

Cours n° 3: Variables et types de données

المتغيرات وأنواع البيانات

1 ère année Sciences économiques, de gestion
et commerciales

Plan du cours

- 1 Introduction
- 2 Définir et nommer les variables تعريف المتغيرات وقواعد التسمية
- 3 Les types de données de base أنواع البيانات
- 4 Exercices pratiques كتابة برامج بسيطة
- 6 Conclusion

Introduction

En Python, tout programme commence par la **manipulation de données**.
التعامل مع البيانات.

Pour stocker et utiliser ces données, on utilise des **variables**.
المتغيرات

تعريف المتغيرات ? Qu'est-ce qu'une variable ?

- Une **variable** est un **nom** que l'on donne à une **valeur** pour pouvoir la réutiliser. اسم نمحه لقيمة معينة
- C'est comme une **boîte** dans laquelle on range une information. يشبه صندوقاً نضع بداخله معلومة

Exemple simple :

nom = "Yasmine"

age = 19

Ici :

On a une boîte appelée nom qui contient "Yasmine"

Et une autre appelée age qui contient 19

On pourra ensuite les afficher, les modifier ou les utiliser dans des calculs.

تعريف المتغيرات? Qu'est-ce qu'une variable

Dans la vie d'un gestionnaire, on manipule constamment des **valeurs** : قيم مختلفة

- سعر المنتج،
- الكمية المباعة،
- إجمالي الإيرادات،
- أو اسم الزبون.
- ➡ le prix d'un produit,
- ➡ la quantité vendue,
- ➡ le revenu total,
- ➡ ou encore le nom d'un client.

Les **variables** (المتغيرات) en Python permettent de représenter ces **informations** (المعلومات) sous forme **numérique** (عددية) ou **textuelle** (نصية) et de les traiter automatiquement (معالجتها تلقائيًا).

Règles de dénomination قواعد التسمية

- ☞ Doit commencer par une **lettre** ou un **underscore** (**_**)
بشرطة سفلية (**_**) أو بحرف
- ☞ Pas **d'espace** ni de **caractères spéciaux**
الرموز الخاصة الفراغات
- ☞ Sensible à la casse (Age \neq age)

Exemples valides et invalides :

nom_client = "Ahmed" # ✓

2nom = "Ali" # ✗ # Commence par un chiffre يبدأ برقم

nom client = "Sara" # ✗ # Espace interdit يمنع الفراغ

قواعد التسمية Règles de dénomination

Remarques:

☞ **Une variable peut être réutilisée :** إعادة استخدام المتغير

```
x = 5
```

```
x = x + 3
```

```
print(x)
```

Résultat: 8

☞ **Affectation multiple:** إسناد عدة قيم في نفس السطر (إسناد متعدد)

```
a, b, c = 1, 2, 3
```

```
print(a, b, c)
```

Résultat: 1 2 3

Les types de données de base أنواع البيانات: Il existe 4 types principaux :

Type	Signification	Exemple	Description
str	chaîne de caractères سلسلة المحارف	"Bonjour", 'Ali '	Texte
int	Entier عدد صحيح	5, -10, 100	Nombre sans virgule
float	Décimal عشري	3.14, -0.5	Nombre à virgule
bool	Booléen منطقي	T True, F alse	Valeurs logiques

Exemples en Python:

```
x = 10      # int
y = 3.14    # float
nom = "Sara" # str
est_etudiant = True # bool
```


أنواع البيانات Les types de données de base

✓ **Vérifier le type** : type() : باستعمال الدالة : التحقق من نوع البيانات

Exemple: print (type(x))

Résultat : <class 'int'>

➤ **Conversions entre types (casting) :** التحويل بين الأنواع

De str à int (لكي نستعمله في الحساب، يجب تحويله إلى نوع int)

```
age = "20"  
age = int(age)  
print(age+5)
```

Résultat : 25

De int à float إذا أردنا إضافة فاصلة عشرية إلى العدد، نحوله إلى نوع (float)

```
x = 5  
x = float(x)  
print(x)
```

Résultat : 5.0

عند طباعة عدد داخل جملة نصية، يجب أولاً تحويله إلى نص.

De float à str

```
prix = 12.5  
prix = str(prix)  
print("le prix est de"+prix)  
Résultat : le prix est de 12.5
```

أنواع البيانات Les types de données de base

سلاسل المحارف Les chaînes de caractères (str)

- Le type str (abréviation de *string*) représente un **texte** écrit **entre guillemets** (" " ou ' ').
النصوص
- On peut les **concaténer**, connaître leur **longueur** et **modifier** leur **forme** (ex. ^{الدمج} en majuscules).
معرفة الطول. تغيير شكلها

Exemples en Python:

```
prenom = "ali"
```

```
nom = "ben"
```

```
print(prenom + " " + nom) # Concaténation
```

دمج النصوص

```
print(len(prenom)) # Longueur de la chaîne
```

طول السلسلة

```
print(prenom.upper()) # Conversion en majuscules
```

تحويل إلى أحرف كبيرة

أنواع البيانات Les types de données de base

□ Les nombres

- **Le type int** (abréviation de *integer*) sert à représenter des **nombres entiers**, positifs ou négatifs, sans virgule. Exemples: 5, 0, -10, 2025
الأعداد الصحيحة بدون فاصلة عشرية

Exemples en Python:

```
année = 2025  
température = -3
```

- **Le type float** (abréviation de *floating point number*) représente les nombres avec une partie décimale (virgule). Exemples : 3.14, -0.5, 2.0
الأعداد العشرية

Exemples en Python:

```
prix = 12.5  
poids = 65.8
```

أنواع البيانات Les types de données de base

القيم المنطقية Les booléens (bool)

Le type bool (abréviation de *boolean*) sert à représenter une valeur **vraie ou fausse**; True (vrai) ou False (faux).

Exemples en Python:

```
est_ouvert = True  
print("Le magasin est ouvert ?", est_ouvert)
```

Résultat :

Le magasin est ouvert ? True

Exercices pratiques — كتابة برامج بسيطة

Exercices proposés :

Ex n°1: Affichage d'informations personnelles

عرض المعلومات الشخصية

nom = "Omar"

age = 21

ville = "Jijel"

```
print("Nom :", nom)
```

```
print("Âge :", age)
```

```
print("Ville :", ville)
```

Résultat:

Nom : Omar

Âge : 21

Ville : Jijel

Exercices pratiques — كتابة برامج بسيطة

Ex n°2: Afficher un texte avec une variable

عرض نص يحتوي على متغيرات

```
nom = "Fatima"
```

```
age = 19
```

```
print("Je m'appelle", nom, "et j'ai", age, "ans.")
```

Résultat:

Je m'appelle Fatima et j'ai 19 ans.

Exercices pratiques — كتابة برامج بسيطة

Ex n°3: Interaction avec l'utilisateur التفاعل مع المستخدم

```
nom = input("Entrez votre nom : ")
```

```
age = input("Entrez votre âge : ")
```

```
print("Bienvenue", nom, "!")
```

```
print("Vous avez", age, "ans.")
```

Résultat:

Entrez votre nom : Yasmine

Entrez votre âge : 19

Bienvenue Yasmine !

Vous avez 19 ans.

Conclusion (Résumé du cours)

- ✓ Une variable = espace mémoire qui stocke une valeur
- ✓ Types principaux : int, float, str, bool
- ✓ Règles de nommage simples mais importantes
- ✓ Les variables sont la base de tout programme Python