



PRÁCTICA 02: Fundamentos y sintaxis de lenguaje.

Facultad de Ingeniería



Laboratorios Salas A y B

Profesor: Yovanni Alvarez Ulloa

Asignatura: Modelos de Programación Orientada a Objetos

Grupo: 07

No. de práctica(s): 03

Nombre del alumno: Dolores Zeferino Yobel

No. de lista: 05

Semestre: 2025-1

Fecha de entrega: 04 de septiembre, 2024

Observaciones:

CALIFICACIÓN: _____

Reporte de Práctica

Dolores Zeferino Yobel

Septiembre 04, 2024

1 INTRODUCCIÓN

En el presente reporte de práctica se abordará el tema de fundamentos y sintaxis de lenguaje. Donde se creará código en lenguaje Swift desde cero, implementando la sintaxis básica propia del lenguaje, todo esto para fines prácticos y para el objetivo siguiente.

Objetivo: Implementar programas utilizando:

- Diversos tipos de datos.
- Expresiones.

2 DESARROLLO

2.1 Actividad 1

Se nos solicita crear una clase de nombre **Promedio** el cual debe contener un método llamado **promedio** que a su vez debe calcular el promedio de 10 números del rango del 1-10 los cuales deben ser proporcionados directamente desde el teclado. Para implementar dicha clase se hizo uso de los operadores básicos del lenguaje y para la entrada de datos directamente desde la terminal se usó el método **readLine**. A continuación se muestra el código:

```
1
2 import Foundation
3
4 class Promedio{
5     var cali1 = 0.0
6     var cali2 = 0.0
7     var cali3 = 0.0
8     var cali4 = 0.0
9     var cali5 = 0.0
10    var cali6 = 0.0
11    var cali7 = 0.0
12    var cali8 = 0.0
13    var cali9 = 0.0
14    var cali10 = 0.0
15    var suma = 0.0
16    var promedioFinal = 0.0
17
18    func ingresaCalificacion(){
19        print("ingrese la calificación 1")
20        cali1 = Double(readLine()!)
21        print("ingrese la calificación 2 \n")
22        cali2 = Double(readLine()!)
23        print("ingrese la calificación 3\n")
24        cali3 = Double(readLine()!)
25        print("ingrese la calificación 4\n")
26        cali4 = Double(readLine()!)
27        print("ingrese la calificación 5\n")
28        cali5 = Double(readLine()!)
29        print("ingrese la calificación 6\n")
30        cali6 = Double(readLine()!)
```

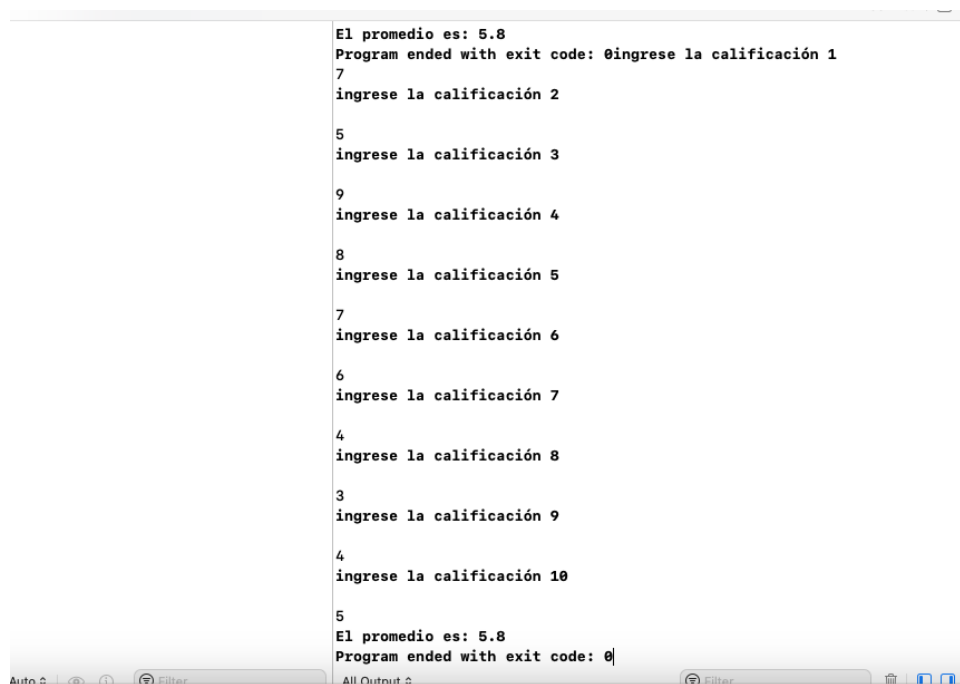
```

31     print("ingrese la calificación 7\n")
32     cali7 = Double(readLine()!)
33     print("ingrese la calificación 8\n")
34     cali8 = Double(readLine()!)
35     print("ingrese la calificación 9\n")
36     cali9 = Double(readLine()!)
37     print("ingrese la calificación 10 \n")
38     cali10 = Double(readLine()!)
39 }
40
41 func calcularPromedio(){
42     suma = cali1+cali2+cali3+cali4+cali5+cali6+cali7+cali8+cali9+cali10
43     promedioFinal = suma/10
44     print("El promedio es: \(promedioFinal)")
45 }
46 }
47
48 var promedio1 = Promedio()
49 promedio1.ingresaCalificacion()
50 promedio1.calcularPromedio()

```

Listing 1: Código en Swift

Posteriormente al compilar obtenemos el siguiente resultado en la terminal:



```

El promedio es: 5.8
Program ended with exit code: 0
ingrese la calificación 1
7
ingrese la calificación 2
5
ingrese la calificación 3
9
ingrese la calificación 4
8
ingrese la calificación 5
7
ingrese la calificación 6
6
ingrese la calificación 7
4
ingrese la calificación 8
3
ingrese la calificación 9
4
ingrese la calificación 10
5
El promedio es: 5.8
Program ended with exit code: 0

```

Figure 1: Promedio

2.2 Actividad 2

Se pide realizar una clase llamada **Carro** el cual debe contener los métodos de: **calcularVelocidad** y **calcularAceleración**. Donde los datos por defecto deben ser los siguientes:

- $V_i = 18.5 \text{ m/s}$
- $V_f = 46.1 \text{ m/s}$
- $d = 105.3 \text{ m}$
- $t = 2.74 \text{ s}$

Implementado lo anterior da resultado al siguiente código:

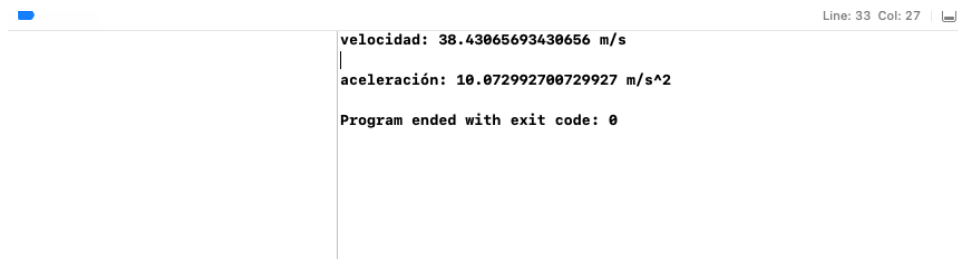
```

1 import Foundation
2
3
4 class Car{
5     var Vi=18.5
6     var Vf=46.1
7     var d=105.3
8     var t=2.74
9     var speed = 0.0
10    var acceleration = 0.0
11
12    func calculeSpeed(){
13        speed = d/t
14        print("velocidad: \(speed) m/s \n")
15    }
16
17    func calculeAcceleration(){
18        acceleration = (Vf-Vi)/t
19        print("aceleración: \(acceleration) m/s^2 \n")
20    }
21 }
22
23
24 var car1 = Car()
25 car1.calculeSpeed()
26 car1.calculeAcceleration()

```

Listing 2: Código en Swift

Al compilar dicho código se obtienen los siguientes resultados en la terminal:



```

Line: 33 Col: 27 |
velocidad: 38.43065693430656 m/s
|
aceleración: 10.072992700729927 m/s^2
Program ended with exit code: 0

```

Figure 2: Velocidad y Aceleración del Auto

2.3 Actividad 3

Ahora se nos solicita que diseñemos una clase llamada `Inventario` el cual se le deberán agregar desde un principio los productos siguientes cuyos valores están también determinados:

- panDulce = 35
- cheetos = 13
- dulces = 123
- refrescos = 16

De esta manera obtenemos el código siguiente:

```

1 import Foundation
2
3 class Inventario{
4     var panDulce=35
5     var cheetos=13
6     var dulces=123

```

```

7  var refrescos=16
8  var productoNuevo = " "
9  var cantidadNuevoProducto = 0
10 var productosTotales = 0
11 }

```

Listing 3: Código en Swift

Posteriormente se pide modificar lo anterior e implementar los siguientes puntos:

- **masProductos:** donde se agrega más stock a un producto del inventario.
- **menosProductos:** en el que se seleccionará un producto y se le restará la cantidad que desee el usuario.
- **agregarProductos:** donde se pedirá al usuario que agregue un producto nuevo, con su respectivo stock.
- **totalProductos:** el cual deberá mostrar el total de todos los productos en el inventario.
- **verProductos:** del mismo modo se debe mostrar la cantidad de cada producto y su stock que hay en el inventario.

Para hacer la implementación de lo anterior se hizo uso de los operadores básicos vistos en la sintaxis del lenguaje, así como el método de entrada de cadenas directamente desde la línea de comandos de la terminal que es `readLine`. Así, habiendo implementado los puntos anteriores, se obtuvo el código que se muestra a continuación:

```

1  import Foundation
2
3
4  class Inventario{
5      var panDulce=35
6      var cheetos=13
7      var dulces=123
8      var refrescos=16
9      var productoNuevo = " "
10     var cantidadNuevoProducto = 0
11     var productosTotales = 0
12
13     func masProducto(){
14         print("Cuanto pan dulce quieres agregar \n")
15         let masPanDulce = Int(readLine()!)
16         print("Cuanto cheetos quieres agregar \n")
17         let masCheetos = Int(readLine()!)
18         print("Cuanto dulces quieres agregar \n")
19         let masDulces = Int(readLine()!)
20         print("Cuanto refrescos quieres agregar \n")
21         let masRefrescos = Int(readLine()!)
22
23         panDulce = panDulce + masPanDulce
24         cheetos = cheetos + masCheetos
25         dulces = dulces + masDulces
26         refrescos = refrescos + masRefrescos
27     }
28
29     func menosProducto(){
30         print("Cuanto pan dulce quieres quitar \n")
31         let menosPanDulce = Int(readLine()!)
32         print("Cuanto cheetos quieres quitar \n")
33         let menosCheetos = Int(readLine()!)
34         print("Cuanto dulces quieres quitar \n")
35         let menosDulces = Int(readLine()!)
36         print("Cuanto refrescos quieres quitar \n")
37         let menosRefrescos = Int(readLine()!)
38     }

```

```

39     panDulce = panDulce - menosPanDulce
40     cheetos = cheetos - menosCheetos
41     dulces = dulces - menosDulces
42     refrescos = refrescos - menosRefrescos
43 }
44
45 func nuevoProducto(){
46     print("Agrega un nuevo producto\n")
47     productoNuevo = readLine()!
48     print("Cual es la cantidad de ese producto\n")
49     cantidadNuevoProducto = Int(readLine()!)!
50 }
51
52 func totalProductos(){
53     productosTotales = panDulce+cheetos+refrescos+dulces+cantidadNuevoProducto
54     print("En total se tienen \(productosTotales) productos")
55 }
56
57 func verProductos(){
58     print("Vista general del inventario:\n")
59     print("Pan dulce: \(panDulce) piezas")
60     print("Cheetos: \(cheetos) piezas")
61     print("Dulces: \(dulces) piezas")
62     print("Refrescos: \(refrescos) piezas")
63     print("\(productoNuevo): \(cantidadNuevoProducto) piezas")
64 }
65 }
66
67 var miInventario = Inventario()
68 miInventario.masProducto()
69 miInventario.menosProducto()
70 miInventario.nuevoProducto()
71 miInventario.totalProductos()
72 miInventario.verProductos()

```

Listing 4: Código en Swift

Al compilar el código anterior se obtiene el siguiente resultado en la terminal:

```

Agrega un nuevo producto

Leche
Cual es la cantidad de ese producto

45
En total se tienen 262 productos
Vista general del inventario:

Pan dulce: 43 piezas
Cheetos: 22 piezas
Dulces: 124 piezas
Refrescos: 28 piezas
Leche: 45 piezas
|

```

Figure 3: Inventario

Donde se muestra el resultado de todos los métodos implementados anteriormente funcionando en conjunto.

2.4 Actividad 4

Se nos solicita implementar una clase `Calculadora` el cual debe calcular las siguientes operaciones:

- suma
- resta
- multiplicación
- división
- raizCuadrada

Para hacer la implementación anterior se hizo uso de los operadores básicos vistos en la sintaxis del lenguaje, para la entrada de datos que el usuario nos proporciona directamente desde la terminal igualmente se usa `readLine`. El código obtenido después de hacer dicha implementación es el siguiente:

```
1
2 import Foundation
3
4 class Calculator{
5     var number1 = 0.0
6     var number2 = 0.0
7     var sum = 0.0
8     var subtraction = 0.0
9     var product = 0.0
10    var quotient = 0.0
11    var sqrtNumber1 = 0.0
12    var sqrtNumber2 = 0.0
13
14    func inputValues(){
15        print("Ingresa el valor para el primer numero")
16        number1 = Double(readLine()!)!
17        print("ingresa el valor para el segundo numero")
18        number2 = Double(readLine()!)!
19    }
20
21    func additionTwoNumbers(){
22        sum = number1+number2
23        print("La suma de los dos numeros es \(sum) \n")
24    }
25    func subtractionTwoNumbers(){
26        subtraction = number1-number2
27        print("La resta de los dos numeros es \(subtraction) \n")
28    }
29
30    func productTwoNumbers(){
31        product = number1*number2
32        print("El producto de los numeros es \(product) \n")
33    }
34
35    func divisionTwoNumbers(){
36        quotient = number1/number2
37        print("El cociente \(number1) entre \(number2) es \(quotient) \n")
38    }
39
40    func sqrtOfNumbers(){
41        sqrtNumber1 = sqrt(number1)
42        sqrtNumber2 = sqrt(number2)
43        print("La raiz cuadrada de \(number1) = \(sqrtNumber1) \n")
44        print("La raiz cuadrada de \(number2) = \(sqrtNumber2) \n")
45    }
46 }
47
48 var myFirstCalculation = Calculator()
49 myFirstCalculation.inputValues()
50 myFirstCalculation.additionTwoNumbers()
51 myFirstCalculation.subtractionTwoNumbers()
```

```

52 myFirstCalculation.productTwoNumbers()
53 myFirstCalculation.divisionTwoNumbers()
54 myFirstCalculation.sqrtOfNumbers()

```

Listing 5: Código en Swift

El cual al compilarlo genera el siguiente resultado en la salida de la terminal:

Output

```

swift /tmp/vkZ7Ac4Uyn.swift
Ingresa el valor para el primer numero
10
ingresa el valor para el segundo numero
4
La suma de los dos numeros es 14.0

La resta de los dos numeros es 6.0

El producto de los numeros es 40.0

El cociente 10.0 entre 4.0 es 2.5

La raiz cuadrada de 10.0 = 3.1622776601683795

La raiz cuadrada de 4.0 = 2.0

```

Figure 4: Operaciones de la Calculadora

2.5 Actividad 5

En esta actividad se solicita que se cree una clase de nombre **Figuras**, el cual debe calcular el área y perímetro de las siguientes figuras:

- calculeCirculo
- calculeCuadrado
- calculeTriangulo
- calculeCubo

Para dicha clase se crearon distintos métodos para cada una de las figuras, donde en cada método se hicieron uso de operadores básicos propios del lenguaje y para leer los datos proporcionados por el usuario al igual que en los casos anteriores se usó `readLine`. El resultado de esta implementación se muestra en el código siguiente:

```

1  import Foundation
2
3
4  class Figuras{
5      var radioCircle = 0.0
6      var side = 0.0
7      var side2 = 0.0
8      var side3 = 0.0
9      var height = 0.0
10     var pi = 3.141592
11

```



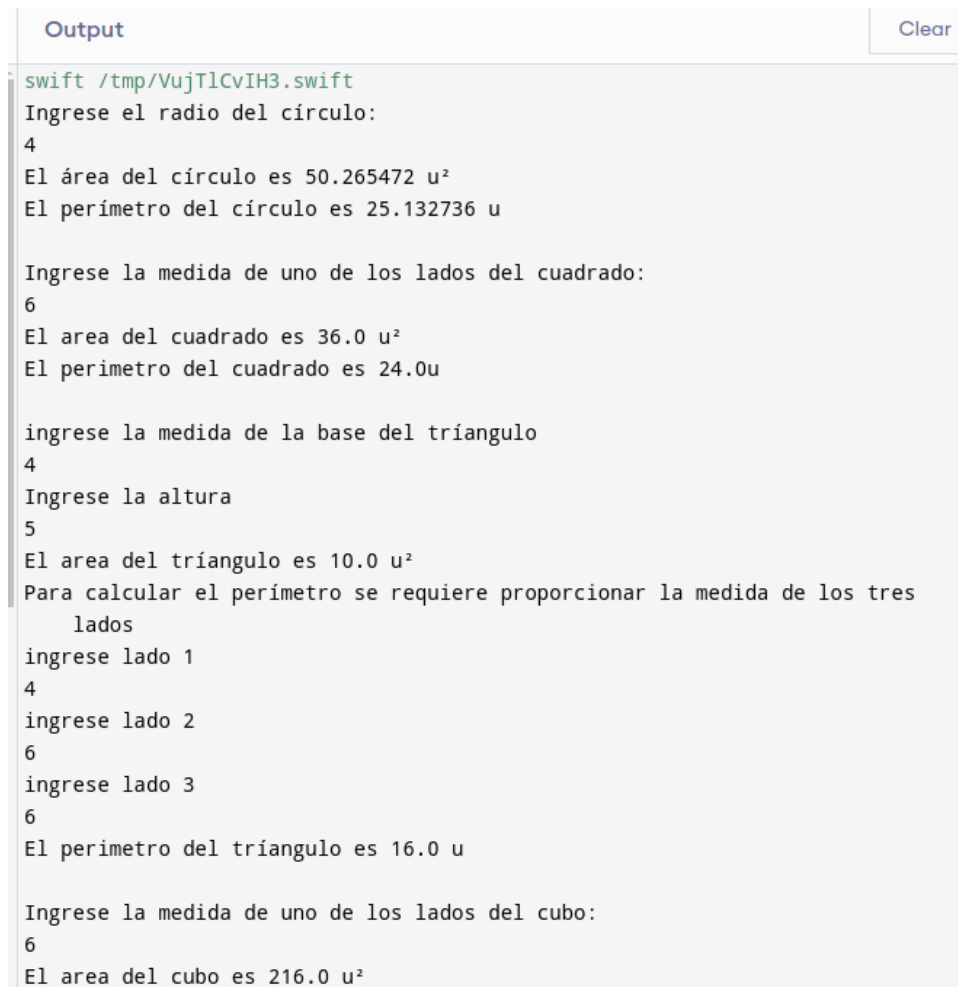
```

12 func calculateCircle(){
13     print("Ingrese el radio del círculo:")
14     radioCircle = Double(readLine()!!)
15     let areaCircle = (pow(radioCircle,2))*pi
16     let perimeterCircle = 2*pi*radioCircle
17     print("El área del círculo es \(areaCircle) u ")
18     print("El perímetro del círculo es \(perimeterCircle) u \n")
19 }
20
21 func calculateTriangle(){
22     print("ingrese la medida de la base del triángulo")
23     side = Double(readLine()!!)
24     print("Ingrese la altura")
25     height = Double(readLine()!!)
26     let areaTriangle = (side*height)/2
27     print("El area del triángulo es \(areaTriangle) u ")
28     print("Para calcular el perímetro se requiere proporcionar la medida de los tres
29         lados")
29     print("ingrese lado 1")
30     side = Double(readLine()!!)
31     print("ingrese lado 2")
32     side2 = Double(readLine()!!)
33     print("ingrese lado 3")
34     side3 = Double(readLine()!!)
35     let perimeterTriangle = side+side2+side3
36     print("El perímetro del triángulo es \(perimeterTriangle) u \n")
37 }
38
39 func calculateSquare(){
40     print("Ingrese la medida de uno de los lados del cuadrado:")
41     side = Double(readLine()!!)
42     let areaSquare = pow(side,2)
43     let perimeterSquare = side*4
44     print("El area del cuadrado es \(areaSquare) u ")
45     print("El perímetro del cuadrado es \(perimeterSquare)u \n")
46 }
47
48 func calculateCube(){
49     print("Ingrese la medida de uno de los lados del cubo:")
50     side = Double(readLine()!!)
51     let areaCube = (side*side)*6
52     let perimeterCube = side*12
53     print("El area del cubo es \(areaCube) u ")
54     print("El perímetro del cubo es \(perimeterCube) u")
55 }
56 }
57
58 var myFigures = Figures()
59 myFigures.calculateCircle()
60 myFigures.calculateSquare()
61 myFigures.calculateTriangle()
62 myFigures.calculateCube()

```

Listing 6: Código en Swift

Posteriormente al compilar dicho código, se muestra el siguiente resultado en la terminal:



```
Output Clear
swift /tmp/VujTlCvIH3.swift
Ingrese el radio del círculo:
4
El área del círculo es 50.265472 u²
El perímetro del círculo es 25.132736 u

Ingrese la medida de uno de los lados del cuadrado:
6
El area del cuadrado es 36.0 u²
El perímetro del cuadrado es 24.0u

ingrese la medida de la base del triangulo
4
Ingrese la altura
5
El area del triangulo es 10.0 u²
Para calcular el perímetro se requiere proporcionar la medida de los tres
    lados
ingrese lado 1
4
ingrese lado 2
6
ingrese lado 3
6
El perímetro del triángulo es 16.0 u

Ingrese la medida de uno de los lados del cubo:
6
El area del cubo es 216.0 u²
```

Figure 5: Áreas y Perímetros

3 CONCLUSIÓN

Como pudimos ver durante el desarrollo de la práctica, la sintaxis del lenguaje en este caso Swift es muy importante ya que en mayor parte, esta es la base de los métodos de una clase para llevar a cabo diferentes tareas que requiera nuestro programa, o del problema que se esté queriendo modelar. Entonces, a modo de resumen, la sintaxis así como la implementación de diferentes tipos de datos son muy importantes para nuestra materia ya que se utilizará muy a menudo en los programas de prácticas posteriores.

4 Repositorio GitHub

Para una mejor visualización de los códigos visitar el siguiente repositorio: https://github.com/yobel3210/practica_3.git