

L Codent L Créent

Séance 3

C. HINARD M. MIALLIER T. PRÉVOST

ENSTA Bretagne

Collège Croas ar Pennoc

17 mai 2022

- 1 Rappels
 - Boucles
 - Fonctions
- 2 On dessine !
 - Bonjour turtle !
 - À vous !
 - Pour finir

Plan

- 1 Rappels
 - Boucles
 - Fonctions
- 2 On dessine !
 - Bonjour turtle !
 - À vous !
 - Pour finir

1 Rappels

- Boucles
- Fonctions

2 On dessine !

- Bonjour turtle !
- À vous !
- Pour finir

Quelle boucle ?

- Je sais combien de fois je dois exécuter la boucle \implies for

Quelle boucle ?

- Je sais combien de fois je dois exécuter la boucle \Rightarrow for

```
for nombre in range(1,10):  
    print(nombre * 6)
```

Quelle boucle ?

- Je sais combien de fois je dois exécuter la boucle \Rightarrow for

```
for nombre in range(1,10):  
    print(nombre * 6)
```

- Je ne sais pas combien de fois je dois exécuter la boucle \Rightarrow while

Quelle boucle ?

- Je sais combien de fois je dois exécuter la boucle \Rightarrow for

```
for nombre in range(1,10):  
    print(nombre * 6)
```

- Je ne sais pas combien de fois je dois exécuter la boucle \Rightarrow while

```
while 3 * nombre < 1000:  
    nombre = nombre + 1
```


1 Rappels

- Boucles
- Fonctions

2 On dessine !

- Bonjour turtle !
- À vous !
- Pour finir

Le triple

```
def triple(x):  
    return ...
```

Le triple

```
def triple(x):  
    return 3 * x
```

Le triple

```
def triple(x):  
    return 3 * x  
x = triple(3)  
print(x)
```

Liste des pairs

```
def nombres_pairs(liste):  
    liste_pairs = ...  
    for nombre in ...:  
        if ...  
            liste_pairs.append(...)  
    return liste_pairs
```

Liste des pairs

```
def nombres_pairs(liste):  
    liste_pairs = []  
    for nombre in liste:  
        if ...  
            liste_pairs.append(...)  
    return liste_pairs
```

Liste des pairs

```
def nombres_pairs(liste):  
    liste_pairs = []  
    for nombre in liste:  
        if nombre % 2 == 0:  
            liste_pairs.append(nombre)  
    return liste_pairs
```

Première et dernière lettre

```
def premiere_derniere_lettre(mot):  
    premiere = ...  
    derniere = ...  
    return ...
```


Première et dernière lettre

```
def premiere_derniere_lettre(mot):  
    premiere = mot[0]  
    derniere = ...  
    return ...
```

Première et dernière lettre

```
def premiere_derniere_lettre(mot):  
    premiere = mot[0]  
    derniere = mot[-1]  
    return ...
```

Première et dernière lettre

```
def premiere_derniere_lettre(mot):  
    premiere = mot[0]  
    derniere = mot[-1]  
    return premiere + "... " + derniere
```

Plan

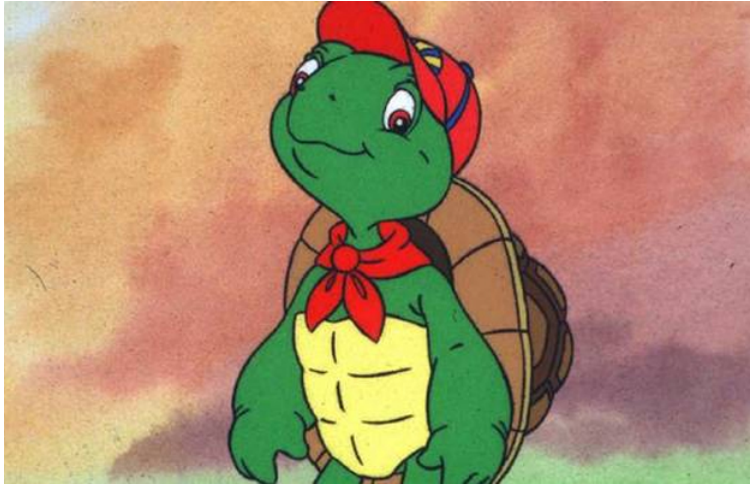
- 1 Rappels
 - Boucles
 - Fonctions
- 2 On dessine !
 - Bonjour turtle !
 - À vous !
 - Pour finir

Plan

- 1 Rappels
 - Boucles
 - Fonctions
- 2 On dessine !
 - Bonjour turtle !
 - À vous !
 - Pour finir

Le module turtle

```
import turtle as t
```



Création du fichier

```
import turtle as t  
  
t.shape("turtle")  
t.exitonclick()
```

Remarque

Sans l'instruction `import turtle as t`, Python ne sait pas que l'on veut utiliser le module `turtle`, donc ça ne marchera pas !

Plan

- 1 Rappels
 - Boucles
 - Fonctions
- 2 On dessine !
 - Bonjour turtle !
 - À vous !
 - Pour finir


```
def carre(x):  
    t.pendown()  
    for ... in range(4):  
        t.forward(...)  
        t.left(...)  
    t.penup()
```

```
def carre(x):  
    t.pendown()  
    for i in range(4):  
        t.forward(x)  
        t.left(90)  
    t.penup()
```

```
def maison():
```

La maison

```
def maison():  
    # les murs  
    carre(80)
```

La maison

```
def maison():  
    # les murs  
    carre(80)  
  
    # on va à l'emplacement de la porte  
    t.forward(25)
```

La maison

```
def maison():  
    # les murs  
    carre(80)  
  
    # on va à l'emplacement de la porte  
    t.forward(25)  
  
    # on dessine la porte  
    carre(30)
```

La maison

```
def maison():  
    # les murs  
    carre(80)  
  
    # on va à l'emplacement de la porte  
    t.forward(25)  
  
    # on dessine la porte  
    carre(30)  
  
    # on va à la position du début du toit  
    t.forward(25)  
    t.left(90)  
    t.forward(80)
```

Maison (suite et fin)

```
# on dessine le toit (triangle équilatéral)  
t.pendown()
```


Maison (suite et fin)

```
# on dessine le toit (triangle équilatéral)  
t.pendown()  
t.left(30)  
t.forward(80)
```

Maison (suite et fin)

```
# on dessine le toit (triangle équilatéral)  
t.pendown()  
t.left(30)  
t.forward(80)  
t.left(120)  
t.forward(80)
```

Maison (suite et fin)

```
# on dessine le toit (triangle équilatéral)  
t.pendown()  
t.left(30)  
t.forward(80)  
t.left(120)  
t.forward(80)  
t.penup()
```

Maison (suite et fin)

```
# on dessine le toit (triangle équilatéral)  
t.pendown()  
t.left(30)  
t.forward(80)  
t.left(120)  
t.forward(80)  
t.penup()  
t.home()
```

Le village

```
def village(nb_maisons):  
    x, y = 0, 0  
    t.home()
```

Le village

```
def village(nb_maisons):  
    x, y = 0, 0  
    t.home()  
    for n in range(nb_maisons):
```

Le village

```
def village(nb_maisons):  
    x, y = 0, 0  
    t.home()  
    for n in range(nb_maisons):  
        maison()
```

Le village

```
def village(nb_maisons):  
    x, y = 0, 0  
    t.home()  
    for n in range(nb_maisons):  
        maison()  
        x += 90
```


Le village

```
def village(nb_maisons):  
    x, y = 0, 0  
    t.home()  
    for n in range(nb_maisons):  
        maison()  
        x += 90  
        t.setpos(x, 0)
```

Le village

```
def village(nb_maisons):  
    x, y = 0, 0  
    t.home()  
    for n in range(nb_maisons):  
        maison()  
        x += 90  
        t.setpos(x, 0)  
village(3)
```

Plan

- 1 Rappels
 - Boucles
 - Fonctions
- 2 On dessine !
 - Bonjour turtle !
 - À vous !
 - Pour finir

Petit dessin pour finir

```
t.penup()  
t.setpos(0,0)  
for i in range(0, 361, 5):  
    t.pendown()  
    t.circle(80)  
    t.penup()  
    t.circle(200, i)
```