



FUNCIONES NATIVAS E IMPORTADAS

10145 - FUNDAMENTOS DE PROGRAMACIÓN PARA
INGENIERÍA

10110 – FUNDAMENTOS DE COMPUTACIÓN Y
PROGRAMACIÓN



RESUMEN DE CONTENIDOS



FUNCIONES NATIVAS

Función	Entradas	Salida	Proceso
abs(x)	Un número	Un número	Devuelve el valor absoluto de un número
pow(x, y)	Dos números	Un número	Es equivalente a realizar $x^{**}y$
max(x)	Un elemento iterable	Un elemento	Devuelve el elemento de mayor valor del elemento iterable
min(x)	Un elemento iterable	Un elemento	Devuelve el elemento de menor valor del elemento iterable
round(x,y)	Un flotante (x) y un entero (y)	Un flotante	Devuelve el elemento x redondeado a la cantidad de decimales informada en y



FUNCIONES NATIVAS

Función

abs(x)

pow(x, y)

max(x)

min(x)

round(x,y)

- Existen más funciones nativas en Python que las aquí señaladas, que podrían ser útiles para su proyecto o para algún problema propuesto durante el semestre

- Para consultarlas, es posible ir a la documentación de Python, bajo el apartado funciones nativas (*built-in functions*)

Un flotante
entero (y)

Un fl

can

elemento x redondeado a la
e decimales informada en y

valor del

menor valor del



SINTAXIS DE IMPORTACIÓN

```
import math
```

```
sen_angulo =  
math.sin(angulo)  
cos_angulo =  
math.cos(angulo)
```

```
import math as m
```

```
sen_angulo = m.sin(angulo)  
cos_angulo = m.cos(angulo)
```

```
from math import sin, cos
```

```
sen_angulo = sin(angulo)  
cos_angulo = cos(angulo)
```

```
from math import *
```

```
sen_angulo = sin(angulo)  
cos_angulo = cos(angulo)
```



SINTAXIS DE IMPORTACIÓN

import math

```
sen_angulo =  
math.sin(angulo)  
cos_angulo =  
math.cos(angulo)
```



from math **import** sin, cos

```
sen_angulo = sin(angulo)  
cos_angulo = cos(angulo)
```



import math **as** m

```
sen_angulo = m.sin(angulo)  
cos_angulo = m.cos(angulo)
```



from math **import** *

```
sen_angulo = sin(angulo)  
cos_angulo = cos(angulo)
```





CONSIDERACIONES A LA HORA DE IMPORTAR

- Las importaciones de módulos **SIEMPRE DEBEN SER LAS PRIMERAS LÍNEAS FUNCIONALES DE CÓDIGO**
- Es decir, antes de las importaciones sólo deberían existir comentarios, no deberían definirse constantes, funciones propias u otros elementos
- Sea cual sea la sintaxis de importación que se decida usar, **sólo debe usarse una sola en un mismo código**
- Basta con que importemos **una vez** el módulo o función para poder usarlo

CONSIDERACIONES A LA HORA DE IMPORTAR

- Cada **módulo a importar debería ir en una línea separada**, por ejemplo:

- Correcto:

- **import** os

- **import** sys

- Incorrecto:

- **import** os, sys



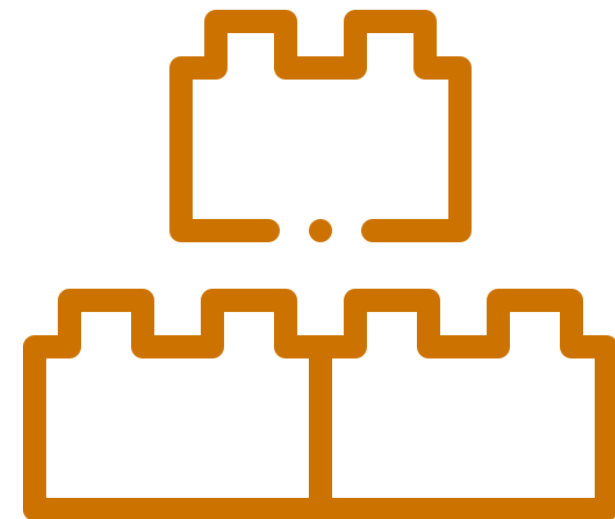
- Sin embargo, es correcto importar en una línea varias funciones de un mismo módulo, por ejemplo:

- **from** random **import** randint, choice, random



MÓDULOS

- Python trae una biblioteca estándar de módulos (generados por el mismo equipo que desarrolla y mantiene Python) donde está **math**, y otros módulos como:
 - **random**, que permite la generación de números pseudoaleatorios
 - **os**, que permite interactuar con el sistema operativo
 - **datetime**, para trabajar con horas y fechas
 - **sockets**, para trabajar con puertos de protocolos de internet
 - Entre otros
- La documentación de todos estos módulos está en la ayuda de Python, en el apartado ***The Python Standard Library***





MÓDULOS EXTERNOS

- Además de los módulos de la biblioteca estándar de Python, existe la posibilidad de importar módulos creados por terceras personas para resolver problemas particulares
- Cómo estos módulos no vienen con la instalación de Python estándar, **deben instalarse manualmente**
- Python tiene incorporado un gestor de paquetes para instalar módulos externos llamado **pip**



EJERCICIOS



EJERCICIO PROPUESTO

- Construya un programa en Python que imprima por pantalla la raíz cuadrada de los números múltiplos de 3 de la siguiente lista de valores
 - `lista = [10,33,9,14,18,14,12,21,50,55,60]`
- **Restricción: No puede usar `pow`, ni exponenciar a 0.5 para realizar el cálculo**
- **Ayuda: El módulo `math` tiene una función para calcular la raíz cuadrada**



EJERCICIO PROPUESTO

- Construya un programa en Python que simule el sorteo de un juego de azar como el LOTO o el KINO
- Considere que existe en Python el módulo random y tiene las funciones:
 - `randint(inicio, final)`: Obtiene un número pseudoaleatorio entre el inicio y el final dado
 - `choice(iterable)`: Selecciona un elemento desde la colección iterable



TAREAS PARA TRABAJO AUTÓNOMO

- Revisa la lectura 8 – Funciones propias
- Realiza los ejercicios propuestos anteriormente y, consultando la documentación de Python, intenta resolver el ejercicio:
 - Construya un programa en Python que me pida mi fecha de nacimiento en formato DD/MM/AAAA y me indique mi edad actual y cuántos días faltan para mi próximo cumpleaños
 - Considere que el módulo `datetime` tiene herramientas para apoyar la solución de este problema



¿CONSULTAS?