

ENTRADA I SORTIDA DE DADES

print

input



M^a Belén Tortosa Pedrón

print



1. **print()** separa cada argument usant un espai en blanc.
2. Cada impressió amb **print()** acaba amb un salt de línia.
3. **print()** admet un argument, anomenat **sep**, que conté la cadena de caràcters que utilitzarem per separar cada element. Si no s'especifica, pren el seu valor por defecte, un espai en blanc, **sep=" "**. Per exemple: Separació dels diferents elements amb un guió:

```
>>> print("Nom","Cognom", "Edat",sep="-")
```

```
>>> print("cogombre","tomaquet","enciam",sep="\t")
```

4. **print()** admet indicar també com es vol que termini la impressió utilitzant un nou argumento opcional, **end**. Si no s'especifica, pren el seu valor per defecte, provoca un salt de línia, representat pel caràcter d'escapament `\n`.

```
>>> print("cogombre","colorant", sep=" ",end="---")
```

```
>>> print("tomaquet")
```

```
>>> a=Blau'
```

```
>>> b='Verd'
```

```
>>> c='Vermell'
```

```
>>> print(a,b,c,sep=', ',end='.')
```

```
Blau, Verd, Vermell.
```

input



- ❑ **input()**: Funció que permet demanar dades a l'usuari i emmagatzemar-les en un variable. A l'arribar a aquesta funció, el programa es para esperant que l'usuari escrigui alguna cosa pel teclat i polsi INTRO.

```
1
2 print("Com et dius?")
3 nom = input()
4 print("Encantat de conèixer-lo,", nom)
5
```

```
C:\WINDOWS\system32\cmd.exe
Com et dius?
POL
Encantat de conèixer-lo, POL
```

- ❑ Si es prefereix que l'usuari escrigui la seva resposta a continuació de la pregunta, es podria utilitzar l'argument opcional **end** en la funció **print()**, que indica el caràcter o caràcters a utilitzar en lloc del salt de línia. Per a separar la resposta de la pregunta s'ha afegit un espai al final de la pregunta.

```
1
2 print("Com et dius? ", end="")
3 nom = input()
4 print("Encantat de conèixer-lo,", nom)
```

```
C:\WINDOWS\system32\cmd.exe
Com et dius? NIL
Encantat de conèixer-lo, NIL
```

- ❑ Una altra solució és: aprofitar que a la funció **input()** se li pot enviar un argument que s'escriu en la pantalla (sense afegir un salt de línia):

```
1
2 print("Com et dius?")
3 nom = input()
4 print("Encantat de conèixer-lo,", nom)
5
```

```
C:\WINDOWS\system32\cmd.exe
Com et dius?
POL
Encantat de conèixer-lo, POL
```

CONVERSIÓ DE TIPUS



- De forma predeterminada, la funció **input()** converteix l'entrada en una cadena. Si es vol que Python intèrpret l'entrada com un número enter, s'ha d'utilitzar la funció **int()**:

```
1
2 quantitat = int(input("Díga'm una quantitat en pesetes: "))
3 print(quantitat, "pesetes són", quantitat / 166.386, "euros")
4
```

```
C:\WINDOWS\system32\cmd.exe
Díga'm una quantitat en pesetes: 200
200 pesetes són 1.2020242087675646 euros
```

- Per a que Python interpreti l'entrada com un número decimal, s'ha d'utilitzar la funció **float()**:

```
1
2 quantitat = float(input("Díga'm una quantitat en euros (fins a 2 decimals): "))
3 print(quantitat, "euros són", quantitat * 166.386, "pesetes")
4
```

```
C:\WINDOWS\system32\cmd.exe
Díga'm una quantitat en euros (fins a 2 decimals): 1.56
1.56 euros són 259.56216 pesetes
```

- Però si l'usuari no escriu un número, les funcions **int()** o **float()** produiran una errada:

```
2 quantitat = float(input("Díga'm una quantitat en euros (fins a 2 decimals): "))
3 print(quantitat, "euros són", quantitat * 166.386, "pesetes")
4
5 C:\WINDOWS\system32\cmd.exe
6 Díga'm una quantitat en euros (fins a 2 decimals): Pepito
7 Traceback (most recent call last):
8   File "M:\Curs2016-2017\codi\ejercicios\UF1\prova1.py", line 2, in <module>
9     quantitat = float(input("Díga'm una quantitat en euros (fins a 2 decimals): "))
10 ValueError: could not convert string to float: 'Pepito'
```

- Si l'usuari escriu un número decimal, la funció **int()** produirà una errada. Però si l'usuari escriu un número enter, la funció **float()** **no** produirà una errada, encara que el número s'escrigui amb part decimal (.0):

VARIABLES COM ARGUMENT DE LA FUNCIO input()



- La funció **input()** només pot tenir un argument. Però utilitzant l'operador concatenació (+), podem passar variables com argument a la funció **input()**. Exemple:

```
1
2 nom = input("Com et dius?: ")
3 cognom = input("El seu cognom?, " + nom + ": ")
4 print("Encantat de conèixer-lo,", nom, cognom)
5
```

C:\> Seleccionar C:\WINDOWS\system32\cmd.exe

Com et dius?: antonio
El seu cognom?, antonio: esteve
Encantat de conèixer-lo, antonio esteve

- Si es volen concatenar números, s'ha d'utilitzar la funció **str()**:

```
1
2
3 numero1 = int(input("Entra un número: "))
4 numero2 = int(input("Entra un número major que " + numero1 + ": "))
5 print("La diferencia entre ells és", numero2 - numero1)
6
7
```

C:\> C:\WINDOWS\system32\cmd.exe

Entra un número: 1
Traceback (most recent call last):
File "M:\Curs2016-2017\codi\ejercicios\UF1\prova1.py", line 4, in <module>
 numero2 = int(input("Entra un número major que " + numero1 + ": "))
TypeError: Can't convert 'int' object to str implicitly

```
1
2
3 numero1 = int(input("Entra un número: "))
4 numero2 = int(input("Entra un número major que " + str(numero1) + ": "))
5 print("La diferencia entre ells és", numero2 - numero1)
6
```

C:\> Seleccionar C:\WINDOWS\system32\cmd.exe

Entra un número: 2
Entra un número major que 2: 3
La diferencia entre ells és 1

```
2 nom = input("Com es diu? ")
3 print("Quin és el seu cognom?,", nom, ": ", end="")
4 cognom = input()
5 print("Encantat de conèixer-lo.", nom, cognom)
```

C:\> Seleccionar C:\WINDOWS\system32\cmd.exe

Com es diu? Carlos
Quin és el seu cognom?, Carlos : Saura
Encantat de conèixer-lo, Carlos Saura

MÈTODES PER PRINTAR AMB FORMAT



- ❑ El mètode **str.rjust()** dels objectes cadena: Ordena una cadena a la dreta en un camp de l'ample donat omplint-lo amb espais a l'esquerra.
- ❑ Mètodes similars **str.ljust()** i **str.center()**.
- ❑ Aquests mètodes no escriuen res, només retornen una nova cadena. Si la cadena d'entrada és massa llarga, no la trunquen, si no que la retornen intacta; encara que això trenqui l'alineació de les columnes.

```
1 x=2
2 print(str(x).rjust(2), str(x * x).rjust(3), end=' ')
3     # notar el uso de 'end' en la linea anterior
4 print(repr(x * x * x).rjust(4))
5
6 x=10
7 print(str(x).rjust(2), str(x * x).rjust(3), end=' ')
8     # notar el uso de 'end' en la linea anterior
9 print(str(x * x * x).rjust(4))
10
```

C:\WINDOWS\system32\cmd.exe

```
2  4  8
10 100 1000
```

- ❑ **str.zfill()**: Omple una cadena numèrica a l'Esquerra amb zeros. Distingeix signes positius i negatius

```
1 x=12
2 print(str(x).zfill(5))
3
4 x=-3.14
5 print(str(x).zfill(7))
6
7 x=3.14159265359
8
9 print(str(x).zfill(5))
```

C:\WINDOWS\system32\cmd.exe

```
00012
-003.14
3.14159265359
```

UTILITZACIÓ MÈTODE `format`



- ❑ **format**: Els claudàtors i caràcters a dintre de les mateixes (anomenats camps de format) són reemplaçats amb els objectes passats en el mètode **`str.format()`**.

```
1
2 print('Somos los {} quienes decimos "{}!".format('caballeros', 'Nop'))
3
```

```
C:\WINDOWS\system32\cmd.exe
Somos los caballeros quienes decimos "Nop!"
```

- ❑ Un número en las llaves es refiere a la posición del objeto pasado en el método.

```
3
4 print('{0} i {1}'.format('carn', 'ous'))
5
6 print('{1} i {0}'.format('carn', 'ous'))
7
```

```
C:\WINDOWS\system32\cmd.exe
carn i ous
ous i carn
```

- ❑ Si s'usen arguments nombrats en el mètode **`str.format()`**, els seus valors serán referits usant el nom de l'argument.

```
3
4 print('Aquesta {comida} és {adjetivo}.'.format(comida='carn', adjetivo='espantosa'))
5
```

```
Seleccionar C:\WINDOWS\system32\cmd.exe
Aquesta carn és espantosa.
```

- ❑ Es poden barrejar arguments posicionals i nombrats.

```
2
3 print('La història de {0}, {1}, y {altre}.'.format('Bill', 'Manfred', altre='Georg'))
.
```

```
Seleccionar C:\WINDOWS\system32\cmd.exe
La història de Bill, Manfred, y Georg.
```

UTILITZACIÓ MÈTODE format



- ❑ Un ':' i especificador de format opcionals poden anar després del nom del camp. Això augmenta el control sobre com el valor es formatat. El següent exemple arrodoneix Pi a tres llocs després del punt decimal.

```
3 import math
4 print('El valor de PI és aproximadament {0:.3f}'.format(math.pi))
5
6
7 C:\WINDOWS\system32\cmd.exe
8 El valor de PI és aproximadament 3.142.
```

- ❑ Posant un número enter després del ':' causarà que el camp sigui d'un mínim número de caràcters d'ample. Això és útil per fer taules.

```
2 print('{0:6} ==> {1:50}'.format("AWS1", "Desenvolupament d'Aplicacions Web"))
3
4 print('{0:6} ==> {1:50}'.format("TIM1", "Explotació de Sistemes Informàtics"))
5
6
7 C:\WINDOWS\system32\cmd.exe
8 AWS1 ==> Desenvolupament d'Aplicacions Web
9 TIM1 ==> Explotació de Sistemes Informàtics
```


VELL FORMATEIG DE CADENES



- ❑ L'operador **%** també pot usar-se per formateig de cadenes. Interpreta l'argument de l'Esquerra amb l'estil de formateig de **sprintf()** per ser aplicat a l'argument de la dreta, i retorna la cadena resultant d'aquesta operació de formateig. Per exemple:

```
3 import math
4 print('El valor de PI és aproximadament %5.2f.' % math.pi)
5
```

```
C:\WINDOWS\system32\cmd.exe
```

```
El valor de PI és aproximadament  3.14.
```

EXERCICI



- ❑ Crea las variables necessàries per realitzar un programa que generi la següent sortida per pantalla:

```
AW1 -> Programació  
AWS1 -> Elements  
AW2 -> Població
```

```
6  
56  
123  
4
```

```
003  
013  
023  
004
```