

Actividad | 3 | Aplicación 3.

Desarrollo de Aplicaciones Móviles III.

Ingeniería en Desarrollo de
Software.



TUTOR: Sandra Luz Lara Dévora

ALUMNO: Carlos Ariel Nicolini

FECHA: 08/11/2025

Índice

Introducción	3
Descripción	4
Justificación	5
Desarrollo.....	6
• Codificación	6
• Prueba de la aplicación.....	11
Conclusión.....	14
Referencias.....	15

Introducción

Una gran ventaja de Swift es su facilidad de aprendizaje. Además, es lo suficientemente potente como para usarse en entornos profesionales. Su sintaxis clara lo hace ideal para principiantes. Además, cuenta con funciones avanzadas como seguridad de tipo y gestión de memoria. Estas ayudan a los desarrolladores de Swift a crear aplicaciones fiables y de alto rendimiento.

Chris Lattner y su equipo de desarrolladores de Apple comenzaron a trabajar en Swift en 2010. Su objetivo era crear un lenguaje de programación con la velocidad de los lenguajes compilados, pero tan fácil de usar como los lenguajes de script. En 2014, el equipo lo logró, lanzaron Swift en la WWDC de 2014 para su primera aparición. En 2015, este potente e intuitivo lenguaje se convirtió en código abierto, y todo el mundo de la programación pudo participar.

Aunque Objective-C ha sido el principal lenguaje de programación de código abierto de Apple desde la década del 1980, los desarrolladores prefieren Swift por varias razones.

Descripción

Contextualización:

Se necesita crear una aplicación en lenguaje Swift, el cual debe contar con un menú de opciones para calcular distintas áreas de figuras geométricas.

Actividad:

Crear la aplicación requerida en lenguaje Swift, la cual debe contar con un menú de opciones que tenga las siguientes especificaciones:

Calcular el área de las siguientes figuras:

- Área del cuadrado.
- Área del rectángulo.
- Área del triángulo
- Área del círculo.

Realizar este trabajo resulto más fácil que los anteriores, por estar ya utilizando este lenguaje y tener mucho más contexto sobre su sintaxis y funcionamiento, además de que es impresionante este lenguaje, ya que escribía una parte del código y el programa me definía lo que seguía, donde al principio me quede revisando sin creerlo, pero a continuación me acostumbre lo cual lo hizo muy llevadero y solo realizar las pequeñas modificaciones que creyera necesario, como cambios en la definición de las variables o la distribución del código.

Justificación

En esta oportunidad y como trabajo final de la materia desarrollo de aplicaciones móviles III realizaremos un programa que nos ayude de manera automática al ingresar ciertos datos necesarios calcular el área de ciertas figuras geométricas, en este caso seria de un cuadrado, un rectángulo, un triángulo y un círculo.

Para este trabajo me apoye muchísimo en la clase dada por la profesora, la cual se me hizo muy pero muy completa y sencilla de entender. Cabe mencionar que después de estar trabajando con este lenguaje ya mucha de la sintaxis me es familiar y el manejo de manera predictiva al querer pensar en algo y aplicarlo sobre el código. Otro detalle importantísimo y que me dejó absorto, es que el lenguaje es completamente intuitivo, por que al empezar a poner código él solito me ponía las posibles opciones siguientes lo cual ayuda muchísimo en tiempo de codificación, obvio con pequeñas modificaciones para las variables que uno maneja con cierto nombre y el programa después las recuerda y las aplica.

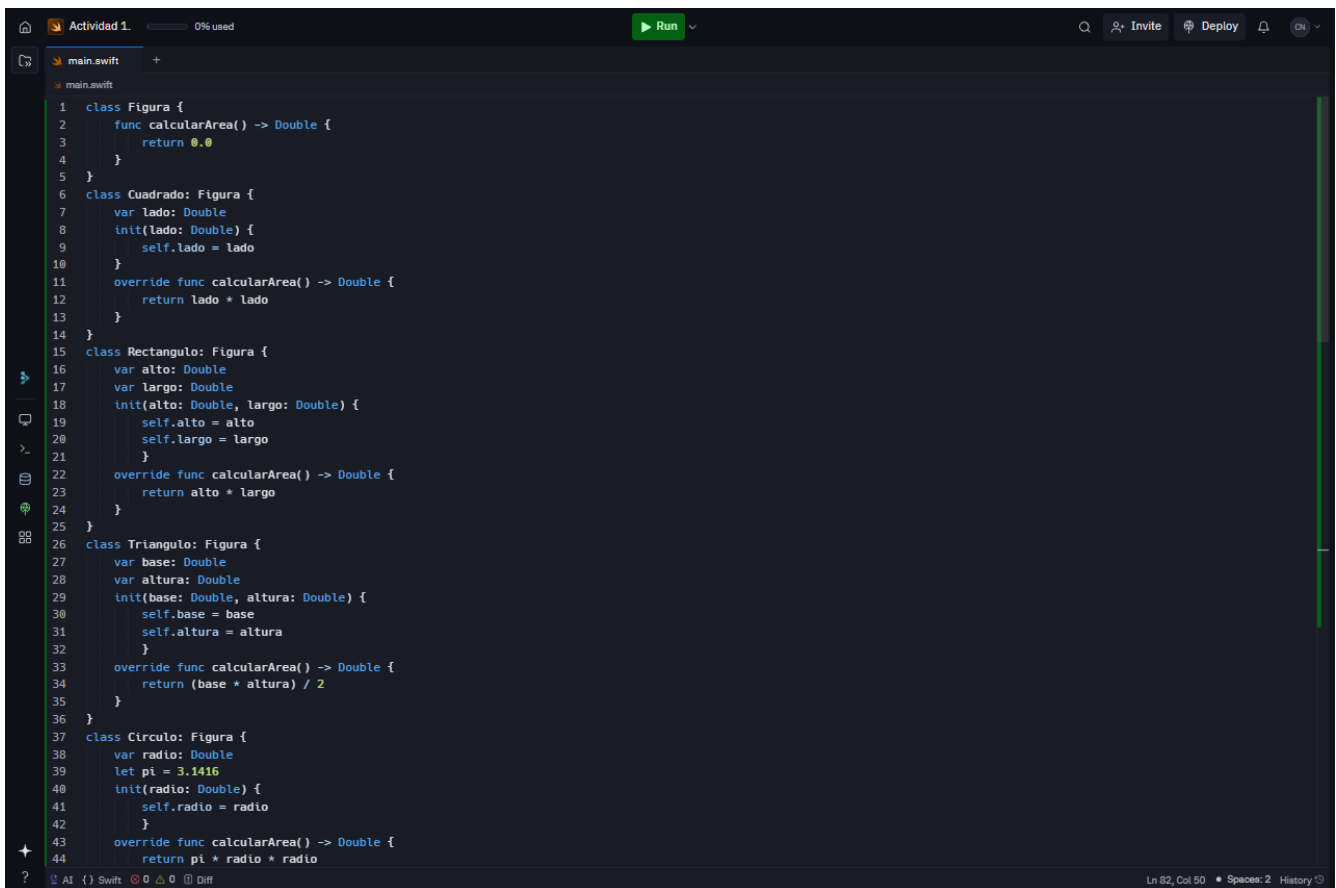
Este trabajo fue subido al siguiente enlace de GitHub

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

Desarrollo

Codificación

En esta parte del ejercicio realizamos una aplicación para calcular el área de ciertas figuras geométricas, en tal caso de un cuadrado, un rectángulo, un triángulo y un círculo. A continuación, se muestra el código realizado y una breve explicación:



```
1 class Figura {
2     func calcularArea() -> Double {
3         return 0.0
4     }
5 }
6 class Cuadrado: Figura {
7     var lado: Double
8     init(lado: Double) {
9         self.lado = lado
10    }
11    override func calcularArea() -> Double {
12        return lado * lado
13    }
14 }
15 class Rectangulo: Figura {
16     var alto: Double
17     var largo: Double
18     init(alto: Double, largo: Double) {
19         self.alto = alto
20         self.largo = largo
21     }
22     override func calcularArea() -> Double {
23         return alto * largo
24     }
25 }
26 class Triangulo: Figura {
27     var base: Double
28     var altura: Double
29     init(base: Double, altura: Double) {
30         self.base = base
31         self.altura = altura
32     }
33     override func calcularArea() -> Double {
34         return (base * altura) / 2
35     }
36 }
37 class Circulo: Figura {
38     var radio: Double
39     let pi = 3.1416
40     init(radio: Double) {
41         self.radio = radio
42     }
43     override func calcularArea() -> Double {
44         return pi * radio * radio
45     }
46 }
```

```

Actividad 1 0% used Run
main.swift
main.swift
37 class Circulo: Figura {
38     var radio: Double
39     let pi = 3.1416
40     init(radio: Double) {
41         self.radio = radio
42     }
43     override func calcularArea() -> Double {
44         return pi * radio * radio
45     }
46 }
47
48 var opcion: Int = 0
49 var bandera = true
50 repeat {
51     print("Menu de opciones")
52     print("1.- Cuadrado")
53     print("2.- Rectangulo")
54     print("3.- Triangulo")
55     print("4.- Circulo")
56     print("5.- Salir")
57     let opcion = Int(readLine()!)?
58     switch opcion {
59         case 1:
60             print("Ingrese el lado del cuadrado")
61             let xlado = Double(readLine()!)?
62             let calculadorDeAreas = Cuadrado(lado: xlado)
63             print("El area del cuadrado es: \(calculadorDeAreas.calcularArea())")
64             print("")
65         case 2:
66             print("Ingrese la altura del rectangulo")
67             let xalto = Double(readLine()!)?
68             print("Ingrese la base del rectangulo")
69             let xlargo = Double(readLine()!)?
70             let calculadorDeAreas = Rectangulo(alto: xalto, largo: xlargo)
71             print("El area del rectangulo es: \(calculadorDeAreas.calcularArea())")
72             print("")
73         case 3:
74             print("Ingrese la base del triangulo")
75             let xbase = Double(readLine()!)?
76             print("Ingrese la altura del triangulo")
77             let xaltura = Double(readLine()!)?
78             let calculadorDeAreas = Triangulo(base: xbase, altura: xaltura)
79             print("El area del triangulo es: \(calculadorDeAreas.calcularArea())")
80             print("")
81     }
82 } while bandera == true

```

```

Actividad 1 0% used Run
main.swift
main.swift
61     let xlado = Double(readLine()!)?
62     let calculadorDeAreas = Cuadrado(lado: xlado)
63     print("El area del cuadrado es: \(calculadorDeAreas.calcularArea())")
64     print("")
65
66     case 2:
67         print("Ingrese la altura del rectangulo")
68         let xalto = Double(readLine()!)?
69         print("Ingrese la base del rectangulo")
70         let xlargo = Double(readLine()!)?
71         let calculadorDeAreas = Rectangulo(alto: xalto, largo: xlargo)
72         print("El area del rectangulo es: \(calculadorDeAreas.calcularArea())")
73         print("")
74     case 3:
75         print("Ingrese la base del triangulo")
76         let xbase = Double(readLine()!)?
77         print("Ingrese la altura del triangulo")
78         let xaltura = Double(readLine()!)?
79         let calculadorDeAreas = Triangulo(base: xbase, altura: xaltura)
80         print("El area del triangulo es: \(calculadorDeAreas.calcularArea())")
81         print("")
82     case 4:
83         print("Ingrese el radio del circulo")
84         let xradio = Double(readLine()!)?
85         let calculadorDeAreas = Circulo(radio: xradio)
86         print("El area del circulo es: \(calculadorDeAreas.calcularArea())")
87         print("")
88     case 5:
89         print("Saliendo del programa")
90         bandera = false
91     default:
92         print("Opcion no valida")
93         print("")
94 } while bandera == true
95

```

A continuación, explicaremos lo que realiza el código:

-

```
1 class Figura {  
2     func calcularArea() -> Double {  
3         return 0.0  
4     }  
5 }
```

Se define una clase padre, se define una función calcularArea() que devuelve un area predeterminada de 0.0 y esta función se sobrescribe sobre las clases hijas.

-

```
6 class Cuadrado: Figura {  
7     var lado: Double  
8     init(lado: Double) {  
9         self.lado = lado  
10    }  
11    override func calcularArea() -> Double {  
12        return lado * lado  
13    }  
14 }  
15 class Rectangulo: Figura {  
16     var alto: Double  
17     var largo: Double  
18     init(alto: Double, largo: Double) {  
19         self.alto = alto  
20         self.largo = largo  
21     }  
22     override func calcularArea() -> Double {  
23         return alto * largo  
24     }  
25 }  
26 class Triangulo: Figura {  
27     var base: Double  
28     var altura: Double  
29     init(base: Double, altura: Double) {  
30         self.base = base  
31         self.altura = altura  
32     }  
33     override func calcularArea() -> Double {  
34         return (base * altura) / 2  
35     }  
36 }  
37 class Circulo: Figura {  
38     var radio: Double  
39     let pi = 3.1416  
40     init(radio: Double) {  
41         self.radio = radio  
42     }  
43     override func calcularArea() -> Double {  
44         return pi * radio * radio  
45     }  
46 }
```


Se crea el menú interactivo en el que se muestran las opciones que se espera que el usuario elija (opción del 1 al 5) y se repite mientras la bandeja será true.

-

```

58     switch opcion {
59         case 1:
60             print("Ingrese el lado del cuadrado")
61             let xlado = Double(readLine()!!)
62             let calculadorDeAreas = Cuadrado (lado: xlado)
63             print("El area del cuadrado es: \(calculadorDeAreas.calcularArea())")
64             print("")
65         case 2:
66             print("Ingrese la altura del rectangulo")
67             let xalto = Double(readLine()!!)
68             print("Ingrese la base del rectangulo")
69             let xlargo = Double(readLine()!!)
70             let calculadorDeAreas = Rectangulo (alto: xalto, largo: xlargo)
71             print("El area del rectangulo es: \(calculadorDeAreas.calcularArea())")
72             print("")
73         case 3:
74             print("Ingrese la base del triangulo")
75             let xbase = Double(readLine()!!)
76             print("Ingrese la altura del triangulo")
77             let xaltura = Double(readLine()!!)
78             let calculadorDeAreas = Triangulo (base: xbase, altura: xaltura)
79             print("El area del triangulo es: \(calculadorDeAreas.calcularArea())")
80             print("")
81         case 4:
82             print("Ingrese el radio del circulo")
83             let xradio = Double(readLine()!!)
84             let calculadorDeAreas = Circulo (radio: xradio)
85             print("El area del circulo es: \(calculadorDeAreas.calcularArea())")
86             print("")
87         case 5:
88             print("Saliendo del programa")
89             bandera = false
90         default:
91             print("Opcion no valida")
92             print("")
93     }
94 } while bandera == true

```

Se crean los casos de uso, donde según elija el usuario (opción de 1 al 4) el programa pide los valores necesarios, crea el objeto de la figura elegida, calcula el área y la muestra. El caso de uso 5 termina el programa y si se elige un número que no sea del 1 al 5 el programa enviará un mensaje de opción no valida y pedirá ingresar nuevamente un número del 1 al 5.

Prueba de la aplicación

A continuación, realizaremos las pruebas del aplicativo

Al iniciar el programa, elegimos la opción de cuadrado (1), ingresamos el lado del cuadrado como es solicitado (50) y nos da de resultado automáticamente el área del cuadrado (2500).

The screenshot shows a Swift IDE with a file named 'main.swift' and a console window. The code is a Swift program that calculates the area of a square. It starts with a menu of options: 1.- Cuadrado, 2.- Rectangulo, 3.- Triangulo, 4.- Circulo, 5.- Salir. The user selects option 1. The program prompts for the side of the square, and the user enters 50. The program then calculates the area (2500.0) and displays it. The console output shows the menu, the selection of option 1, the prompt for the side, the input 50, and the calculated area 2500.0.

```

61 let xlado = Double(readLine())!
62 let calculadorDeAreas = Cuadrado (lado: xlado)
63 print("El area del cuadrado es: \(calculadorDeAreas.calcularArea())")
64 print("")
65 case 2:
66 print("Ingrese la altura del rectangulo")
67 let xalto = Double(readLine())!
68 print("Ingrese la base del rectangulo")
69 let xlargo = Double(readLine())!
70 let calculadorDeAreas = Rectangulo (alto: xalto, largo: xlargo)
71 print("El area del rectangulo es: \(calculadorDeAreas.calcularArea())")
72 print("")
73 case 3:
74 print("Ingrese la base del triangulo")
75 let xbase = Double(readLine())!
76 print("Ingrese la altura del triangulo")
77 let xaltura = Double(readLine())!
78 let calculadorDeAreas = Triangulo (base: xbase, altura: xaltura)
79 print("El area del triangulo es: \(calculadorDeAreas.calcularArea())")
80 print("")
81 case 4:
82 print("Ingrese el radio del circulo")
83 let xradio = Double(readLine())!
84 let calculadorDeAreas = Circulo (radio: xradio)
85 print("El area del circulo es: \(calculadorDeAreas.calcularArea())")
86 print("")
87 case 5:
88 print("Saliendo del programa")
89 bandera = false
90 default:
91 print("Opcion no valida")
92 print("")
93 }
94 } while bandera == true
95

```

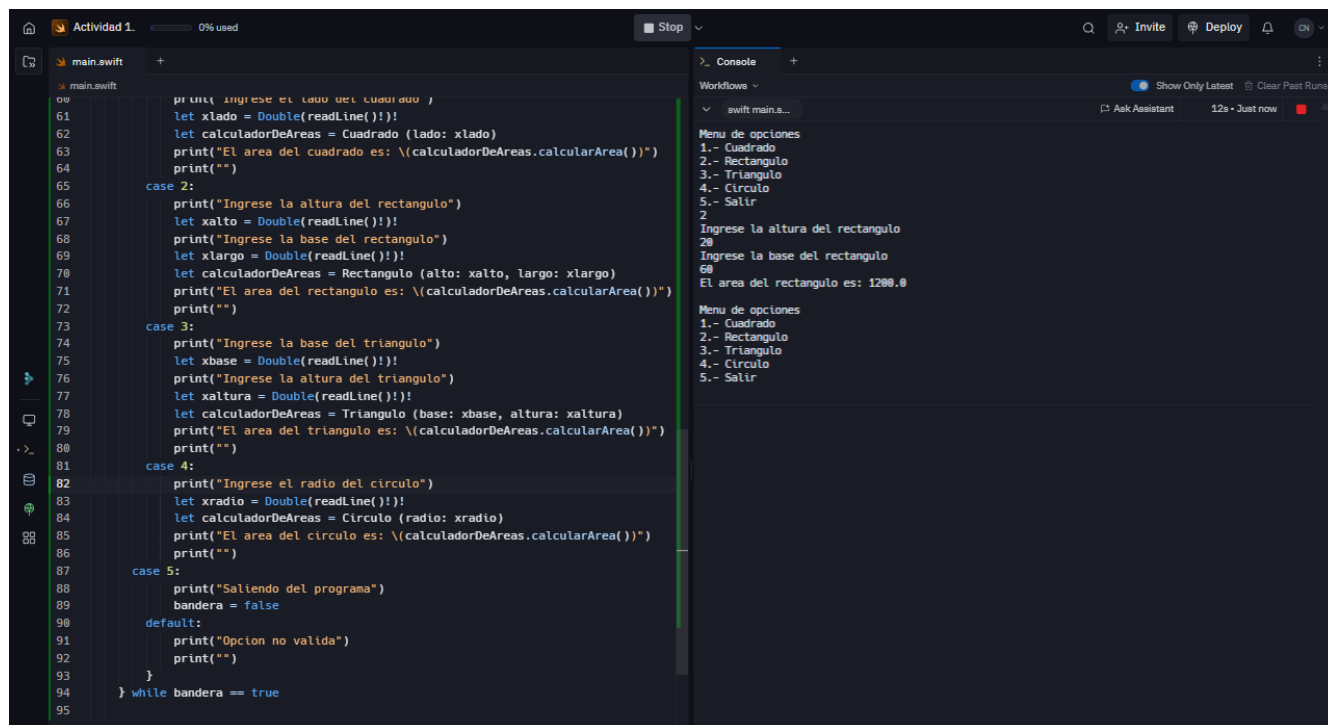
Console Output:

```

Menu de opciones
1.- Cuadrado
2.- Rectangulo
3.- Triangulo
4.- Circulo
5.- Salir
1
Ingrese el lado del cuadrado
50
El area del cuadrado es: 2500.0
Menu de opciones
1.- Cuadrado
2.- Rectangulo
3.- Triangulo
4.- Circulo
5.- Salir

```

A continuación elegimos la opción del rectángulo (2), a lo cual nos solicita ingresar la altura rectángulo (20) y además ingresamos la base del rectángulo (60), lo que nos da de resultado el área del rectángulo (1200)



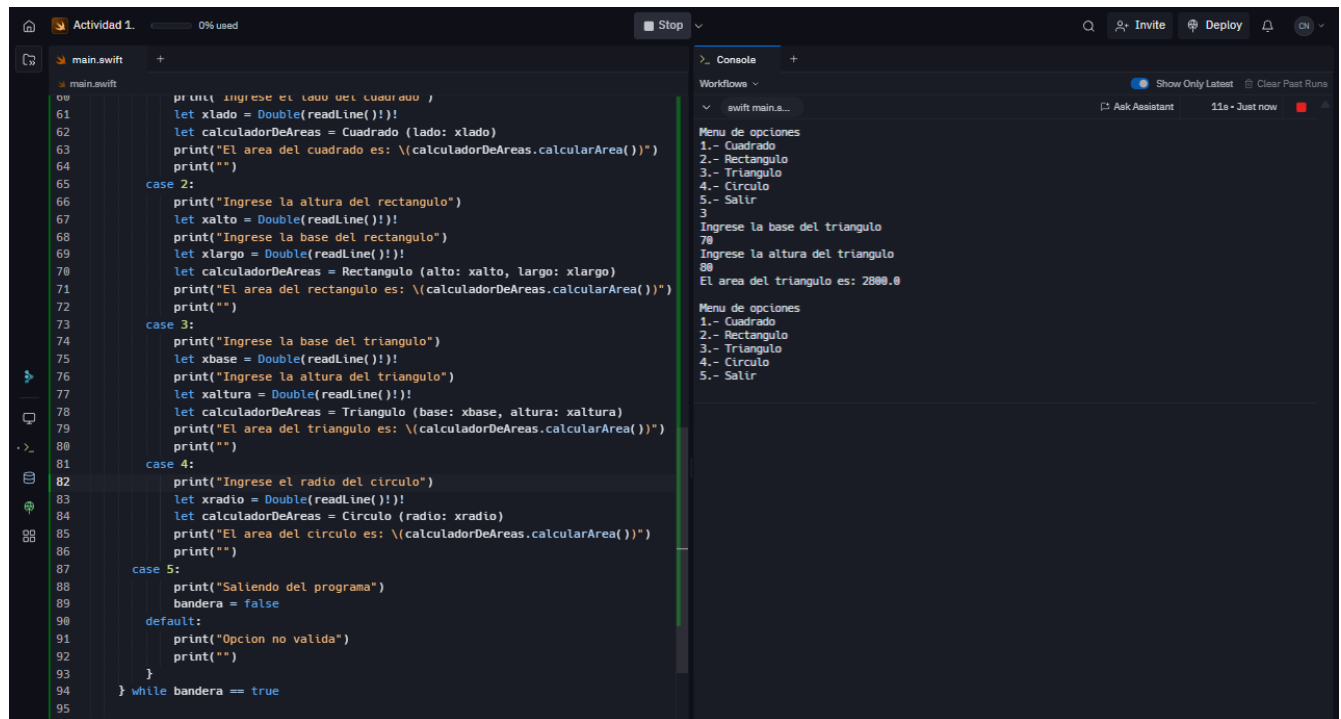
```
Actividad 1 0% used
main.swift
main.swift
61 print("Ingrese el lado del cuadrado ")
62 let xlado = Double(readLine()!)?
63 let calculadorDeAreas = Cuadrado (lado: xlado)
64 print("El area del cuadrado es: \(calculadorDeAreas.calcularArea())")
65 print("")
66 case 2:
67 print("Ingrese la altura del rectangulo")
68 let xalto = Double(readLine()!)?
69 print("Ingrese la base del rectangulo")
70 let xlarg = Double(readLine()!)?
71 let calculadorDeAreas = Rectangulo (alto: xalto, largo: xlarg)
72 print("El area del rectangulo es: \(calculadorDeAreas.calcularArea())")
73 print("")
74 case 3:
75 print("Ingrese la base del triangulo")
76 let xbase = Double(readLine()!)?
77 print("Ingrese la altura del triangulo")
78 let xaltura = Double(readLine()!)?
79 let calculadorDeAreas = Triangulo (base: xbase, altura: xaltura)
80 print("El area del triangulo es: \(calculadorDeAreas.calcularArea())")
81 print("")
82 case 4:
83 print("Ingrese el radio del circulo")
84 let xradio = Double(readLine()!)?
85 let calculadorDeAreas = Circulo (radio: xradio)
86 print("El area del circulo es: \(calculadorDeAreas.calcularArea())")
87 print("")
88 case 5:
89 print("Saliendo del programa")
90 bandera = false
91 default:
92 print("Opcion no valida")
93 print("")
94 }
95 } while bandera == true
```

Workflow: swift main.s... Show Only Latest Clear Past Runs

Menu de opciones
1.- Cuadrado
2.- Rectangulo
3.- Triangulo
4.- Circulo
5.- Salir
2
Ingrese la altura del rectangulo
20
Ingrese la base del rectangulo
60
El area del rectangulo es: 1200.0

Menu de opciones
1.- Cuadrado
2.- Rectangulo
3.- Triangulo
4.- Circulo
5.- Salir

Cuando elegimos la opción del triángulo (3) nos pide ingresar la base del triángulo (70) y también la altura del triángulo (80), a lo cual nos indica que el área del triángulo es 2800



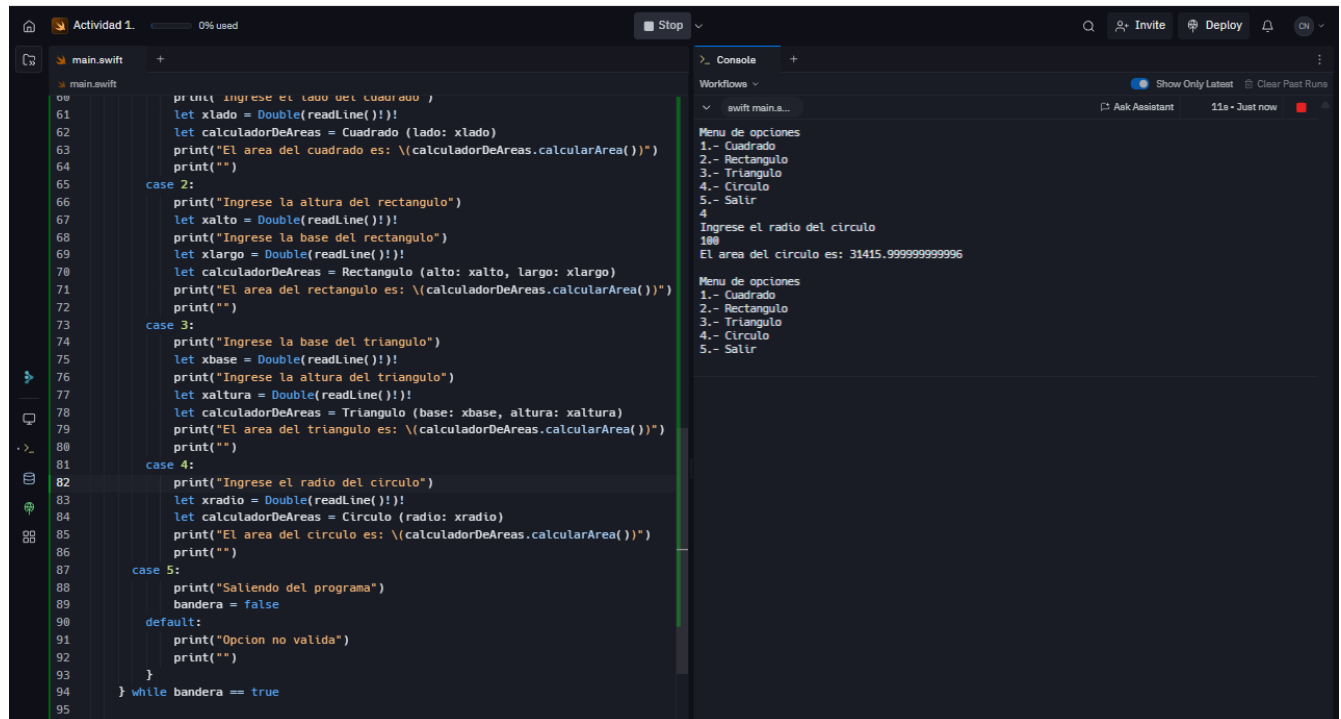
```
Actividad 1 0% used
main.swift
main.swift
61 print("Ingrese el lado del cuadrado")
62 let xlado = Double(readLine()!)
63 let calculadorDeAreas = Cuadrado(lado: xlado)
64 print("El area del cuadrado es: \(calculadorDeAreas.calcularArea())")
65 print("")
66 case 2:
67 print("Ingrese la altura del rectangulo")
68 let xalto = Double(readLine()!)
69 print("Ingrese la base del rectangulo")
70 let xlarg = Double(readLine()!)
71 let calculadorDeAreas = Rectangulo(alto: xalto, largo: xlarg)
72 print("El area del rectangulo es: \(calculadorDeAreas.calcularArea())")
73 print("")
74 case 3:
75 print("Ingrese la base del triangulo")
76 let xbase = Double(readLine()!)
77 print("Ingrese la altura del triangulo")
78 let xaltura = Double(readLine()!)
79 let calculadorDeAreas = Triangulo(base: xbase, altura: xaltura)
80 print("El area del triangulo es: \(calculadorDeAreas.calcularArea())")
81 print("")
82 case 4:
83 print("Ingrese el radio del circulo")
84 let xradio = Double(readLine()!)
85 let calculadorDeAreas = Circulo(radio: xradio)
86 print("El area del circulo es: \(calculadorDeAreas.calcularArea())")
87 print("")
88 case 5:
89 print("Saliendo del programa")
90 bandera = false
91 default:
92 print("Opcion no valida")
93 print("")
94 }
95 } while bandera == true
```

Workflow: swift main.s... Show Only Latest Clear Past Runs

Menu de opciones
1.- Cuadrado
2.- Rectangulo
3.- Triangulo
4.- Circulo
5.- Salir
3
Ingrese la base del triangulo
70
Ingrese la altura del triangulo
80
El area del triangulo es: 2800.0

Menu de opciones
1.- Cuadrado
2.- Rectangulo
3.- Triangulo
4.- Circulo
5.- Salir

Ingresamos a la opción para calcular el área de un círculo (4), a lo cual nos solicita agregar el radio del círculo (100) y nos da el área del círculo de manera automática (31415.999999)



```
main.swift
ow
61 print("Ingrese el lado del cuadrado ")
62 let xlado = Double(readLine()!)
63 let calculadorDeAreas = Cuadrado (lado: xlado)
64 print("El area del cuadrado es: \(calculadorDeAreas.calcularArea())")
65 print("")
66 case 2:
67 print("Ingrese la altura del rectangulo")
68 let xalto = Double(readLine()!)
69 print("Ingrese la base del rectangulo")
70 let xlarg = Double(readLine()!)
71 let calculadorDeAreas = Rectangulo (alto: xalto, largo: xlarg)
72 print("El area del rectangulo es: \(calculadorDeAreas.calcularArea())")
73 print("")
74 case 3:
75 print("Ingrese la base del triangulo")
76 let xbase = Double(readLine()!)
77 print("Ingrese la altura del triangulo")
78 let xaltura = Double(readLine()!)
79 let calculadorDeAreas = Triangulo (base: xbase, altura: xaltura)
80 print("El area del triangulo es: \(calculadorDeAreas.calcularArea())")
81 print("")
82 case 4:
83 print("Ingrese el radio del circulo")
84 let xradio = Double(readLine()!)
85 let calculadorDeAreas = Circulo (radio: xradio)
86 print("El area del circulo es: \(calculadorDeAreas.calcularArea())")
87 print("")
88 case 5:
89 print("Saliendo del programa")
90 bandera = false
91 default:
92 print("Opcion no valida")
93 print("")
94 }
95 } while bandera == true
```

Console

Workflows

swift main.a...

Show Only Latest Clear Past Runs

Ask Assistant 11s • Just now

Menu de opciones

1.- Cuadrado

2.- Rectangulo

3.- Triangulo

4.- Circulo

5.- Salir

Ingrese el radio del circulo

100

El area del circulo es: 31415.99999999996

Menu de opciones

1.- Cuadrado

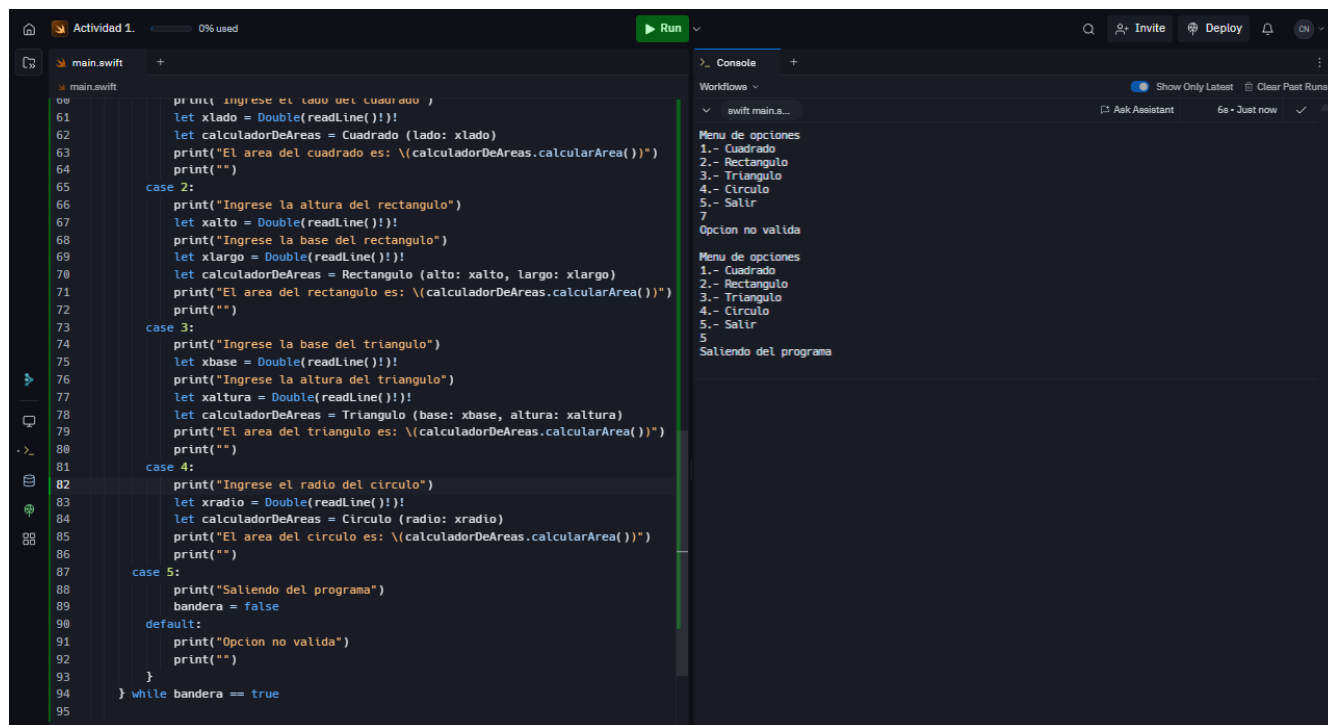
2.- Rectangulo

3.- Triangulo

4.- Circulo

5.- Salir

En este caso mostramos como al ingresar el numero 6 (el cual no existe en el menú) nos marca opción no valida y nuevamente nos pide ingresar un numero del menú. A continuación, elegimos la opción 5 y el programa se termina mostrándonos un mensaje en consola de saliendo del programa.



```
Actividad 1 0% used Run
main.swift
main.swift
60 print("Ingrese el lado del cuadrado ")
61 let xlado = Double(readLine()!))!
62 let calculadorDeAreas = Cuadrado(lado: xlado)
63 print("El area del cuadrado es: \(calculadorDeAreas.calcularArea())")
64 print("")
65 case 2:
66 print("Ingrese la altura del rectangulo")
67 let xalto = Double(readLine()!))!
68 print("Ingrese la base del rectangulo")
69 let xlarg = Double(readLine()!))!
70 let calculadorDeAreas = Rectangulo(alto: xalto, largo: xlarg)
71 print("El area del rectangulo es: \(calculadorDeAreas.calcularArea())")
72 print("")
73 case 3:
74 print("Ingrese la base del triangulo")
75 let xbase = Double(readLine()!))!
76 print("Ingrese la altura del triangulo")
77 let xaltura = Double(readLine()!))!
78 let calculadorDeAreas = Triangulo(base: xbase, altura: xaltura)
79 print("El area del triangulo es: \(calculadorDeAreas.calcularArea())")
80 print("")
81 case 4:
82 print("Ingrese el radio del circulo")
83 let xradio = Double(readLine()!))!
84 let calculadorDeAreas = Circulo(radio: xradio)
85 print("El area del circulo es: \(calculadorDeAreas.calcularArea())")
86 print("")
87 case 5:
88 print("Saliendo del programa")
89 bandera = false
90 default:
91 print("Opcion no valida")
92 print("")
93 }
94 } while bandera == true
95
```

Workflow swift main.s... Show Only Latest Clear Past Runs

Menu de opciones
1.- Cuadrado
2.- Rectangulo
3.- Triangulo
4.- Circulo
5.- Salir
7
Opcion no valida
Menu de opciones
1.- Cuadrado
2.- Rectangulo
3.- Triangulo
4.- Circulo
5.- Salir
5
Saliendo del programa

Conclusión

En toda esta materia empezamos a utilizar el lenguaje Swift y a tratar de dominarlo, conocerlo, acostumbrarnos a sus formas, a utilizar clases, herencias, polimorfismo, encapsular propiedades dentro de cada objeto, utilizar clases hijas y utilizar sobreescritura de métodos.

Esta clase de codificación y programas que se pueden realizar tienen muchos beneficios y cosas en las que se pueden utilizar en el día a día, en cosas sencillas como lo demostró esta materia, en mi área de trabajo puedo realizar programas para pequeños procesos sencillos, que llevan implícito sumas o ciertos comandos, que de esta manera pueden realizarse de manera automática y dar resultados correctos sin tener detalle del factor humano que muchas veces es donde está la falla en procesos sencillos y además, de esta manera se pueden hacer de manera masiva para aplicarse a muchas instancias en una sola vez, además de que realice verificaciones si dicha ejecución se realizó de manera correcta, sino, esperar un cierto periodo de tiempo y volver a intentarlo hasta que se halla ejecutado de manera correcta, además también sirve para configuraciones, donde usuarios le mueven, un programa realizado con este código puede revisar ciertas llaves y si están modificadas, repararlas siempre que las vea incorrectas.

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

Referencias

10 razones para aprender Swift y desarrollar en iOS. (2022, December 3).
<https://platzi.com>; Platzi. <https://platzi.com/blog/10-razones-para-aprender-swift-y-desarrollar-en-ios/>

Is Swift a good language for a beginner to learn? (2022, August 22). Swift Forums.
<https://forums.swift.org/t/is-swift-a-good-language-for-a-beginner-to-learn/59814>

Master the Swift basics: A quick guide for new programmers. (2025, March 5). BairesDev.
<https://www.bairesdev.com/blog/swift-basics/>