

Estructura de Datos y POO

Semana 1





Contenido

- Módulos
- Paquetes



PYTHON



¿Qué es un módulo?

Un módulo es un fichero que contiene definiciones y sentencias **Python**. El nombre del fichero es el nombre del módulo con el sufijo **.py**.





Importación de módulos

Un módulo puede contener tanto declaraciones de variables como definiciones de funciones. Estas declaraciones están pensadas para inicializar el módulo. Se ejecutan solamente la primera vez que el módulo se encuentra una sentencia **import**.

- **import math**
- **from math import pi**





Módulos estándar

Python viene con una biblioteca de módulos estándar los cuáles contienen funciones ya desarrolladas listas para ser utilizadas. Ejemplos de estos módulos estándar son: **math**, **sys**, **io**.





El módulo math

Este módulo contiene un buen puñado de funciones para manejar números, hacer redondeos, sumatorios precisos, truncamientos, entre otras operaciones... además de constantes.

Redondeos

```
print(math.floor(3.99)) # 3
```

```
print(math.ceil(3.01)) # 4
```

Sumatorio

```
numeros = [1, 2, 3, 4]
```

```
math.fsum(numeros) # 10
```

Trucamiento

```
math.trunc(123.45) # 123
```

Potencias y raíces

```
math.pow(2, 3) # 8.0
```

```
math.sqrt(9) # 3.0
```

Constantes

```
print(math.pi) # 3.141592653589793
```

```
print(math.e) # 2.718281828459045
```

MÓDULOS, PAQUETES



El módulo datetime

Este módulo contiene las clases esenciales para manejar fechas y horas.

```
from datetime import datetime
```

```
dt = datetime.now() # Fecha y hora actual
```

```
print(dt)
```

```
print(dt.year) # año
```

```
print(dt.month) # mes
```

```
print(dt.day) # día
```

```
print(dt.hour) # hora
```

```
print(dt.minute) # minutos
```

```
print(dt.second) # segundos
```

```
print(dt.microsecond) # microsegundos
```

```
print("{}: {}: {}".format(dt.hour, dt.minute, dt.second))
```

```
print("{} / {} / {}".format(dt.day, dt.month, dt.year))
```





El módulo random

Ofrece algunos mecanismos que permiten operar con números aleatorios

random() → genera un número flotante x entre el rango (0.0, 1.0)

randint(inicio, fin) → genera un número entero entre inicio y fin inclusive

randrange(inicio, fin, incremento) → genera un número entero entre inicio y fin-1 con incremento



MÓDULOS, PAQUETES

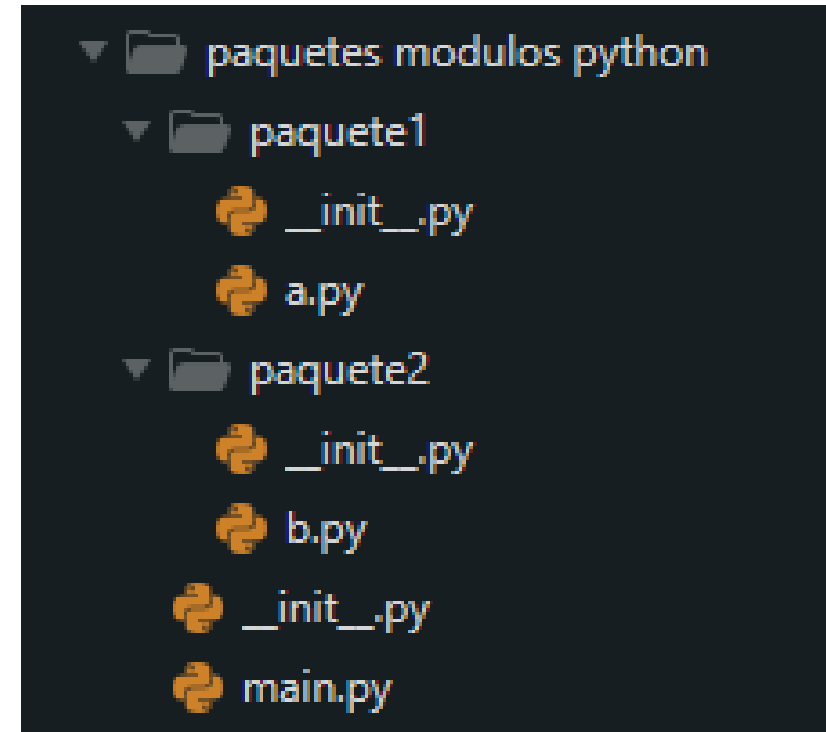


¿Qué es un paquete?

Los paquetes son ficheros que contienen definiciones que se pueden importar en otros scripts para reutilizar sus funcionalidades.

Utilizar paquetes ofrece varias ventajas. En primer lugar, permite unificar distintos módulos bajo un mismo nombre de paquete, pudiendo crear jerarquías de módulos y submódulos, o también subpaquetes.

Para crear un paquete lo que tenemos que hacer es crear un fichero especial `__init__.py` vacío en el directorio donde tengamos todos los módulos que queremos agrupar. De esta forma cuando **Python** recorra este directorio será capaz de interpretar una jerarquía de módulos.





Gracias