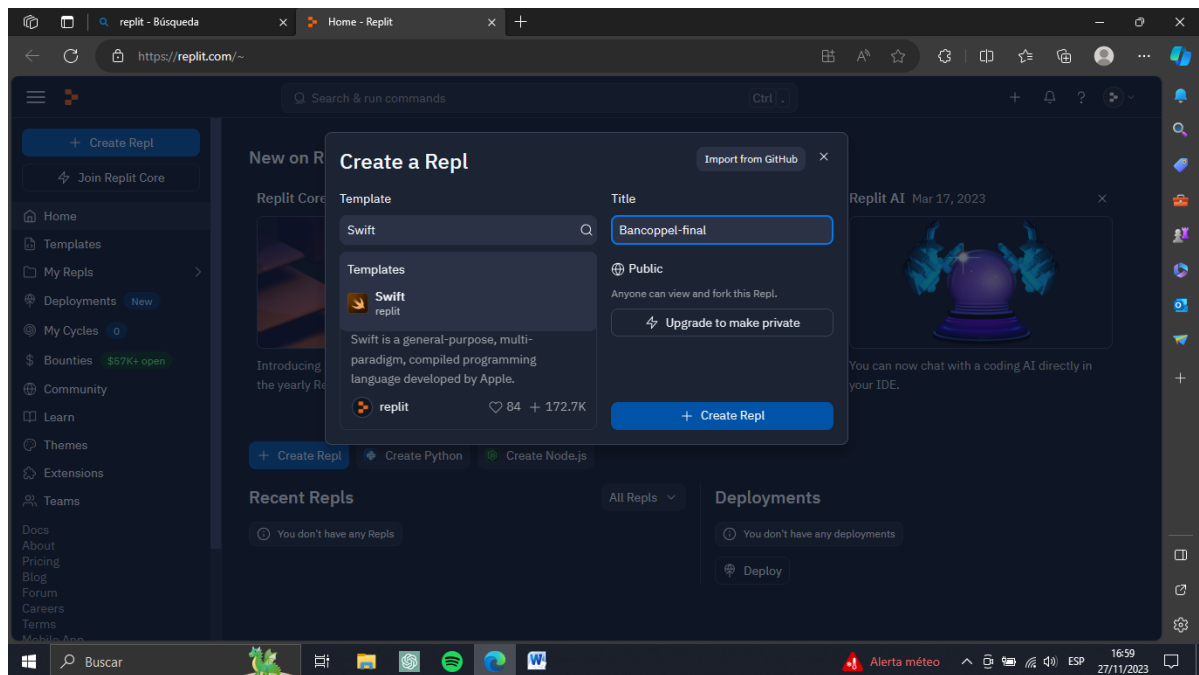
```

```
    }  
  } else {  
    print("Opción no válida. Inténtelo de nuevo.")  
    mostrarMenu()  
  }  
}  
}
```

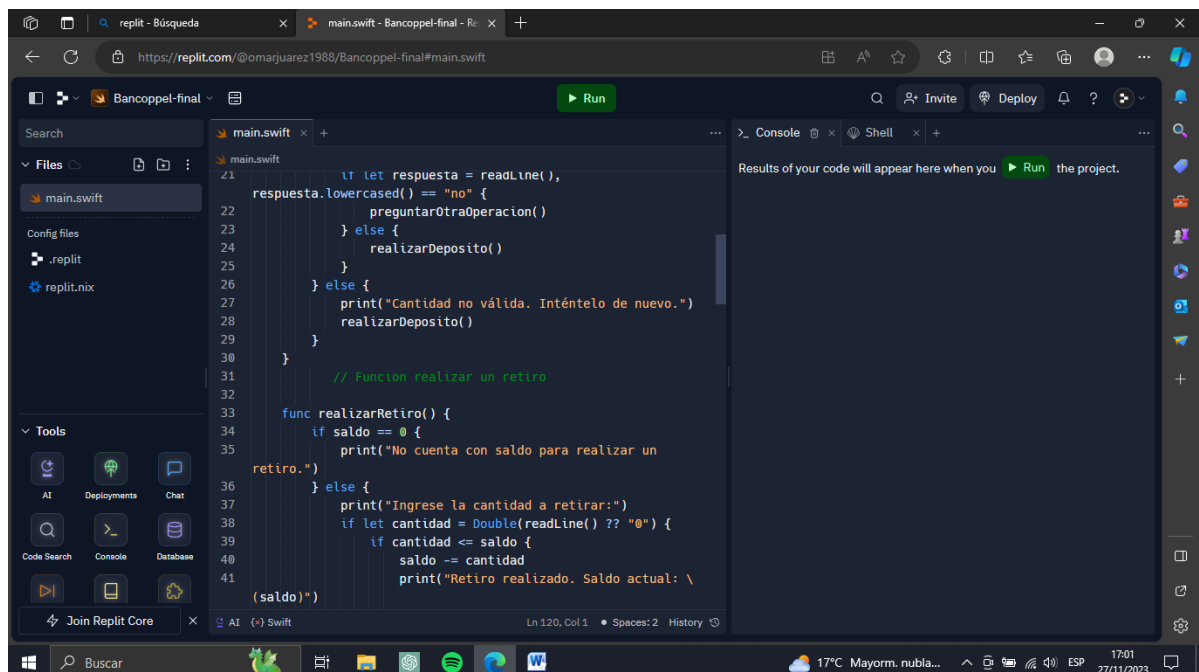
```
let banco = Banco()  
banco.mostrarMenu()
```

Prueba del programa

Creando Replit



Codificando proyecto final de Bancoppel



Probando app con opción 1 (Deposito)

```
main.swift
21 let respuesta = readLine(),
22 respuesta.lowercased() == "no" {
23     preguntarOtraOperacion()
24 } else {
25     realizarDeposito()
26 }
27 } else {
28     print("Cantidad no válida. Inténtelo de nuevo.")
29     realizarDeposito()
30 }
31 // Funcion realizar un retiro
32
33 func realizarRetiro() {
34     if saldo == 0 {
35         print("No cuenta con saldo para realizar un
36         retiro.")
37     } else {
38         print("Ingrese la cantidad a retirar:")
39         if let cantidad = Double(readLine() ?? "") {
40             if cantidad <= saldo {
41                 saldo -= cantidad
42                 print("Retiro realizado. Saldo actual: \
43                 (saldo)")
44             }
45         }
46     }
47 }
```

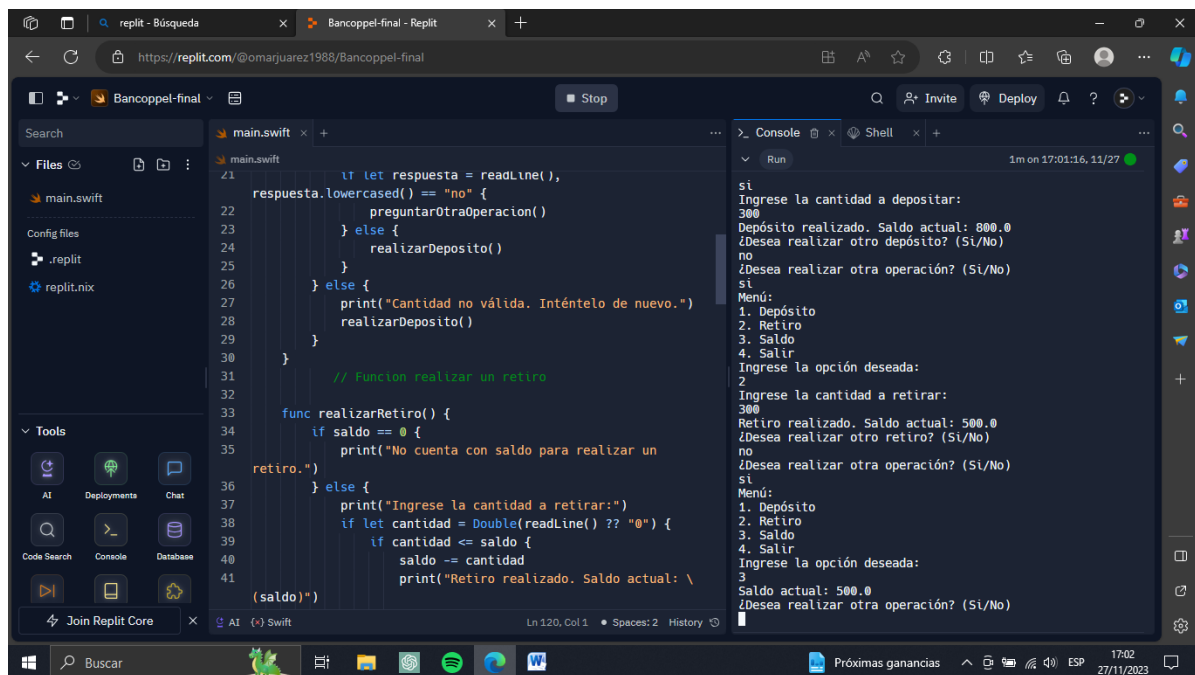
```
Console
Run
26s on 17:01:16, 11/27
Menú:
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese la opción deseada:
1
Ingrese la cantidad a depositar:
500
Depósito realizado. Saldo actual: 500.0
¿Desea realizar otro depósito? (Si/No)
si
Ingrese la cantidad a depositar:
300
Depósito realizado. Saldo actual: 800.0
¿Desea realizar otro depósito? (Si/No)
```

Probando app con opción 2 (Retiro)

```
main.swift
21 let respuesta = readLine(),
22 respuesta.lowercased() == "no" {
23     preguntarOtraOperacion()
24 } else {
25     realizarDeposito()
26 }
27 } else {
28     print("Cantidad no válida. Inténtelo de nuevo.")
29     realizarDeposito()
30 }
31 // Funcion realizar un retiro
32
33 func realizarRetiro() {
34     if saldo == 0 {
35         print("No cuenta con saldo para realizar un
36         retiro.")
37     } else {
38         print("Ingrese la cantidad a retirar:")
39         if let cantidad = Double(readLine() ?? "") {
40             if cantidad <= saldo {
41                 saldo -= cantidad
42                 print("Retiro realizado. Saldo actual: \
43                 (saldo)")
44             }
45         }
46     }
47 }
```

```
Console
Run
43s on 17:01:16, 11/27
Menú:
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese la opción deseada:
1
Ingrese la cantidad a depositar:
500
Depósito realizado. Saldo actual: 500.0
¿Desea realizar otro depósito? (Si/No)
si
Ingrese la cantidad a depositar:
300
Depósito realizado. Saldo actual: 800.0
¿Desea realizar otro depósito? (Si/No)
no
¿Desea realizar otra operación? (Si/No)
si
Menú:
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese la opción deseada:
2
Ingrese la cantidad a retirar:
300
Retiro realizado. Saldo actual: 500.0
¿Desea realizar otro retiro? (Si/No)
```

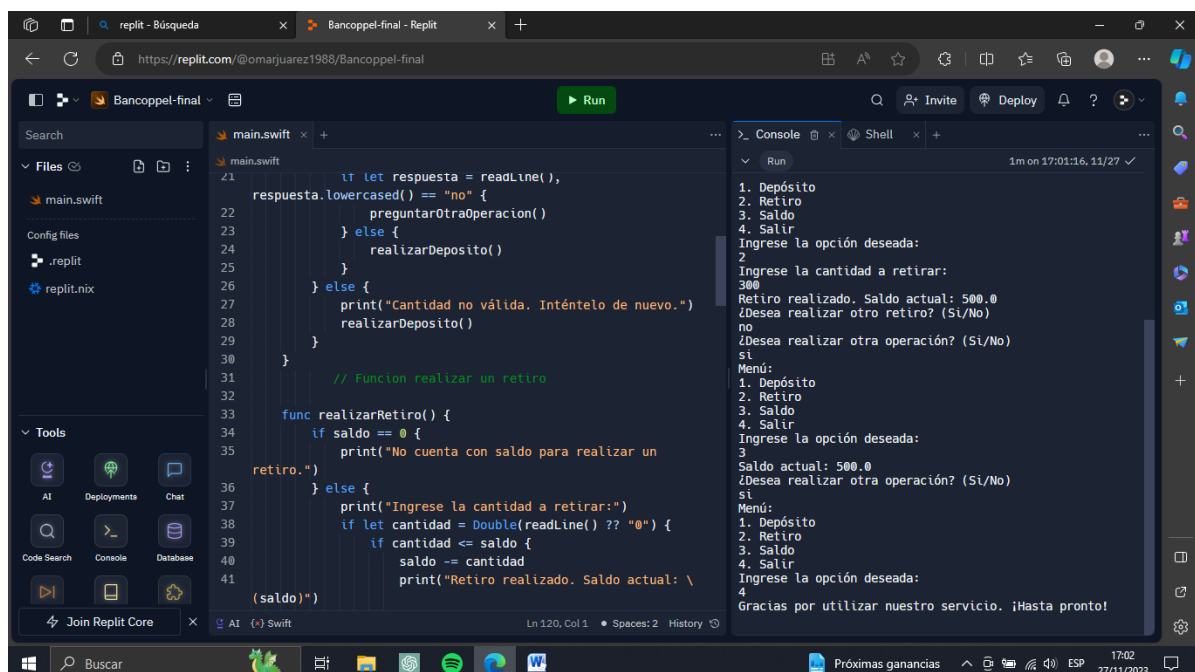

Probando app con opción 3 (Consulta de saldo)



```
main.swift
21 let respuesta = readLine(),
22 respuesta.lowercased() == "no" {
23     preguntarOtraOperacion()
24 } else {
25     realizarDeposito()
26 } else {
27     print("Cantidad no válida. Inténtelo de nuevo.")
28     realizarDeposito()
29 }
30 }
31 // Funcion realizar un retiro
32
33 func realizarRetiro() {
34     if saldo == 0 {
35         print("No cuenta con saldo para realizar un
36         retiro.")
37     } else {
38         print("Ingrese la cantidad a retirar:")
39         if let cantidad = Double(readLine() ?? "") {
40             if cantidad <= saldo {
41                 saldo -= cantidad
42                 print("Retiro realizado. Saldo actual: \
43                 (saldo)")
44             }
45         }
46     }
47 }
```

```
Console
Run
1m on 17:01:16, 11/27
si
Ingrese la cantidad a depositar:
300
Depósito realizado. Saldo actual: 800.0
¿Desea realizar otro depósito? (Si/No)
no
¿Desea realizar otra operación? (Si/No)
si
Menú:
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese la opción deseada:
2
Ingrese la cantidad a retirar:
300
Retiro realizado. Saldo actual: 500.0
¿Desea realizar otro retiro? (Si/No)
no
¿Desea realizar otra operación? (Si/No)
si
Menú:
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese la opción deseada:
3
Saldo actual: 500.0
¿Desea realizar otra operación? (Si/No)
```

Probando app con opción 4 (Salir)



```
main.swift
21 let respuesta = readLine(),
22 respuesta.lowercased() == "no" {
23     preguntarOtraOperacion()
24 } else {
25     realizarDeposito()
26 } else {
27     print("Cantidad no válida. Inténtelo de nuevo.")
28     realizarDeposito()
29 }
30 }
31 // Funcion realizar un retiro
32
33 func realizarRetiro() {
34     if saldo == 0 {
35         print("No cuenta con saldo para realizar un
36         retiro.")
37     } else {
38         print("Ingrese la cantidad a retirar:")
39         if let cantidad = Double(readLine() ?? "") {
40             if cantidad <= saldo {
41                 saldo -= cantidad
42                 print("Retiro realizado. Saldo actual: \
43                 (saldo)")
44             }
45         }
46     }
47 }
```

```
Console
Run
1m on 17:01:16, 11/27
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese la opción deseada:
2
Ingrese la cantidad a retirar:
300
Retiro realizado. Saldo actual: 500.0
¿Desea realizar otro retiro? (Si/No)
no
¿Desea realizar otra operación? (Si/No)
si
Menú:
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese la opción deseada:
3
Saldo actual: 500.0
¿Desea realizar otra operación? (Si/No)
si
Menú:
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese la opción deseada:
4
Gracias por utilizar nuestro servicio. ¡Hasta pronto!
```

CONCLUSION

En esta actividad final de nuestra materia de desarrollo de aplicaciones móviles IV, hemos aprendido a programar con funciones avanzadas en Swift.

Debemos de saber que las funciones en Swift son muy potentes y permiten manipular la lógica de tu aplicación. Cuantas más pequeñas sean las funciones, más fácil serán de reutilizar por toda tu aplicación. Las funciones en Swift agrupan código relacionado para realizar tareas.

Las funciones pueden retornar o no valores. En Swift, las funciones tienen un tipo. El tipo de una función consiste en los tipos de los parámetros de entrada y el tipo de retorno.

Las funciones también pueden devolver tuplas. No las suelo utilizar mucho, pero en ocasiones son muy útiles.

Al usar una función, podemos crear dos nombres para el parámetro. El primero es el nombre del parámetro que se usa desde fuera de la función, y el segundo es para usarlo desde dentro del scope de la función.

La sintaxis de las funciones en Swift se considera bastante flexible; es así como, en la guía de Apple sobre este lenguaje de programación, se menciona que con las funciones en Swift puedes expresar muchas cosas, desde una función simple de estilo C que no tenga nombres de parámetros, hasta un método complejo de estilo Objective-C que tenga nombres y etiquetas de argumentos para cada uno de los parámetros.

Hemos aprendido a crear nuestra aplicación de un banco con funciones en programación Swift, en donde creamos un menú bancario para las diferentes opciones que tiene una app bancaria.

Nos llevamos los conocimientos necesarios para ahora si poder crear desde cero una app de cualquier contextualización en el lenguaje de programación Swift.

Muchas gracias por enriquecernos con sus conocimientos de programación tutor.

REFERENCIAS Y LINK

Sign up. (n.d.). Replit. Retrieved October 30, 2023, from <https://replit.com/~>

SwiftBeta. (2020, August 25). *FUNCIONES en Swift en Español - Curso Swift.*

SwiftBeta. <https://www.swiftbeta.com/funciones-en-swift/>

Swift lenguaje de programación. (n.d.). Bing. Retrieved October 30, 2023, from

<https://www.bing.com/search?q=swift+lenguaje+de+programaci%C3%B3n&qsr=RQ&pq=lenguaje+de+swift&sc=10-17&cvid=FCE2C003BB394F77BA724AA4F440DAFA&FORM=QBRE&sp=1&lq=0&showconv=1>

Chat gpt descargar. (n.d.). Bing. Retrieved November 22, 2023, from

<https://www.bing.com/search?q=chat+gpt+descargar&qsr=n&form=QBRE&sp=-1&ghc=1&lq=0&pq=chat+gpt+descargar&sc=11-18&sk=&cvid=C5E7D28F436645C6BC9450AE0221C283&ghsh=0&ghacc=0&ghpl>

LINKS

GitHub - Drive - Replit

[Omarsitho1988 \(github.com\)](https://github.com/Omarsitho1988)

https://drive.google.com/drive/folders/1zuQGG7Yr_K4HYqVS2uj569fGpslifYol?usp=sharing

[Bancoppel-final - Replit](#)