

▼ Clase 3: Módulos y Programas Interactivos

(Capítulo 4 del Apunte)

Variables constantes

La variable `pi` se repite varias veces en el mismo programa y no cambia nunca. Las variables que son constantes se pueden definir como tales afuera de la función. (modificamos el código de arriba para comprobarlo.)

Dar nombres relevantes a las constantes que serán utilizadas frecuentemente en su programa, utilizando estos nombres en lugar de hacer referencia a su valor

Esto permite que el código sea más entendible y si hay que modificar el valor de la constante se hace una sola vez.

▼ Módulos

La programación usando módulos (programación modular) es una técnica de diseño que separa las funciones de un programa en módulos.

- Un módulo tiene una finalidad única y contienen todo lo necesario para llevar a cabo esa funcionalidad (código fuente, variables).
- Conceptualmente un módulo representa separación de intereses.
- Mejoran la mantenibilidad de un software.

▼ Ejemplo:

Módulo [triangulo.py](#) (debemos crearlo en un archivo aparte con el nombre del modulo+".py")

Luego, para utilizar cualquiera de las funciones del módulo desde otros programas debemos decir:

```
import triangulo
```

O

```
from triangulo import *
```

esto importa todas las funciones del módulo, o alternatively:

```
from triangulo import nombre_funcion
```

esto importa solo la función especificada por `nombre_funcion`, que puede ser cualquier funcion definida en `triangulo.py`

```
from triangulo import *  
perimetro(2,3,2)
```


 7

▼ Módulos que vienen con python


`math.py`

```
import math  
# help(math)  
# help muestra la documentacion, esta comentado para imprimir el pdf
```

`math.pi`

 3.141592653589793

`math.pow(10,10)`

 10000000000.0

`math.sqrt(25)`

 5.0

`math.trunc(math.pi)`

 3

`help(math)` # ejecutar y ver lo que pasa (muestra el manual con funciones de math)


▼ Ejemplo de uso:

```
import math
```

```
grados = 90
radianes = math.pi*grados/180
seno = math.sin(radianes)
seno
```

 1.0


```
coseno = math.cos(radianes)
coseno
```

 6.123233995736766e-17

▼ random.py

```
import random
```

```
random.random() #float al azar entre 0 y 1
```

 0.86791076236485

```
random.randint(2,7) #int al azar entre a y b
```

 4

▼ Otras funciones que vienen incluidas en python (no se importan con módulos vienen por defecto)

```
abs(-1)
```

 1

```
max(3,4,2)
```


 4

```
min(2,0)
```

 0

▼ Ejemplo usando random y funciones incluidas en python

```
#mayor y menor de 3 enteros al azar entre 1 y 100
import random
a=random.randint(1,100)
b=random.randint(1,100)
c=random.randint(1,100)
print ("mayor=",max(a,b,c))
print ("menor=",min(a,b,c))
```

 mayor= 84
menor= 62

Propuestos

Escribir las siguientes funciones:


- `azar(x,y)` entrega entero al azar entre x e y
- `medio(x,y,z)` entrega el valor del medio entre x,y,z
- `mayor(x,y)` entrega el mayor entre x e y (sin usar max)

▼ Programas interactivos


Muchas veces un programa necesita interactuar con el usuario (cuando el usuario debe dar parámetros) y luego el programa tiene que mostrar resultados.

- **input** permite pedir datos al usuario
 - **input**: la variable ingresada será del texto
- **print()**, permite imprimir a pantalla Ej:

```
input('Ingrese un numero ')
```

 Ingrese un numero 4
'4'

```
numero = int(input('Ingrese un numero '))
```

 Ingrese un numero 4


```
numero
```

 4


```
doble = numero * 2
doble
```

 8

```
nombre = input('Cual es su nombre?')
```

 Cual es su nombre?Barbara

```
nombre
```

 'Barbara'

▼ Programa interactivo usando triangulo.py

(debe estar guardado todo en el mismo directorio y ser ejecutado desde ahí)

Ejercicio en clases (5 min)


```
import triangulo
print('Calcula el area y perimetro de un triangulo')

L1 = int(input('Ingrese el largo del primer lado '))
L2 = int(input('Ingrese el largo del segundo lado '))
L3 = int(input('Ingrese el largo del tercer lado '))

perimetro = triangulo.perimetro(L1, L2, L3)

area = triangulo.area(L1, L2, L3)

print("El perimetro es: ", perimetro)
print("El area es: ", area)
```

 Calcula el area y perimetro de un triangulo
Ingrese el largo del primer lado 2
Ingrese el largo del segundo lado 3
Ingrese el largo del tercer lado 2
El perimetro es: 7
El area es: 1.984313483298443