

Actividad 3 - Aplicación 3.


Desarrollo de Aplicaciones Móviles 3.

Ingeniería en Desarrollo de Software

Tutor: Sandra Luz Lara Dévora.

Alumno: Homero Ramirez Hurtado.

Fecha: 06 de Agosto del 2024.



Índice.

. Introducción.

. Descripción.

. Justificación.

. Desarrollo.

- Codificación.
- Prueba de la Aplicación.

. Conclusión.

. Referencias.



Introducción.

Título: Calculadora de Áreas Geométricas en Swift

En este proyecto, desarrollaremos una aplicación en lenguaje Swift que permitirá a los usuarios calcular áreas de diferentes figuras geométricas. La aplicación tendrá un menú de opciones intuitivo y fácil de usar.

Pasos clave:

1. Diseño de la Interfaz de Usuario:
 - Crearemos una pantalla inicial con un menú que presente las opciones disponibles (por ejemplo, círculo, triángulo, cuadrado, etc.).
 - Cada opción llevará al usuario a una pantalla específica para ingresar los datos necesarios.
2. Entrada de Datos:
 - En cada pantalla específica, solicitaremos al usuario los valores necesarios (como radio, base, altura, etc.).
 - Validaremos que los datos ingresados sean numéricos y positivos.
3. Cálculo y Resultado:
 - Realizaremos el cálculo del área según la fórmula seleccionada.
 - Mostraremos el resultado al usuario en una pantalla aparte.
4. Replit:
 - Utilizaremos Replit para desarrollar y probar nuestra aplicación.
 - Replit es una plataforma en línea que permite escribir, compilar y ejecutar código Swift directamente en el navegador.

Descripción.

Este emocionante proyecto consiste en desarrollar una aplicación en el lenguaje Swift que permita a los usuarios calcular áreas de diversas figuras geométricas. Imagina tener una herramienta virtual que te ayude a resolver esos enigmas matemáticos relacionados con cuadrados, rectángulos, triángulos y círculos. Aquí está cómo funcionará:

- **Menú de Opciones:** Al abrir la aplicación, los usuarios serán recibidos por un elegante menú. En él, podrán seleccionar la figura cuya área desean calcular. Las opciones estarán dispuestas como las deliciosas tapas en un buffet matemático.
- **Cálculo Mágico:** Una vez que el usuario elija una figura, la aplicación le pedirá los valores necesarios. Por ejemplo, para el cuadrado, solo necesitarán la longitud de un lado. Para el rectángulo, la base y la altura. Y así sucesivamente. La aplicación aplicará las fórmulas mágicas (¡matemáticas!) para encontrar el área exacta.
- **Resultados Instantáneos:** Después de unos segundos de suspense, la aplicación revelará el área calculada. Los usuarios podrán admirar su destreza matemática o, en caso de errores, corregir los valores y volver a intentarlo.

Justificación.

1. Accesibilidad y Comodidad:

- Replit es una plataforma en línea que permite escribir, compilar y ejecutar código Swift directamente en el navegador. Esto significa que no necesitas instalar un entorno de desarrollo local en tu máquina. Puedes comenzar a programar desde cualquier lugar con acceso a Internet.
- Es especialmente útil para proyectos pequeños y pruebas rápidas, como nuestra calculadora de áreas geométricas. No hay necesidad de configurar complicadas herramientas locales; simplemente inicia sesión en Replit y comienza a codificar.

2. Colaboración y Compartición:

- Replit facilita la colaboración en tiempo real. Puedes invitar a otros desarrolladores a trabajar en el mismo proyecto, lo que es ideal para trabajos en equipo o para recibir retroalimentación de tus compañeros.
- Además, puedes compartir fácilmente tu proyecto con otras personas mediante un enlace. Esto es genial para mostrar tu trabajo a profesores, amigos o posibles empleadores.

3. Entorno Preconfigurado:

- Replit proporciona un entorno preconfigurado para varios lenguajes de programación, incluido Swift. No tienes que preocuparte por instalar bibliotecas o configurar herramientas adicionales. Simplemente comienza a escribir código.
- La calculadora de áreas geométricas no requiere una infraestructura compleja, por lo que Replit es una elección perfecta.

4. Retroalimentación Instantánea:

- Replit ofrece una experiencia de desarrollo fluida. A medida que escribes código, puedes ver los resultados en tiempo real. Esto es especialmente útil para depurar y corregir errores rápidamente.
- Imagina calcular el área de un triángulo y ver al instante si tus fórmulas están funcionando correctamente.

5. Sin Barreras de Entrada:

- Replit es amigable para principiantes. Si estás aprendiendo Swift o si este es tu primer proyecto, Replit te quitará la presión de configurar un entorno complicado. Puedes concentrarte en la lógica y la creatividad.

Desarrollo.

Aplicación 1.

Código.

```
class figura{  
    func calculaArea()->Double {  
        return 0.0  
    }  
}  
  
class cuadrado: figura{  
    var lado: Double  
    init(lado: Double){  
        self.lado = lado  
    }  
    override func calculaArea()->Double{  
        return lado*lado  
    }  
}  
  
class rectangulo: figura{  
    var alto: Double  
    var largo: Double  
    init(alto: Double, largo: Double){  
        self.alto = alto  
        self.largo = largo  
    }  
    override func calculaArea()->Double{  
        return alto*largo  
    }  
}
```

```
class triangulo: figura{
    var base: Double
    var altura: Double
    init(base: Double, altura: Double){
        self.base = base
        self.altura = altura
    }
    override func calculaArea()->Double{
        return (base*altura)/2
    }
}

class circulo: figura{
    var radio: Double
    let pi = 3.1416
    init(radio: Double){
        self.radio = radio
    }
    override func calculaArea()->Double{
        return pi*radio*radio
    }
}

var opcion: Int = 0
var bandera = true
repeat{
    print("Menu de opciones: Calcular Area")
    print("1.- cuadrado")
    print("2.- rectangulo")
    print("3.- Triangulo")
}
```

```
print("4.- circulo")
print("5.- salir")
print("Selecciona una opcion: ")
let opcion = Int (readLine()!)
switch opcion{
  case 1:
    print("Ingresa el lado del cuadrado: ")
    let xlado = Double (readLine()!)
    let calculadorDeAreas = cuadrado(lado: xlado)
    print("El area del cuadrado es: \$(calculadorDeAreas.calculaArea())")
  case 2:
    print("Ingresa la altura del rectangulo: ")
    let xalto = Double (readLine()!)
    print("Ingresa la base del rectangulo: ")
    let xlargo = Double (readLine()!)
    let calculadorDeAreas = rectangulo(alto: xalto, largo: xlargo)
    print("El area del rectangulo es: \$(calculadorDeAreas.calculaArea())")
  case 3:
    print("Ingresa la base del triangulo: ")
    let xbase = Double (readLine()!)
    print("Ingresa la altura del triangulo: ")
    let xaltura = Double (readLine()!)
    let calculadorDeAreas = triangulo(base: xbase, altura: xaltura)
    print("El area del triangulo es: \$(calculadorDeAreas.calculaArea())")
  case 4:
    print("Ingresa el radio del circulo: ")
    let xradio = Double (readLine()!)
    let calculadorDeAreas = circulo(radio: xradio)
```

```
print("El area del circulo es: \"(calculadorDeAreas.calculaArea())\"")
```

case 5:

```
print("Saliendo del programa...")
```

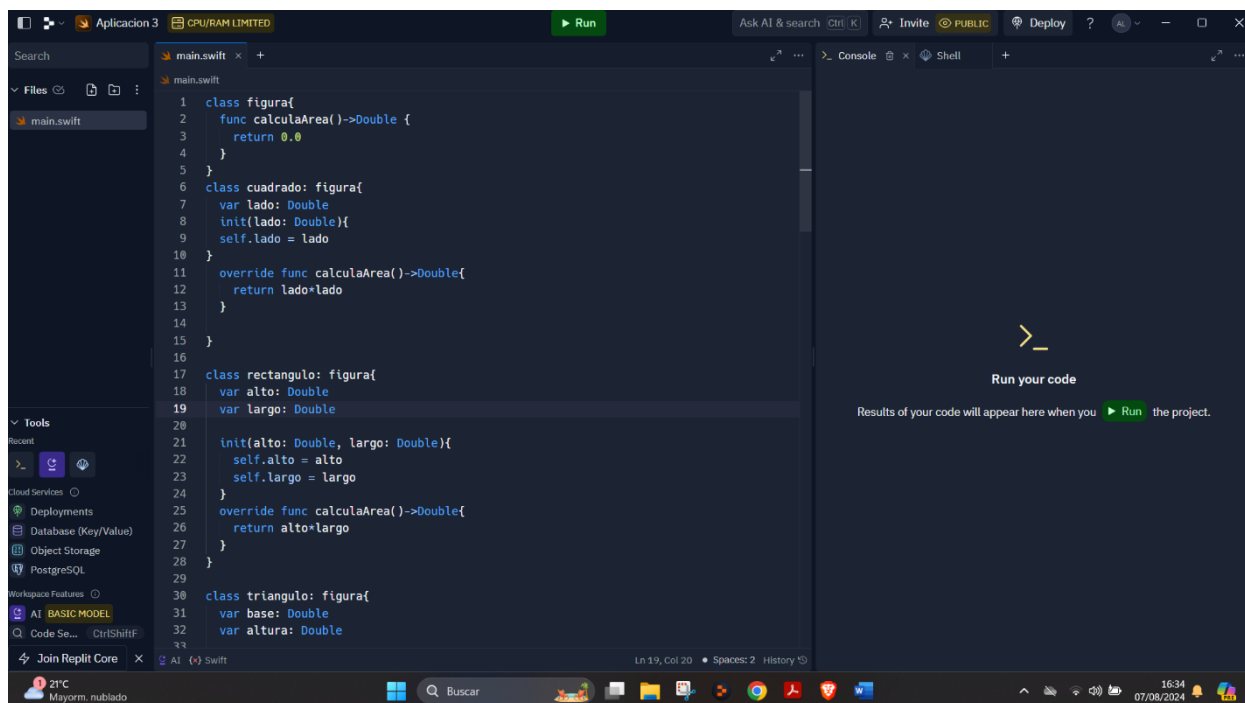
```
bandera = false
```

```
default:
```

```
print("Opcion no valida")
```

```
}
```

```
}while bandera == true
```



```
1 class figura{
2     func calculaArea()->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 calculaArea()->Double{
12        return lado*lado
13    }
14 }
15 }
16
17 class rectangulo: figura{
18     var alto: Double
19     var largo: Double
20
21     init(alto: Double, largo: Double){
22         self.alto = alto
23         self.largo = largo
24     }
25     override func calculaArea()->Double{
26         return alto*largo
27     }
28 }
29
30 class triangulo: figura{
31     var base: Double
32     var altura: Double
33 }
```

Código.


```
1 class figura{
2     func calculaArea()->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 calculaArea()->Double{
12        return lado*lado
13    }
14 }
15 }
16
17 class rectangulo: figura{
18     var alto: Double
19     var largo: Double
20
21     init(alto: Double, largo: Double){
22         self.alto = alto
23         self.largo = largo
24     }
25     override func calculaArea()->Double{
26         return alto*largo
27     }
28 }
29
30 class triangulo: figura{
31     var base: Double
32     var altura: Double
33 }
```

Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- círculo
5.- salir
Selecciona una opcion:
1
Ingresa el lado del cuadrado:
2569
El area del cuadrado es: 6599761.0
Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- círculo
5.- salir
Selecciona una opcion:
[]

Cuadrado.

```
1 class figura{
2     func calculaArea()->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 calculaArea()->Double{
12        return lado*lado
13    }
14 }
15 }
16
17 class rectangulo: figura{
18     var alto: Double
19     var largo: Double
20
21     init(alto: Double, largo: Double){
22         self.alto = alto
23         self.largo = largo
24     }
25     override func calculaArea()->Double{
26         return alto*largo
27     }
28 }
29
30 class triangulo: figura{
31     var base: Double
32     var altura: Double
33 }
```

Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- círculo
5.- salir
Selecciona una opcion:
2
Ingresa la altura del rectangulo:
9863
Ingresa la base del rectangulo:
7584
El area del rectangulo es: 74800992.0
Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- círculo
5.- salir
Selecciona una opcion:
[]

Rectángulo.

The screenshot shows a Swift REPL environment with the following code in `main.swift`:

```
1 class figura{
2     func calculaArea()->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 calculaArea()->Double{
12        return lado*lado
13    }
14 }
15 }
16
17 class rectangulo: figura{
18     var alto: Double
19     var largo: Double
20
21     init(alto: Double, largo: Double){
22         self.alto = alto
23         self.largo = largo
24     }
25     override func calculaArea()->Double{
26         return alto*largo
27     }
28 }
29
30 class triangulo: figura{
31     var base: Double
32     var altura: Double
33 }
```

The console output shows the execution of the program:

```
Run
[5] Ask AI 31s on 16:36:44, 08/07
Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- circulo
5.- salir
Selecciona una opcion:
3
Ingresa la base del triangulo:
1000
Ingresa la altura del triangulo:
1200
El area del triangulo es: 600000.0
Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- circulo
5.- salir
Selecciona una opcion:

```

Triangulo.

The screenshot shows a Swift REPL environment with the following code in `main.swift`:

```
1 class figura{
2     func calculaArea()->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 calculaArea()->Double{
12        return lado*lado
13    }
14 }
15 }
16
17 class rectangulo: figura{
18     var alto: Double
19     var largo: Double
20
21     init(alto: Double, largo: Double){
22         self.alto = alto
23         self.largo = largo
24     }
25     override func calculaArea()->Double{
26         return alto*largo
27     }
28 }
29
30 class triangulo: figura{
31     var base: Double
32     var altura: Double
33 }
```

The console output shows the execution of the program:

```
Run
[5] Ask AI 21s on 16:37:37, 08/07
Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- circulo
5.- salir
Selecciona una opcion:
4
Ingresa el radio del circulo:
2000
El area del circulo es: 12566400.0
Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- circulo
5.- salir
Selecciona una opcion:

```

Circulo.

```
1 class figura{
2     func calculaArea()->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 calculaArea()->Double{
12        return lado*lado
13    }
14 }
15 }
16
17 class rectangulo: figura{
18     var alto: Double
19     var largo: Double
20
21     init(alto: Double, largo: Double){
22         self.alto = alto
23         self.largo = largo
24     }
25     override func calculaArea()->Double{
26         return alto*largo
27     }
28 }
29
30 class triangulo: figura{
31     var base: Double
32     var altura: Double
33 }
```

Run

Menu de opciones: Calcular Area
1.- cuadrado
2.- rectangulo
3.- Triangulo
4.- circulo
5.- salir
Selecciona una opcion:
5
Saliendo del programa...

Salida.

Aplicación 2.

Código.

```
func areaCuadrado(lado: Int)->Int {  
    return lado * lado  
}  
  
func areaRectangulo(largo: Int, ancho: Int)->Int {  
    return largo * ancho  
}  
  
func areaTriangulo(base: Int, altura: Int)->Int {  
    return (base * altura) / 2  
}  
  
func areaCirculo(radio: Int)->Float {  
    return 3.1416 * Float(radio * radio)  
}  
  
var opcion = 0
```

```
repeat{
    print("Menu de Areas")
    print("1. Area del cuadrado")
    print("2. Area del rectangulo")
    print("3. Area del triangulo")
    print("4. Area del circulo")
    print("5. Salir")
    print("Selecciona una opcion (numero):")
    opcion = Int(readLine()!)
    switch opcion{
        case 1:
            print("Ingresa el lado del cuadrado:")
            let lado = Int(readLine()!)
            let resultado = areaCuadrado(lado: lado)
            print("El area del cuadrado es: \(resultado)")
        case 2:
            print("Ingresa el largo del rectangulo:")
            let dato1 = Int(readLine()!)
            print("Ingresa el ancho del rectangulo:")
            let dato2 = Int(readLine()!)
            let resultado2 = areaRectangulo(largo: dato1, ancho: dato2)
            print("El area del rectangulo es: \(resultado2)")
        case 3:
            print("Ingresa la base del triangulo:")
            let dato3 = Int(readLine()!)
            print("Ingresa la altura del triangulo:")
            let dato4 = Int(readLine()!)
            let resultado3 = areaTriangulo(base: dato3, altura: dato4)
```

```

print("El area del triangulo es: \"(resultado3)\")

case 4:

print("Ingresa el radio del circulo:")

let dato5 = Int(readLine()!)

let resultado4 = areaCirculo(radio: dato5)

print("El area del circulo es: \"(resultado4)\")

case 5:

print("Saliendo del programa...")

default:

print("Opcion no valida")

}

} while opcion != 5

```

The screenshot shows a Replit IDE window titled 'Aplicacion 3.1 - Replit'. The main editor displays a Swift file named 'main.swift' with the following code:

```

1 func areaCuadrado(lado: Int)->Int {
2     return lado * lado
3 }
4 func areaRectangulo(largo: Int, ancho: Int)->Int {
5     return largo * ancho
6 }
7 func areaTriangulo(base: Int, altura: Int)->Int {
8     return (base * altura) / 2
9 }
10 func areaCirculo(radio: Int)->Float {
11     return 3.1416 * Float(radio * radio)
12 }
13
14 var opcion = 0
15 repeat{
16     print("Menu de Areas")
17     print("1. Area del cuadrado")
18     print("2. Area del rectangulo")
19     print("3. Area del triangulo")
20     print("4. Area del circulo")
21     print("5. Salir")
22     print("Selecciona una opcion (numero):")
23     opcion = Int(readLine()!)
24     switch opcion{
25     case 1:
26         print("Ingresa el lado del cuadrado:")
27         let lado = Int(readLine()!)
28         let resultado = areaCuadrado(lado: lado)

```

The right sidebar shows the 'Run' console with the following output:

```

Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):

```

The bottom status bar indicates 'Ln 56, Col 20 • Spaces: 2 • History'.

Código.

```
1 func areaCuadrado(lado: Int)->Int {
2     return lado * lado
3 }
4 func areaRectangulo(largo: Int, ancho: Int)->Int {
5     return largo * ancho
6 }
7 func areaTriangulo(base: Int, altura: Int)->Int {
8     return (base * altura) / 2
9 }
10 func areaCirculo(radio: Int)->Float {
11     return 3.1416 * Float(radio * radio)
12 }
13
14 var opcion = 0
15 repeat{
16     print("Menu de Areas")
17     print("1. Area del cuadrado")
18     print("2. Area del rectangulo")
19     print("3. Area del triangulo")
20     print("4. Area del circulo")
21     print("5. Salir")
22     print("Selecciona una opcion (numero):")
23     opcion = Int(readLine()!)
24     switch opcion{
25     case 1:
26         print("Ingresa el lado del cuadrado:")
27         let lado = Int(readLine()!)
28         let resultado = areaCuadrado(lado: lado)
```

Run

Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):
1
Ingresa el lado del cuadrado:
3000
El area del cuadrado es: 9000000
Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):
[]

Cuadrado.

```
1 func areaCuadrado(lado: Int)->Int {
2     return lado * lado
3 }
4 func areaRectangulo(largo: Int, ancho: Int)->Int {
5     return largo * ancho
6 }
7 func areaTriangulo(base: Int, altura: Int)->Int {
8     return (base * altura) / 2
9 }
10 func areaCirculo(radio: Int)->Float {
11     return 3.1416 * Float(radio * radio)
12 }
13
14 var opcion = 0
15 repeat{
16     print("Menu de Areas")
17     print("1. Area del cuadrado")
18     print("2. Area del rectangulo")
19     print("3. Area del triangulo")
20     print("4. Area del circulo")
21     print("5. Salir")
22     print("Selecciona una opcion (numero):")
23     opcion = Int(readLine()!)
24     switch opcion{
25     case 1:
26         print("Ingresa el lado del cuadrado:")
27         let lado = Int(readLine()!)
28         let resultado = areaCuadrado(lado: lado)
```

Run

Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):
2
Ingresa el largo del rectangulo:
4500
Ingresa el ancho del rectangulo:
9500
El area del rectangulo es: 42750000
Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):
[]

Rectángulo.

```
1 func areaCuadrado(lado: Int)->Int {
2     return lado * lado
3 }
4 func areaRectangulo(largo: Int, ancho: Int)->Int {
5     return largo * ancho
6 }
7 func areaTriangulo(base: Int, altura: Int)->Int {
8     return (base * altura) / 2
9 }
10 func areaCirculo(radio: Int)->Float {
11     return 3.1416 * Float(radio * radio)
12 }
13
14 var opcion = 0
15 repeat{
16     print("Menu de Areas")
17     print("1. Area del cuadrado")
18     print("2. Area del rectangulo")
19     print("3. Area del triangulo")
20     print("4. Area del circulo")
21     print("5. Salir")
22     print("Selecciona una opcion (numero):")
23     opcion = Int(readLine()!)
24     switch opcion{
25         case 1:
26             print("Ingresa el lado del cuadrado:")
27             let lado = Int(readLine()!)
28             let resultado = areaCuadrado(lado: lado)
```

Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):
3
Ingresa la base del triangulo:
5000
Ingresa la altura del triangulo:
10000
El area del triangulo es: 25000000
Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):
[]

Triangulo.

```
1 func areaCuadrado(lado: Int)->Int {
2     return lado * lado
3 }
4 func areaRectangulo(largo: Int, ancho: Int)->Int {
5     return largo * ancho
6 }
7 func areaTriangulo(base: Int, altura: Int)->Int {
8     return (base * altura) / 2
9 }
10 func areaCirculo(radio: Int)->Float {
11     return 3.1416 * Float(radio * radio)
12 }
13
14 var opcion = 0
15 repeat{
16     print("Menu de Areas")
17     print("1. Area del cuadrado")
18     print("2. Area del rectangulo")
19     print("3. Area del triangulo")
20     print("4. Area del circulo")
21     print("5. Salir")
22     print("Selecciona una opcion (numero):")
23     opcion = Int(readLine()!)
24     switch opcion{
25         case 1:
26             print("Ingresa el lado del cuadrado:")
27             let lado = Int(readLine()!)
28             let resultado = areaCuadrado(lado: lado)
```

Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):
4
Ingresa el radio del circulo:
25650
El area del circulo es: 2.0669293e+09
Menu de Areas
1. Area del cuadrado
2. Area del rectangulo
3. Area del triangulo
4. Area del circulo
5. Salir
Selecciona una opcion (numero):
[]

Circulo.


```
1 func areaCuadrado(lado: Int)->Int {
2     return lado * lado
3 }
4 func areaRectangulo(largo: Int, ancho: Int)->Int {
5     return largo * ancho
6 }
7 func areaTriangulo(base: Int, altura: Int)->Int {
8     return (base * altura) / 2
9 }
10 func areaCirculo(radio: Int)->Float {
11     return 3.1416 * Float(radio * radio)
12 }
13
14 var opcion = 0
15 repeat{
16     print("Menu de Areas")
17     print("1. Area del cuadrado")
18     print("2. Area del rectangulo")
19     print("3. Area del triangulo")
20     print("4. Area del circulo")
21     print("5. Salir")
22     print("Selecciona una opcion (numero):")
23     opcion = Int(readLine()!)
24     switch opcion{
25     case 1:
26         print("Ingresa el lado del cuadrado:")
27         let lado = Int(readLine()!)
28         let resultado = areaCuadrado(lado: lado)
```

Salida.

Conclusión.

Como desarrollador, cada proyecto es un viaje. Y este, mi amigo, ha sido un paseo geométrico fascinante. Al final del camino, aquí están mis reflexiones:

1. Sencillez y Agilidad con Replit:
 - Replit me ha demostrado que no necesito complicaciones para crear algo útil. Su interfaz minimalista y su entorno preconfigurado me han permitido concentrarme en la lógica y la creatividad, sin perder tiempo en configuraciones tediosas.
2. La Magia de las Matemáticas:
 - Calcular áreas puede parecer trivial, pero detrás de esas fórmulas hay belleza. El triángulo que se despliega en la pantalla, el círculo que gira como un péndulo: son como pequeños secretos matemáticos revelados.
3. Compartir y Colaborar:
 - Replit me ha dado la posibilidad de compartir mi creación con otros. ¿Quién sabe? Tal vez algún estudiante de geometría o un apasionado de las figuras encuentre utilidad en esta calculadora. La colaboración es la salsa secreta de la programación.
4. El Futuro es Luminoso:
 - Este proyecto modesto es solo el comienzo. Replit me ha mostrado que puedo construir cosas asombrosas sin barreras. La próxima vez, quizás sea una aplicación más grande, un juego o incluso un sistema de inteligencia artificial.

Referencias. Video de la Tutoría 3.