universidad cenfotec_
La U de la informática

Principios de Programación.

Introducción a Rutinas: Funciones.

Objetivos

universidad cenfotec
La U de la informática

- Comprender el concepto de **funciones**.
- Aplicar el concepto de abstracción procedimental.
- Crear soluciones que incluyan el diseño y programación de funciones en el lenguaje de programación Python.

Repaso: Rutinas (conceptos importantes)

universidad cenfotec
La U de la informática

Hay que recordar que existen dos conceptos fundamentales a la hora de hablar de rutinas, el primero de ellos son los **valores de retorno**.

Valor de retorno.

- Un valor de retorno es el valor de **salida** que produce una *función* después de ejecutar un proceso. ***Ejemplo***: el área de un círculo, el salario de una persona, etc.
- Esta salida **no** es una impresión en pantalla, sino que está representada por una variable, que tendrá tipo de dato, nombre y valor.
- El **valor** de esta variable es adquirido durante la ejecución de la *función* y es justamente el valor de retorno.

Repaso: Rutinas (conceptos importantes)

universidad cenfotec
La U de la informática

El segundo concepto fundamental en relación a la construcción de rutinas son los **parámetros**.

Parámetro.

- Un parámetro es un **valor de entrada** a la rutina, necesario para que ésta pueda lograr su objetivo.
- Un parámetro está representado por una **variable**, que tendrá tipo de dato, nombre y valor.
- El valor del parámetro es adquirido **antes** de la rutina, se usa dentro de la rutina y puede ser modificado durante la ejecución de la rutina.

Repaso: Rutinas (conceptos importantes)

universidad cenfotec
La U de la informática

Las rutinas pueden ser clasificadas por su objetivo dentro del programa de la siguiente forma:

RUTINAS

- **Funciones**: trozo o bloque de código que retorna un valor.
  - Ejemplos: Calcular el área de un círculo, el salario de un empleado, decir si una fecha es válida o no, etc.
- **Procedimiento**: trozo o bloque de código que no retorna ningún valor.
  - Ejemplos: Imprimir un menú de opciones, imprimir los datos de salida de un programa con formato especial, etc.

Repaso: Rutinas (clasificación de rutinas por su objetivo)

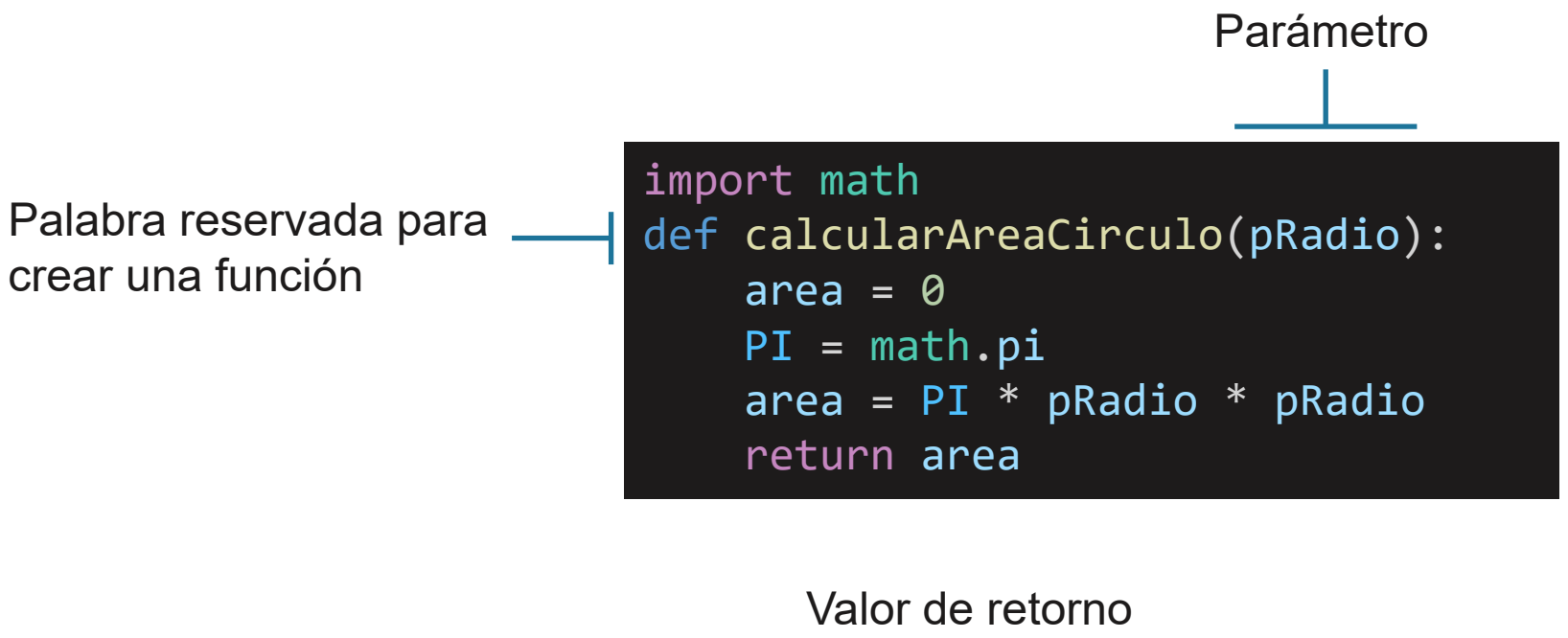
universidad cenfotec
La U de la informática

Funciones

- Una función es una rutina diseñada para realizar una tarea muy específica y **retornar un valor**.
- El valor de retorno de una función puede ser **de cualquier tipo de dato**: *boolean*, *int*, *float*, *String*, etc.
- Una función puede recibir cero o más **parámetros**.
- Ejemplos: Calcular el mayor de dos números y retornar el valor del número mayor, verificar si una fecha es válida y retornar un valor booleano dependiendo del resultado, verificar si un año dado es bisiesto y retornar un valor booleano dependiendo del resultado, etc.

Repaso: Rutinas (clasificación de rutinas por su objetivo)

Un ejemplo de una **función** en Python que calcula el área de un círculo es:



Funciones: (otros conceptos importantes)

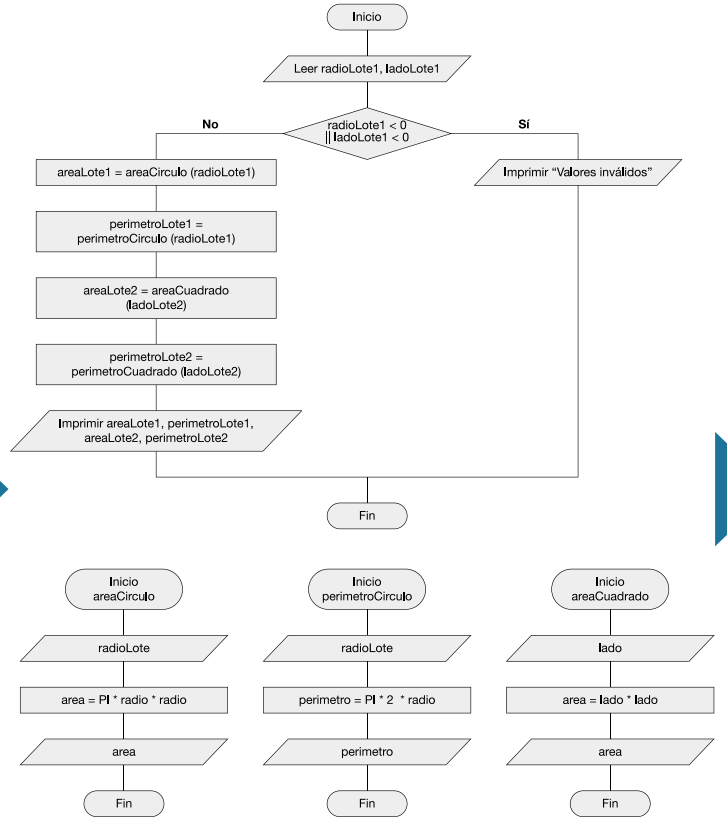
universidad cenfotec
La U de la informática

1. Todas las funciones tienen un **nombre** que las identifica. Se recomienda que dicho nombre debe siga la notación *camelCase*.
2. Todas las funciones deben iniciar con la palabra reservada **def**.
3. Toda función debe indicar el **tipo de dato** del valor de retorno.
4. Toda función siempre tiene un **valor de retorno**, que contiene el valor del dato para el que fue creada la función. Además, una función solo puede retornar un único valor de retorno.
5. Toda función puede tener cero o más **parámetros**.
6. Toda función tiene **un inicio y un fin**, que en Python está indicado por los **:** y el fin de la indentación.
7. Todas las instrucciones indicadas dentro de la función son el **cuerpo de la función**.

Fases de la abstracción procedimental

universidad cenfotec
La U de la informática

ENTRADAS			
Descripción	Notación		Ejemplo
	Nombre	Tipo de dato	
Radio del lote circular	radioLote1	float	5.5
Lado del lote cuadrado	ladoLote2	float	5.8
SALIDAS			
Descripción	Notación		Ejemplo
	Nombre	Tipo de dato	
Valor del área del Lote 1	areaLote1	float	143.5
Valor del perímetro del Lote 1	perimetroLote1	float	122.5
Valor del área del Lote 2	areaLote2	float	123.5
Valor del perímetro del Lote 2	perimetroLote2	float	172.5




```python
import math
radioLote1 = 0.0
ladoLote2 = 0.0
areaLote1 = 0.0
perimetroLote1 = 0.0
areaLote2 = 0.0
perimetroLote2 = 0.0
PI = math.pi

radioLote1 = float(input('Por favor escriba el radio del Lote 1:'))
ladoLote2 = float(input('Por favor escriba el lado del Lote 2: '))

#Definición de las rutinas

def calcularAreaCirculo(pRadio):
    area = 0
    area = PI * pRadio * pRadio
    return area

def calcularPerimetroCirculo(pRadio):
    perimetro = 0
    perimetro = 2 * PI * pRadio
    return perimetro


def calcularAreaCuadrado(pLado):
    area = 0
    area = pLado * pLado
    return area

def calcularPerimetroCuadrado(pLado):
    perimetro = 0
    perimetro = pLado * 4
    return perimetro

# Flujo principal

if (radioLote1 < 0 or ladoLote2 < 0):
    print('Por favor ingrese valores positivos.')
else:
    areaLote1 = calcularAreaCirculo(radioLote1)
    perimetroLote1 = calcularPerimetroCirculo(radioLote1)
    print(f'El área del Lote 1 es: {areaLote1}')
    print(f'El perímetro del Lote 1 es: {perimetroLote1}')

    areaLote2 = calcularAreaCuadrado(ladoLote2)
    perimetroLote2 = calcularPerimetroCuadrado(ladoLote2)
    print(f'El área del Lote 2 es: {areaLote2}' )
    print(f'El perímetro del Lote 2 es: {perimetroLote2}' )
```

Análisis **Diseño** **Implementación**

Ejemplo (aplicando abstracción procedimental)

universidad cenfotec
La U de la informática

Se quiere encontrar el área y el perímetro de unos lotes de figuras geométricas, para esto se calcula el área de cada figura, se calcula el perímetro de cada figura y luego se suman para tener el área y el perímetro de cada uno de los lotes.

Para hallar el área y el perímetro de cada uno de los lotes se usan las siguientes fórmulas:

- Área de un círculo = ***π * radio²***.
- Perímetro de un círculo (circunferencia) = ***2 * π * radio***.
- Área de un triángulo = ***(base * altura) / 2***.
- Perímetro de un triángulo = ***lado1 + lado2 + lado3***.
- Área de un cuadrado = ***lado1 * lado2***.
- Perímetro de un cuadrado = ***lado1 + lado2 + lado3 + lado4***.

Dividiendo el problema original en problemas más pequeños que sepa resolver, y luego uniendo estas soluciones, podrá generar la solución al problema original.

Ejemplo (aplicando abstracción procedimental)

universidad cenfotec
La U de la informática

1. **Análisis.**

Se inicia con uno de los problemas pequeños: calcular el área de un círculo.

Para esto, es necesario definir la tabla de entradas o parámetros de la función y la tabla de salida o valor de retorno de la función.

ENTRADAS			
Descripción	**Notación**		**Ejemplo**
	Nombre	**Tipo de dato**	
Radio del círculo (parámetro)	pRadio	float	5.5

SALIDAS			
Descripción	**Notación**		**Ejemplo**
	Nombre	**Tipo de dato**	
Valor del área de un círculo (valor de retorno)	area	float	122.5

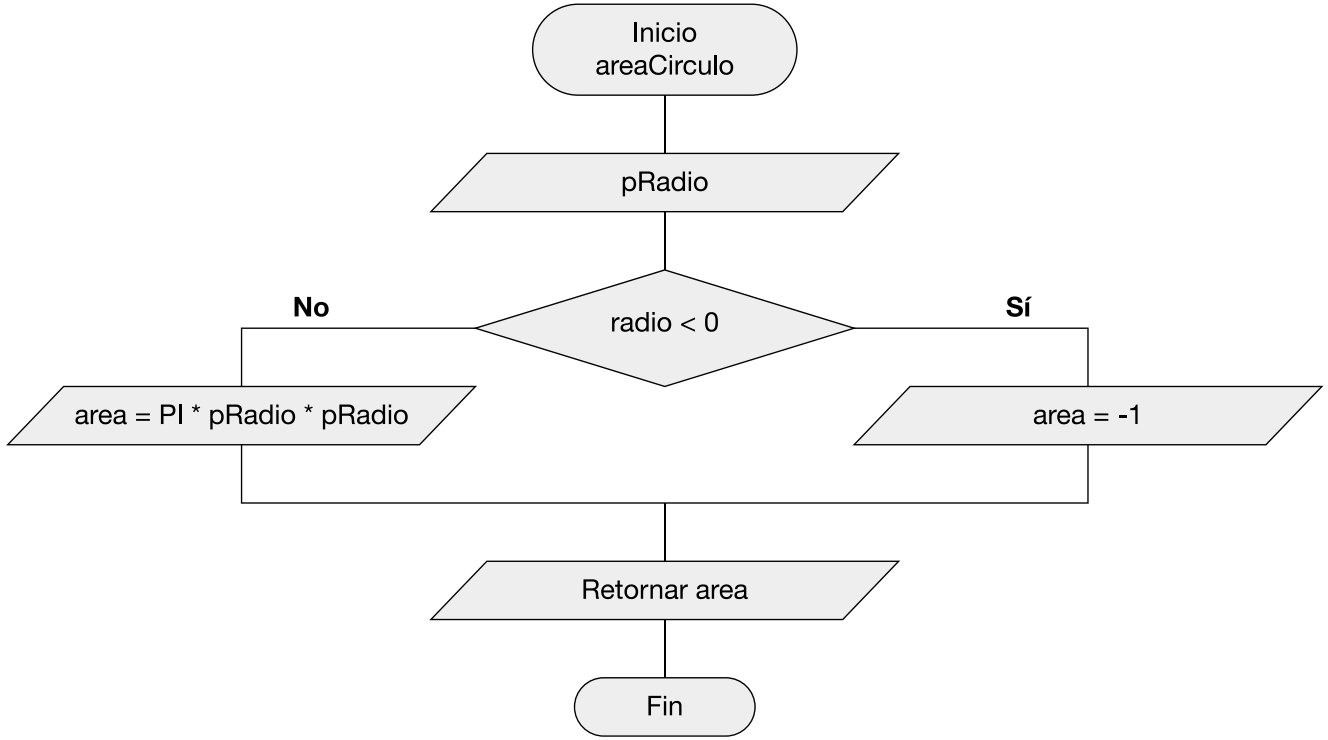
Ejemplo (aplicando abstracción procedimental)

universidad cenfotec
La U de la informática

2. Diseño.

Luego, se procede a diseñar el algoritmo de la función que calcula el área de un círculo.

Para esto se utiliza la notación de los diagramas de flujo. En lugar de leer un dato de entrada, el dato de entrada es el parámetro de la función. En lugar de *imprimir* un dato de salida, se retorna el valor de retorno.



Ejemplo (aplicando abstracción procedimental)

universidad cenfotec
La U de la informática

3. Implementación.

Por último, se implementa el algoritmo de la función en el lenguaje de programación Python, teniendo en cuenta todos los conceptos importantes de las funciones.

No se necesita recibir como parámetro a la constante PI, porque es recomendable declararla de manera globa, para que pueda ser usada por el código principal y sus funciones.

```
import math
PI = math.pi
def calcularAreaCirculo(pRadio):
    area = 0
    if(pRadio < 0):
        area = -1
    else:
        area = PI * pRadio * pRadio
    return área
```

Ejemplo (aplicando abstracción procedimental)

universidad cenfotec
La U de la informática

1. Análisis.

Continuando con el análisis del problema original, para el cual se necesita saber el área de un lote de forma circular, dado el radio del lote.

ENTRADAS			
Descripción	**Notación**		**Ejemplo**
	Nombre	**Tipo de dato**	
Radio de un lote circular.	radioLote	float	5.5
SALIDAS			
Descripción	**Notación**		**Ejemplo**
	Nombre	**Tipo de dato**	
Área de un lote.	areaLote	float	122.5

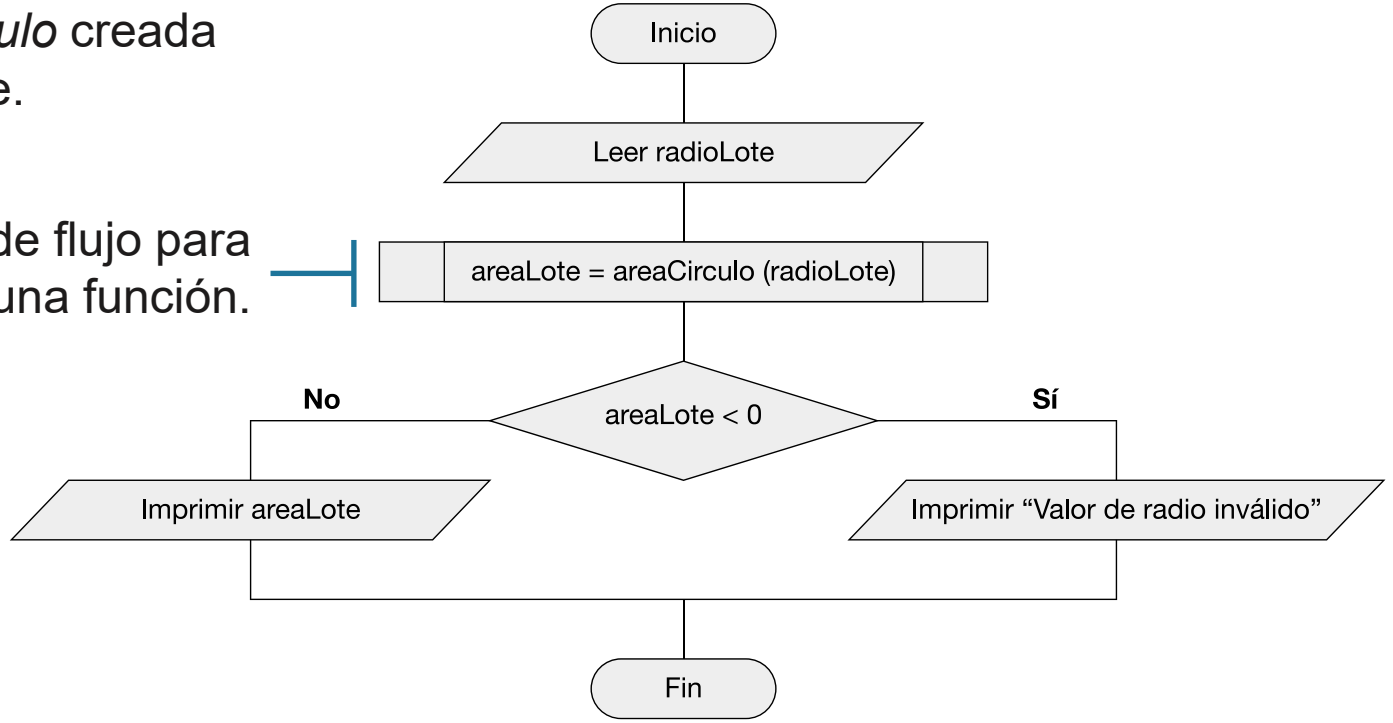
Ejemplo (aplicando abstracción procedimental)

universidad cenfotec
La U de la informática

2. Diseño.

Luego se diseña el algoritmo del programa principal o *main*, el cual ejecutará la función *areaCirculo* creada anteriormente para calcular el área del lote.

Símbolo usado en el diagrama de flujo para indicar el uso o llamado de una función.



Ejemplo (aplicando abstracción procedimental)

universidad cenfotec
La U de la informática

3. Implementación.

La implementación completa del programa principal y la función, se muestra a continuación.

** En Python se requiere que las rutinas hayan sido definidas antes de invocarlas.*

```
import math
PI = math.pi
radioLote = 0.0
areaLote = 0.0

def calcularAreaCirculo(pRadio):
    area = 0
    area = PI * pRadio * pRadio
    return area

radioLote = float(input('Por favor escriba el radio del Lote: '
))
areaLote = calcularAreaCirculo(radioLote)

if(areaLote < 0):
    print('Error, valor del radio del lote inválido.')
else:
    print(f'El área del lote es: {areaLote}')
```

Ejemplo (aplicando abstracción procedimental)

universidad cenfotec
La U de la informática

3. Implementación.

Cuando se define una función sus parámetros son llamados **parámetros formales**. Cuando se llama una función sus parámetros son llamados **parámetros actuales**.

Las variables declaradas en una función son **locales**. Su alcance es la función donde fueron declaradas.

```
import math
PI = math.pi
radioLote = 0.0
areaLote = 0.0

def calcularAreaCirculo(pRadio):
    area = 0
    area = PI * pRadio * pRadio
    return area

radioLote = float(input('Por favor escriba el radio del Lote: '
))
areaLote = calcularAreaCirculo(radioLote)

if(areaLote < 0):
    print('Error, valor del radio del lote inválido.')
else:
    print(f'El área del lote es: {areaLote}')
```

Parámetro formal

Variable local a la función

Parámetro actual


universidad
cenfotec_
La U de la informática