

# Programação

## Módulos em Python

## Recursão

Fátima Leal

**DCT** DEPARTAMENTO DE CIÊNCIA  
E TECNOLOGIA

# Até agora ...

- Classes
- Subclasses
- Variáveis privadas e públicas
- Getters e setters

# Módulos

- Um módulo é um ficheiro Python com funções que podem ser reutilizadas por outros ficheiros Python

- Exemplo:

Cria um ficheiro `adder.py` que contém a função `add (x,y)` que recebe dois valores e retorna a sua soma

```
def add(x,y):  
    return x+y
```

Cria o ficheiro `main.py` que chama a função `add (x,y)` que está no ficheiro `adder.py`

```
import adder  
  
print(adder.add(2,4))
```

# Módulos

- Cria a função `double (x)` no ficheiro `adder.py` em que retorna o dobro do valor que receber

```
def double(x):  
    return x+x
```

- Faz o `import` apenas da função `double` do ficheiro `main.py`

```
from adder import double  
  
print(double(2))
```

# Módulos

- Para renomear módulos existe a palavra as

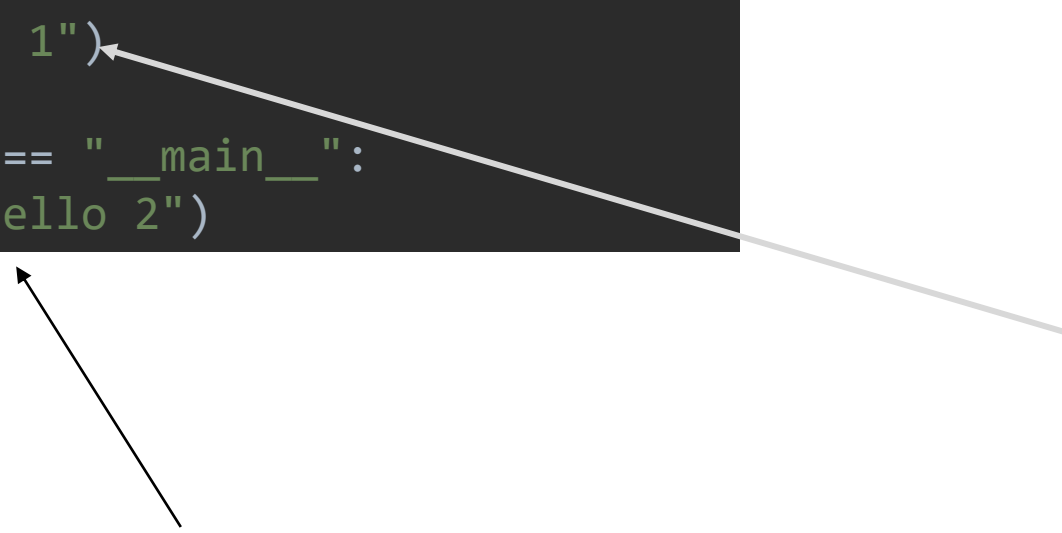
```
import adder as a  
  
print(a.double(2))
```

- Resumo:

```
import adder  
  
import adder as a  
  
from adder import double
```

# Módulos `__name__`

```
print("Hello 1")  
  
if __name__ == "__main__":  
    print("Hello 2")
```



É executado  
sempre que for  
feito o import do  
script

É executado se  
for feito o run no  
script corrente

# Diretórios Python

- A linguagem Python permite que os scripts sejam importados dentro de packages
- Sempre que criamos um package é criado um script de suporte `__init__.py`
- Este script está vazio. Apenas identifica que é um package python

```
import newPackage.add  
newPackage.add.double(2)
```

Import do  
package

Chamada do método que  
estava dentro do novo  
package

# Módulos

- Math module – funções matemáticas
- Numpy module – cálculo de matrizes
- Pandas module – tabelas e dados
- Sklearn module – machine learning
- Etc.



# Recursão

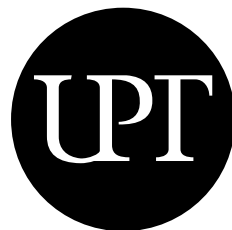
- Recursão acontece quando uma função é chamada dentro da própria função
- Teoricamente, a recursão pode chamar a própria função infinitamente, mas o Python coloca limites

```
def main():  
    message()  
  
def message():  
    print('This is a recursive function.')    message()  
  
# Call the main function.  
main()
```

# Recursão

```
def main():  
    message(5)  
  
def message(times):  
    if times>0:  
        print('This is a recursive function.')        message(times-1)  
  
# Call the main function.  
main()
```

Para  
estabelecer  
um limite na  
chamada da  
função  
recursiva é  
necessário  
definir um if



UNIVERSIDADE  
PORTUCALENSE

Do conhecimento à prática.