

Langage Python

Lecture3: Entrées/Sorties

Abderrahim MESBAH
a.mesbah@um5r.ac.ma



Plan

- Sorties
- Entrées



Sorties

Affichage simple

- Le moyen le plus simple d'utiliser la commande `print` est de répertorier les variables à imprimer, séparées par une virgule.

```
print " Hello World "    # valide en Python 2.x
```

```
print (" Hello World ")  # valide en Python 3.x
```

Affichage simple

- Le moyen le plus simple d'utiliser la commande `print` est de répertorier les variables à imprimer, séparées par une virgule.

```
In [1]: a = 10
```

```
In [2]: b = 'test text'
```

```
In [3]: print(a)
```

```
10
```

```
In [4]: print(b)
```

```
test text
```

```
In [5]: print(a,b)
```

```
10 test text
```

```
In [6]: print("the answer is",a)
```

```
the answer is 10
```

```
In [7]: print("addition result:",5+6)
```

```
Addition result:11
```

Affichage formaté

■ Une chaîne peut contenir des identifiants de format tels que :

- `%f` pour formater en tant que flottants,
- `%d` pour formater en tant qu'entiers,
- `%s` pour formater en tant que chaîne.
- `%e` notation exponentielle

```
In [8]: print ("a = %d b = %d" % (10 ,20))  
a = 10 b = 20
```

- Le format **% W.Df** signifie qu'un flottant doit être imprimé avec une largeur totale de **W** caractères et **D** chiffres après le point décimal.

```
In [2]: from math import pi
```

```
In [3]: print("Pi =%5.2f" % pi)
```

```
Pi = 3.14
```

```
In [4]: print ("Pi = %10.3f" % pi)
```

```
Pi =    3.142
```

```
In [5]: print ("Pi = %10.8f" % pi)
```

```
Pi =    3.14159265
```

```
In [6]: print ("Pi = %d" % pi)
```

```
Pi = 3
```

```
In [7]: print("Pi =%f, 142*pi =%f and pi^2 =%f." %(pi ,142*pi ,pi**2))
```

```
Pi =3.141593, 142*pi =446.106157 and pi^2 =9.869604.
```

- Un nouveau système de formatage intégré a été proposé et vise à remplacer à long terme le formatage d'opérateur de style ancien (%).

```
In [11]: "{} needs {} pints ". format ('Peter ',4)
```

```
Out[11]: 'Peter needs 4 pints '
```

```
In [12]: " {0} needs {1} pints ". format ('Peter ',4)
```

```
Out[12]: ' Peter needs 4 pints '
```

```
In [13]: " {1} needs {0} pints ". format ('Peter ',4)
```

```
Out[13]: ' 4 needs Peter pints '
```

```
In [14]: import math
```

```
In [15]: "Pi is approximately {:.2f} ". format ( math .pi)
```

```
Out[15]: 'Pi is approximately 3.14.'
```


Séquence d'échappement

- Un nouveau système de formatage intégré a été proposé et vise à remplacer à long terme le formatage d'opérateur de style ancien (%).

Séquence	Signification
\saut_ligne	saut de ligne ignoré (en fin de ligne)
\\	affiche un antislash
\'	apostrophe
\"	guillemet
\a	sonnerie (<i>bip</i>)
\b	retour arrière
\f	saut de page
\n	saut de ligne
\r	retour en début de ligne
\t	tabulation horizontale
\v	tabulation verticale



Entrées

- La fonction `input()` effectue toujours une saisie en mode texte (la valeur retournée est une chaîne) dont on peut ensuite changer le type

```
In [18]: f1 = input("Entrez un flottant :")  
Entrez un flottant :12
```

```
In [19]: type(f1)  
Out[19]: str
```

```
In [20]: f2 = float(input("Entrez un autre flottant :"))
```

```
Entrez un autre flottant :12.345
```

```
In [21]: type(f2)  
Out[22]: float
```

- **Ecrire un programme qui demande à l'utilisateur de taper la longueur et la largeur d'un rectangle et qui affiche le périmètre et la surface**