

# Exercices constructions élémentaires Éléments de correction

Christophe Viroulaud

Première - NSI

**Lang 02**

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Exercice 1

1  $a = 3$

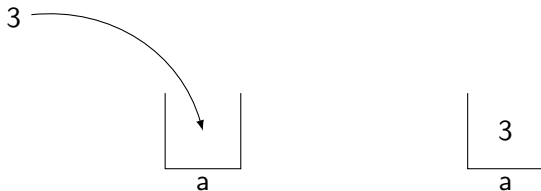


FIGURE 1 – Affectation

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

1  $a = 4$

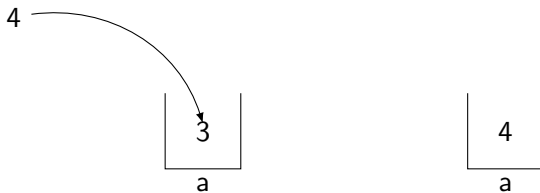


FIGURE 2 – Nouvelle affectation

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

1  $a = a + 2$

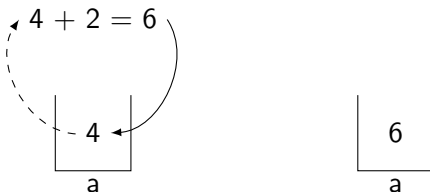


FIGURE 3 – Calcul puis affectation

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

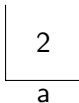
Exercice 9

Exercice 10

Exercice 11

Exercice 11

1  $a = 2$



Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

1  $b = a * a$

2  
a

4  
b

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

1  $b = a * b$

2  
a

8  
b

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11



1  $b = b * b$

2

a

64

b

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

```
1 print("i+") # affiche i+
2 print(i+) # message d'erreur: on essaie d'
    ajouter i à ... rien
```

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

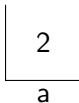
Exercice 9

Exercice 10

Exercice 11

Exercice 11

1  $a = 2$



Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

1  $b = 3$

2  
a

3  
b

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

1 `tmp = a`

2  
a

3  
b

2  
tmp

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

1 `a = b`

3  
a

3  
b

2  
tmp

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

```
1 b = tmp
```

3
---

a

2
---

b

2
---

tmp

## À retenir

La séquence inverse (*swap*) les valeurs de **a** et **b**.

## Remarque

Python facilite cette opération :

```
1 a, b = b, a
```

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11



## Exercice 2

```
1 longueur = int(input("Longueur (en cm): "))
2 largeur = int(input("Largeur (en cm): "))
3 print("L'aire du rectangle est {}cm2".format(
    longueur*largeur))
```

Code 1 – Aire d'un rectangle

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

## Exercice 3

```
1 age = int(input("Quel est votre âge? "))
2 if age >= 18:
3     print("Vous êtes majeur.")
4 else:
5     print("Vous êtes mineur.")
```

Code 2 – Âge

### Remarque

`input` renvoie une chaîne de caractère (**string**). Il faut la convertir en entier (**int**).

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

## Exercice 4

```
1 age = int(input("Quel est votre âge? "))
2 if age < 16:
3     print("Le prix de la carte est €10.")
4 else:
5     if age <= 25:
6         ...
```

Code 3 – Cinéma - première version

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

## Exercice 4

```
1 age = int(input("Quel est votre âge? "))
2 if age < 16:
3     print("Le prix de la carte est €10.")
4 elif age <= 25:
5     print("Le prix de la carte est €15.")
6 elif age <= 59:
7     print("Le prix de la carte est €25.")
8 else:
9     print("Le prix de la carte est €15.")
```

Code 4 – Cinéma - seconde version

### Remarque

Ligne 4 : inutile de vérifier si  $\text{age} \geq 16$ , c'est forcément le cas.

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Exercice 5

```
1 from random import randint
2
3 somme = 0
4 for i in range(10):
5     nb = randint(1, 10)
6     somme += nb
7 print(somme)
```

Code 5 – Somme

## Remarque

Ne pas oublier d'importer la bibliothèque.

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11



# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

## Exercice 6

```
1 from random import randint
2
3 nb = randint(1,10)
4 essai = 0
5 trouve = False
6 while not trouve:
7     proposition = int(input("Quel nombre? "))
8     if proposition == nb:
9         trouve = True
10    essai += 1
11 print(essai)
```

Code 6 – Deviner - première version

### Remarque

On utilise une variable *booléenne*.

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

```
1 from random import randint
2
3 nb = randint(1,10)
4 essai = 1
5 # compare la proposition de l'utilisateur à nb
6 while not(int(input("Quel nombre? ")) == nb):
7     essai += 1
8 print(essai)
```

Code 7 – Deviner - première version

## Remarque

On compare directement l'entrée avec la valeur de `nb`.

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Exercice 7

- ▶  $20/3$  renvoie le résultat de la division. Nous reviendrons plus tard sur le *type* de ce résultat.
- ▶  $20//3$  renvoie la partie entière de la division. C'est un *entier*.
- ▶  $20\%3$  renvoie le reste de la division. C'est un *entier*. On appelle cette opération le *modulo*.

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

```
1 secondes = int(input("Donnez le nombre de  
    secondes: "))  
2 heures = secondes // 3600  
3 minutes = (secondes % 3600) // 60  
4 secondes = (secondes % 3600) % 60
```

### Code 8 – Durée

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

```
1  if heures < 10:  
2      heures = "0"+str(heures)  
3  if minutes < 10:  
4      minutes = "0"+str(minutes)  
5  if secondes < 10:  
6      secondes = "0"+str(secondes)  
7  print("{}h {}min {}s".format(heures, minutes,  
    secondes))
```

### Code 9 – Affichage

#### Remarque

Les variables sont des entiers et deviennent des chaînes de caractères (string). Python permet de changer le type d'une variable. Ce n'est pas le cas de tous les langages.

[Exercice 1](#)[Exercice 2](#)[Exercice 3](#)[Exercice 4](#)[Exercice 5](#)[Exercice 6](#)[Exercice 7](#)[Exercice 8](#)[Exercice 9](#)[Exercice 10](#)[Exercice 11](#)[Exercice 11](#)

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

**Exercice 8**

Exercice 9

Exercice 10

Exercice 11

Exercice 11



# Exercice 8

```
1 nb = int(input("Quelle table? "))
2 for i in range(11): # 11 signifie qu'il y aura
    11 tours
3     print(f"{i}×{nb} = {i*nb}")
```

## Code 10 – Multiplication

### Remarque

Noter ici le f en début de ligne qui est une autre manière de formater le texte (pour des versions récentes de Python ( $\geq 3.6$ )). Il est possible d'écrire :

```
1 print("{}×{} = {}".format(i,nb,i*nb))
```

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Exercice 9

```
1 for i in range(10,-1,-1):  
2     # range(premier terme (inclus), dernier  
   terme (exclu), pas)  
3     print(i)
```

Code 11 – Compte à rebours

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Exercice 10

```
1 for i in range(2,25,2):  
2     print(i, end=" ")
```

Code 12 – Nombres pairs

## Remarque

L'option `end` de `print` définit le caractère à mettre en fin de ligne (retour chariot par défaut).

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Exercice 11

```
1 somme = 0
2 for i in range(10):
3     somme += int(input("note: "))
```

Code 13 – Moyenne

## Remarque

Il faut noter ici l'ordre dans lequel l'interprète lit cette ligne :

- ▶ il lit la valeur du `input`,
- ▶ il la convertit en entier,
- ▶ il additionne cette valeur à `somme`.

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

```
1 somme = 0
2 for i in range(10):
3     somme += int(input("note: "))
4 moyenne = round(somme/10, 2)
```

### Code 14 – Moyenne

#### Remarque

La fonction `round` permet d'arrondir ici à 2 chiffres après la virgule

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11



```
1  if moyenne >= 15:
2      print("{} / 20, félicitations!".format(moyenne))
3
4  elif moyenne >= 10:
5      # il est inutile ici de vérifier si moyenne < 15
6      print("{} / 20, bon travail!".format(moyenne))
7  else:
8      print("{} / 20, doit fournir des efforts!".format(moyenne))
```

## Code 15 – Affichage

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Sommaire

1. Exercice 1

2. Exercice 2

3. Exercice 3

4. Exercice 4

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Exercice 11

```
1 mini = 0
2 maxi = 100
3 trouve = False
4 coups = 0
5
6 print("Pensez à un nombre entre 1 et 100.")
```

Code 16 – Devinette

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11

# Exercice 11

```
1 while not trouve:
2     coups += 1
3     # choix de la valeur (milieu de l'intervalle)
4     proposition = (mini + maxi)//2
5     print("Le nombre est-il {}?".format(
6         proposition))
7     reponse = input("Merci de répondre = + ou - :
8         ")
9     if reponse == "=":
10        print("J'ai trouvé en {} coups!".format(
11            coups))
12        trouve = True
13        elif reponse == "+": # réduction de l'
14            intervalle
15            mini = proposition
16        else: # réduction de l'intervalle
17            maxi = proposition
```

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

Exercice 6

Exercice 7

Exercice 8

Exercice 9

Exercice 10

Exercice 11

Exercice 11