Exercices constructions élémentaires Éléments de correction

Christophe Viroulaud

Première - NSI

ercice 2

vercice 7

kercice 8

xercice 9

ercice 10

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
- 2. Exercice 2
- 3. Exercice 3
- 4 Exercice
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice 7
- 8. Exercice

Exercice 1

ercice 2

ercice 3

ercice 5

rcice b

rcice 8

rcice 9

rcice 10

rcice 11

rcice 11



FIGURE 1 – Affectation

а

Exercices constructions élémentaires Éléments de correction

Exercice 1

Exercice 2

ACI CICC 3

tercice 4

Exercice 5

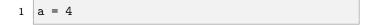
_ . _

xercice 8

xercice 9

Exercice 10

а



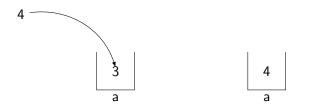


FIGURE 2 – Nouvelle affectation

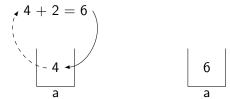
ercice 2

.xercice J

......

xercice 10

$$a = a + 2$$



 $\label{eq:Figure 3-Calcul puis affectation} Figure \ 3-Calcul \ puis \ affectation$

xercice 2

....

Kercice 4

Exercice 5

LXEICICE U

ercice 1

. .

. 10

1 a = 2

2 a Exercice 1

tercice 2

......

Kei Cice 4

xercice 5

xercice 10

Xercice 11

1	b = a*a			
		ı		1
		2	4	

b

а

Exercice 1

xercice 2

ACI CICC 5

....

xercice 5

. . .

xercice 10



Exercice 1

xercice 2

cercice 4

exercice 5

exercice 7

1	b = b*b	

2 a 64 b

Exercice 1

xercice 2

ACI CICC 3

cercice 4

Exercice 5

exercice 7

vorcico 10

```
1
2
```

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

			e	

cercice 4

Exercice 5

Exercice 7

xercice 8

xercice 9

exercice 10

1 a = 2

2 a Exercice 1

cercice 2

....

ercice 4

xercice 5

kercice o

xercice /

Kercice 5

xercice 10

2 a 3 b

Exercice 1

kercice 2

....

ACI CICC 4

xercice 5

Aereice 0

exercice 7

vercice 10

vercice 11

Exercice 1

1	tmp = a	

2 a

3 b 2

tmp

xercice 9

exercice 10

vercice 11

Exercice 1

1	a = b	

3 a

3 b 2

tmp

xercice 9

xercice 10

1 b = tmp

•

2

2 tmp

а

3

À retenir

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

Remarque

Python facilite cette opération :

Exercices constructions élémentaires Éléments de correction

Exercice 1

kercice 2

xercice 5

.....

.....

versice 10

Exercices constructions élémentaires Éléments de correction

1. Exercice 1

2. Exercice 2

3. Exercice 3

4 Exercice

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

ercice 1

Exercice 2

ercice 3

ercice 5

rcice 7

cice 8

rcice 9

rcice 10

cice 10

rcice 11

rcice 11

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

Code 1 – Aire d'un rectangle

Exercices constructions élémentaires Éléments de correction

Exercice 2

ercice 3

xercice 4

exercice 5

0.0.00

xercice 9

xercice 10

xercice 1

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
 - cico 2
- 3. Exercice 3
- 4 Evercice
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice 7
- 8. Exercice 8

ercice 1

tercice 2

Exercice 3

ercice 6

rcice 7

cice 8

rcice 9

cice 10

rcice 11

rcice 11

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

Code 2 - Âge

Remarque

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

xercice 1

xercice 2

Exercice 3

Eversies 5

Exercice 6

kercice 7

.....

. 10

.Xercice 10

vercice 11

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
- 2. Exercice 2
- 3. Exercice 3
- 4. Exercice 4
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice 7
- 8. Exercice 8

ercice 1

rcice 3

Exercice 4

ercice 5

rcice /

cice 8

rcice 10

rcice 11

rcice 11

cice 11

```
age = int(input("Quel est votre âge? "))
if age < 16:
   print("Le prix de la carte est €10.")
else:
   if age <= 25:
   ...</pre>
```

Code 3 – Cinéma - première version

xercice 2

Exercice 4

exercice 5

ercice r

exercice 10

```
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.")</pre>
```

Code 4 - Cinéma - seconde version

Remarque

Ligne 4 : inutile de vérifier si age >=16, c'est forcément le cas.

ercice 1

ercice 2

Exercice 4

LACICICE J

_ . _

......

......

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
 - rice 2
- 3 Exercice 3
- 4 Exercice 4
- 5. Exercice 5
- 6. Exercice
- 7. Exercice 7
- 8. Exercice

ercice 1

rcice 3

ercice 4

Exercice 5

rcice 7

cice 8

rcice 9

cice 10

rcice 11

```
from random import randint

somme = 0
for i in range(10):
   nb = randint(1, 10)
   somme += nb
print(somme)
```

Code 5 - Somme

Remarque

Ne pas oublier d'importer la bibliothèque.

Exercice 1

kercice 2

Exercice 5

ercice 8

orcico O

kercice 10

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
- 2 Exercice 2
- 3 Exercice 3
- 4 Evercice A
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice 7
- 8. Exercice 8

ercice 1

ercice 2

kercice 5

Exercice 6

ercice 8

ercice 9

rcice 10

rcice 11

ercice 11

rcice 11

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

Code 6 – Deviner - première version

Remarque

On utilise une variable booléenne.

Exercice 1

vercice 2

EXCITION O

Exercice 6

Exercice 7

. . .

xercice 10

xercice 1

cercice 11

```
from random import randint

nb = randint(1,10)
essai = 1

# compare la proposition de l'utilisateur à nb
while not(int(input("Quel nombre? ")) == nb):
    essai += 1
print(essai)
```

Code 7 – Deviner - première version

Remarque

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

Evereice 1

xercice 2

F.....

Exercice 6

xercice /

rercice 8

xercice 9

Exercice 10

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
 - cico S
- 3 Exercice :
- 4 Exercice 4
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice 7
- 8. Exercice 8

ercice 1

ercice 2

rcice 4

kercice 5

Exercice 7

ercice 8

ercice 9

ercice 10

ercice 11

cice 11

- ▶ 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*.

xercice 2

Exercice 5

Exercice 6

Exercice 7

ercice o

. 10

kercice 10

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

Code 8 - Durée

ercice 1

kercice 2

........

xercice 5

Exercice 7

xercice 8

vorcico 0

xercice 10

vercice 11

```
TVELCICE I
```

_

exercice 5

Exercice 7

exercice 8

exercice 9

exercice 10

Europeine 11

Exercice 11

```
if heures < 10:
    heures = "0"+str(heures)
if minutes < 10:
    minutes = "0"+str(minutes)
if secondes < 10:
    secondes = "0"+str(secondes)
print("{}h {}min {}s".format(heures, minutes, secondes))</pre>
```

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.

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
 - rcice 2
- 3 Exercice 3
- 4 Evercice
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice 7
- 8. Exercice 8

ercice 1

rcice 2

ercice 4

ercice 5

Exercice 8

ercice 9

ercice 9

ercice 10

rcice 11

rcice 11

ice 11

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 (>=3.6)). Il est possible d'écrire :

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

Exercice 1

ercice 2

Exercice 5

LACICICE U

Exercice 8

_XCICICC 0

33 / 44

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
 - cice 2
- 3 Exercice 3
- 1 Eversion
- 5. Exercice 5
- 6 Exercice (
- 7 Evercice
- 8. Exercice 8

rcice 1

rcice 3

ercice 5

rcice 7

rcice 8

Exercice 9

ercice 10

ercice 11

ercice 11

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

Code 11 - Compte à rebours

Exercices constructions élémentaires Éléments de correction

xercice 2

ercice 8

Exercice 9

Exercice 10

Exercices constructions élémentaires Éléments de correction

- 1. Exercice
 - cice 2
- 3. Exercice 3
- 4 Evercice
- 5. Exercice 5
- 6 Exercice
- 7 Exercice 7
- 8. Exercice

ercice 1

rcice 4

rcice 5

cice /

ice o

raina 10

Exercice 10

rcice 11

ercice 11

```
for i in range(2,25,2):
    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

xercice 2

xercice 5

xercice 8

kercice 9

Exercice 10

LXCICICC II

Exercices constructions élémentaires Éléments de correction

- 1. Exercice
 - xercice 2
- 3. Exercice 3
- 4 Evercice
- 5. Exercice 5
- 6. Exercice
- 7 Exercice
- 8. Exercice 8

ercice 1

rcice 3

ercice 4

rcice 7

rcice 8

rcice 9

ercice 10

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.

- . .

kercice 2

Exercice 5

xercice /

. . . .

xercice 10

Exercice 11

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

Code 14 - Moyenne

Remarque

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

Exercice 1

ercice 2

Aci cice 3

ACICICE 4

LXercice 5

.......

ACICICE 5

EXCICICE 10

Exercice 11

```
if moyenne >= 15:
    print("{}/20, félicitations!".format(moyenne)
    )

elif moyenne >= 10:
    # il est inutile ici de vérifier si moyenne <
    15
    print("{}/20, bon travail!".format(moyenne))
else:
    print("{}/20, doit fournir des efforts!".
    format(moyenne))</pre>
```

Code 15 – Affichage

ercice 1

Exercice 4

Exercice 5

.....

vorcico 10

Exercice 11

- 2 Evercice 2
- 3. Exercice 3
- 4 Evereice
- 5 Evercice 5
- 6 Evercice
- 7 Evercice
- 8. Exercice 8

Exercices constructions élémentaires Éléments de correction

kercice 1

ercice 2

ercice 4

rcice 6

cice 8

cice 9

cice 10

rcice 11

```
Exercices
constructions
élémentaires
Éléments de
correction
```

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

Code 16 – Devinette

exercice 1

xercice 2

........

xercice 4

. -

ACICICC 10

exercice 11

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