

# Pierre Feuille Ciseaux Fonction

Christophe Viroulaud

Première - NSI

**Lang 05**

Le jeu *pierre - feuille - ciseaux* est un jeu effectué avec les mains et opposant deux joueurs.

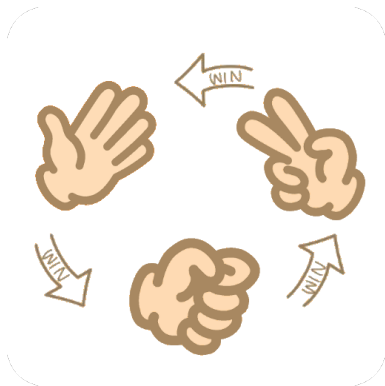


FIGURE 1 – Chaque flèche va dans le sens du gagnant.

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques

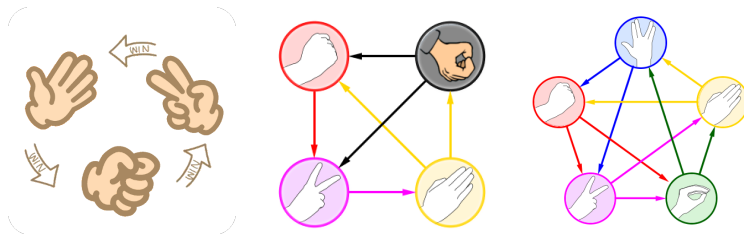


FIGURE 2 – Il existe plusieurs variantes de ce jeu.

Quel concept mettre en place pour implémenter le jeu de manière lisible ?

## 1. Notion de fonction

### 1.1 Fonction mathématique

### 1.2 Fonction en programmation

## 2. Pierre feuille ciseaux

## 3. Bonnes pratiques

### Notion de fonction

Fonction mathématique

Fonction en programmation

### Pierre feuille ciseaux

Création de la fonction

Variante

Variables locales

### Bonnes pratiques

On peut voir une fonction comme une boîte noire qui :

- accepte éventuellement un **paramètre**,
- **renvoie** une valeur.

$$f(x) = x^3$$

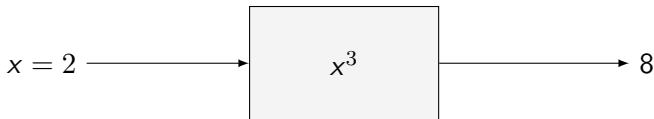


FIGURE 3 – paramètre :  $x = 2$  ; valeur renvoyée : 8

## 1. Notion de fonction

### 1.1 Fonction mathématique

### 1.2 Fonction en programmation

## 2. Pierre feuille ciseaux

## 3. Bonnes pratiques

Notion de fonction

Fonction mathématique

**Fonction en programmation**

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques

```
1 def cube(x):  
2     valeur = x**3  
3     return valeur
```

Code 1 – Création d'une fonction

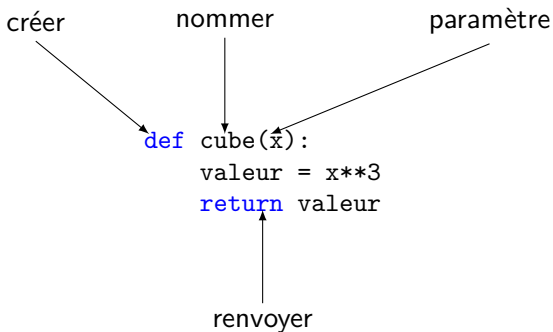


FIGURE 4 – Création de la *boîte*.

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques



```
1 >>> cube(2)
2 8
```

Code 2 – Appel de la fonction

## 1. Notion de fonction

## 2. Pierre feuille ciseaux

### 2.1 Création de la fonction

### 2.2 Variante

### 2.3 Variables locales

## 3. Bonnes pratiques

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques

Pour jouer contre l'utilisateur, la machine doit choisir au hasard parmi les propositions : *pierre*, *feuille*, *ciseaux*.

**Activité 1** : Écrire la fonction `choix_machine()` qui renvoie aléatoirement un des trois mots : *pierre*, *feuille*, *ciseaux*.

```
1 def choix_machine():
2     valeur = randint(1, 3)
3     if valeur == 1:
4         choix = "pierre"
5     elif valeur == 2:
6         choix = "feuille"
7     else:
8         choix = "ciseaux"
9     return choix
```

Code 3 – Création de la fonction

```
1 # la variable 'machine' contient un des trois
   mots
2 machine = choix_machine()
```

Code 4 – Appel de la fonction

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques

## 1. Notion de fonction

## 2. Pierre feuille ciseaux

### 2.1 Création de la fonction

### 2.2 Variante

### 2.3 Variables locales

## 3. Bonnes pratiques

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

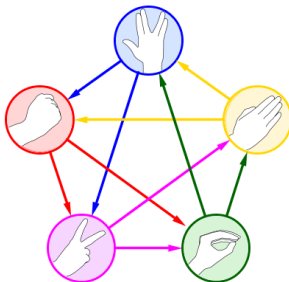
**Variante**

Variables locales

Bonnes pratiques



FIGURE 5 – Pierre feuille ciseaux lézard Spock présenté par Sheldon



Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

**Variante**

Variables locales

Bonnes pratiques

L'utilisateur peut connaître et donc décider de jouer à la variante du jeu contre la machine.

**Activité 2 :** Modifier la fonction telle que sa *signature* s'écrive `choix_machine(variante)`. Le *paramètre* `variante` sera un booléen. Si l'*argument* passé lors de l'appel de la fonction est :

- ▶ **True** : la fonction doit renvoyer un mot parmi les cinq de la variante,
- ▶ **False** : la fonction doit renvoyer un mot parmi les trois de la version classique.

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques

```
1 # La fonction possède un paramètre
2 def choix_machine(variante):
3     if variante:
4         valeur = randint(1, 5)
5     else:
6         valeur = randint(1, 3)
7
8     if valeur == 1:
9         choix = "pierre"
10    elif valeur == 2:
11        choix = "feuille"
12    elif valeur == 3:
13        choix = "ciseaux"
14    elif valeur == 4:
15        choix = "lézard"
16    else:
17        choix = "Spock"
18    return choix
```

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques



```
1 # Lors de l'appel de la fonction on lui passe  
   un argument  
2 machine = choix_machine(True)
```

Code 5 – Appel de la fonction pour jouer à la variante.

## 1. Notion de fonction

## 2. Pierre feuille ciseaux

### 2.1 Création de la fonction

### 2.2 Variante

### 2.3 Variables locales

## 3. Bonnes pratiques

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

**Variables locales**

Bonnes pratiques

```
1 def choix_machine():
2     valeur = randint(1, 3)
3     if valeur == 1:
4         choix = "pierre"
5     elif valeur == 2:
6         choix = "feuille"
7     else:
8         choix = "ciseaux"
9     return choix
```

Visualisation

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques

- Les variables `valeur` et `choix` sont **locales**.

- ▶ Les variables `valeur` et `choix` sont **locales**.
- ▶ La fonction ne renvoie pas la variable `choix` mais le **contenu de cette variable**.

1. Notion de fonction
2. Pierre feuille ciseaux
3. Bonnes pratiques

## Notion de fonction

Fonction mathématique

Fonction en programmation

## Pierre feuille ciseaux

Création de la fonction

Variante

Variables locales

## Bonnes pratiques

```
1 def choix_machine(variante: bool) -> str:
2     if variante:
3         valeur = randint(1, 5)
4     else:
5         valeur = randint(1, 3)
6
7     if valeur == 1:
8         choix = "pierre"
9     elif valeur == 2:
10        choix = "feuille"
11    elif valeur == 3:
12        choix = "ciseaux"
13    elif valeur == 4:
14        choix = "lézard"
15    else:
16        choix = "Spock"
17    return choix
```

Notion de fonction

Fonction mathématique

Fonction en programmation

Pierre feuille  
ciseaux

Création de la fonction

Variante

Variables locales

Bonnes pratiques

## À retenir

On s'attachera à fournir un maximum d'informations  
dans la **signature** d'une fonction.

```
choix_machine(variante: bool) → str
```



```
1 def choix_machine(variante: bool) -> str:
2     """
3     Donne la proposition de la machine.
4
5     Args:
6         variante (booléen): choix de la
7         variante
8
9     Returns:
10        str: renvoie le choix de la machine
11    """
12    if variante:
13        ...
```

## À retenir

On fournira une **docstring** à chaque fonction.