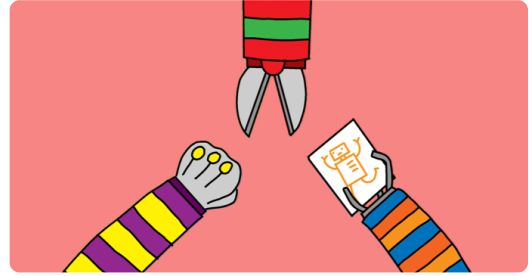




## Chifoumi - Pierre-feuille-ciseaux

Créer ton propre jeu 'Pierre-feuille-ciseaux'.



### Étape 1 Introduction

Dans ce projet, tu vas créer un jeu de Pierre, Feuille, Ciseaux et jouer contre un ordinateur.

Règles: l'ordinateur et toi allez choisir entre pierre feuille et ciseaux. Le gagnant est désigné selon ces règles:

- La pierre brise les ciseaux
- Le papier couvre la pierre
- Les ciseaux coupent le papier

```
pierre (p), feuille (f)  
ou ciseaux (c) ? f  
___ vs >8  
L'ordinateur gagne!
```

#### Informations complémentaires pour les responsables de club

Si tu as besoin d'imprimer ce projet, merci d'utiliser la **Version imprimable** (<https://projects.raspberrypi.org/fr-FR/projects/rock-paper-scissors/print>).



## Notes pour le responsable de club

### Introduction:

Dans ce projet, les enfants apprendront comment faire un programme qui joue au Pierre, Feuille, Ciseaux en utilisant des variables et des instructions conditionnelles.

### Ressources en-ligne

**Ce projet utilise Python 3.** Nous recommandons l'utilisation de **Trinket** (<https://trinket.io/>) pour écrire du code Python en-ligne. Ce projet contient les Trinkets suivants:

- **Ressources 'Pierre, Feuille, Ciseaux' – jumpto.cc/rps-go** (<http://jumpto.cc/rps-go>)

Il existe aussi ce trinket qui contient le projet complété:

- **'Pierre, Feuille, Ciseaux' fini – trinket.io/python/2ab04d9f24** (<https://trinket.io/python/2ab04d9f24>)

### Ressources hors-ligne

Ce projet peut être **complété hors-ligne** (<https://www.codeclubprojects.org/en-GB/resources/python-working-offline/>) si désiré. Vous pouvez accéder aux ressources du projet en cliquant sur le lien "Matériaux du projet". Ce lien contient une section "Ressources du projet" qui inclut les ressources dont les enfants auront besoin pour compléter le projet hors-ligne. Assurez-vous que les enfants ont accès à une copie de ces ressources. Cette section inclut les fichiers suivants:

- rock-paper-scissors/rock-paper-scissors.py

Vous pouvez aussi trouver une version complétée du projet dans la section "Ressources du bénévole" qui contient:

- rock-paper-scissors-finished/rock-paper-scissors.py

(Toutes les ressources ci-dessus peuvent aussi être téléchargées dans les fichiers `.zip` projet et bénévole)

### Objectifs d'apprentissage

- Variables;
- Sélection (`if`, `elif`, `else`);
- Booléen `==` et `and`.

Ce projet couvre les éléments suivants du **Programme Raspberry Pi de Créativité Numérique** (<http://rpf.io/curriculum>):

- **Utiliser les constructs basiques de programmation pour élaborer un programme simple.** (<https://www.raspberrypi.org/curriculum/programming/creator>)

### Défis

- "Art ASCII" - Utiliser la logique conditionnelle pour afficher de l'art ASCII pour la pierre, la feuille et les ciseaux.
- "Créer un nouveau jeu" - Dupliquer le jeu du Pierre, Feuille, Ciseaux et le remixer pour créer un nouveau jeu.



## Matériel pour projet

### Ressources du projet

- Fichier .zip contenant toutes les ressources du projet (<https://projects-static.raspberrypi.org/projects/rock-paper-scissors/05a38c3b2fa6deb8fe4ccd505b5e7c65bda28e2f/fr-FR/resources/rock-paper-scissors-project-resources.zip>)
- Trinket en-ligne contenant des ressources du projet "Pierre, Feuille, Ciseaux" (<http://jumpto.cc/rps-go>)
- rock-paper-scissors/rock-paper-scissors.py (<https://projects-static.raspberrypi.org/projects/rock-paper-scissors/05a38c3b2fa6deb8fe4ccd505b5e7c65bda28e2f/fr-FR/resources/rock-paper-scissors-rock-paper-scissors.py>)

### Ressources pour le responsable de club

- Fichier .zip contenant toutes les ressources du projet (<https://projects-static.raspberrypi.org/projects/rock-paper-scissors/05a38c3b2fa6deb8fe4ccd505b5e7c65bda28e2f/fr-FR/resources/rock-paper-scissors-volunteer-resources.zip>)
- Trinket Python en-ligne vierge (<https://trinket.io/python/2ab04d9f24>)
- rock-paper-scissors-finished/rock-paper-scissors.py (<https://projects-static.raspberrypi.org/projects/rock-paper-scissors/05a38c3b2fa6deb8fe4ccd505b5e7c65bda28e2f/fr-FR/resources/rock-paper-scissors-finished-rock-paper-scissors.py>)

## Étape 2 Le tour du joueur

---

Tout d'abord, laissons le joueur choisir entre Pierre, Feuille ou Ciseaux.

- Ouvre ce trinket: [jumpto.cc/rps-go](http://jumpto.cc/rps-go) (<http://jumpto.cc/rps-go>).
- Le projet contient déjà le code pour importer une fonction que tu vas utiliser dans ce projet.

```
main.py
#!/bin/python3
from random import randint
```

Tu utiliseras plus tard la fonction `randint` pour générer des nombres aléatoires.

- Laissons le joueur choisir Pierre, Feuille ou Ciseaux en tapant "p", "f", "c".

```
from random import randint
joueur = input('pierre (p), feuille (f) ou ciseaux (c) ?')
```



- Maintenant affiche ce que le joueur a choisi:

```
from random import randint
joueur = input('pierre (p), feuille (f) ou ciseaux (c) ?')
print(joueur, '\n')
```



- Teste maintenant ton code en cliquant sur **Run**. Clique dans la fenêtre de sortie du Trinket et entre ton choix.

## Étape 3 Le tour de l'ordinateur

Maintenant c'est au tour de l'ordinateur. Tu peux utiliser la fonction `randint` pour générer un nombre aléatoire pour choisir entre pierre, feuille et ciseaux.

- Utilise `randint` pour générer un nombre aléatoire et choisir si l'ordinateur a choisi Pierre, Feuille ou ciseaux.
- Exécute plusieurs fois le script (tu devras entrer plusieurs fois "p", "f" ou "c" à chaque fois.)

Tu devrais voir que "choisi" est défini aléatoirement sur 1, 2 ou 3.

Disons:

- 1= pierre (p)
- 2= feuille (f)
- 3= ciseaux (c)

Utilise `if` pour vérifier que le nombre choisi est 1 (`==` est utilisé pour voir si deux choses sont identiques).

```
choisi = randint(1,3)
print(choisi)
if choisi == 1:
```

*N'oublies pas les deux points ":"*

- Python utilise **l'indentation** (bouger le code à droite) pour montrer quel code est à l'intérieur de l'instruction `if`. Tu peux à la fois utiliser deux espaces (tapes deux fois sur la barre d'espace) ou taper sur la **touche tabulation** (généralement au dessus de la touche VERR MAJ sur le clavier.)

Paramètre `ordinateur` sur "p" à l'intérieur de l'instruction `if` en utilisant l'indentation:

```
if choisi == 1:
    ordinateur = 'p'
```

*Deux espaces ou une "tabulation"*

- Tu peux ajouter une autre vérification en utilisant `elif` (abréviation de *else if*, sinon si):

```
if choisi == 1:
    ordinateur = 'p'

elif choisi == 2:
    ordinateur = 'f'
```

*Indentation avec deux espaces ou une tabulation*

Cette condition ne sera seulement vérifiée que si la première condition échoue (si l'ordinateur ne choisi pas 1)

- Et enfin, si l'ordinateur ne choisi pas 1 ou 2 alors il doit avoir choisi 3.

Cette fois nous pouvons juste utiliser `else` qui signifie autrement.

```
if choisi == 1:
    ordinateur = 'p'

elif choisi == 2:
    ordinateur = 'f'

else:
    ordinateur = 'c'
```

- Maintenant, à la place d'afficher le nombre aléatoire que l'ordinateur choisi tu peux imprimer la lettre.

```
choisi = randint(1,3)
#print(choisi)
if choisi == 1:
    ordinateur = 'p'
elif choisi == 2:
    ordinateur = 'f'
else:
    ordinateur = 'c'
print(ordinateur)
```

Commente la ligne en mettant un '#' au début

```
pierre (p), feuille (f) ou
ciseaux (c) ? s
s vs
p
```

Tu peux soit supprimer la ligne `print(choisi)`, ou faire en sorte que l'ordinateur l'ignore en ajoutant un `#` au début de la ligne.

- Teste ton code en cliquant sur Run et en choisissant tes options.
- Hmm, le choix de l'ordinateur est imprimé sur une nouvelle ligne. Tu peux choisir de corriger cela en ajoutant `end= ' '` après `vs`, qui va dire à Python de finir avec un espace au lieu d'une nouvelle ligne.

```
print(joueur, 'vs', end=' ')
choisi = randint(1,3)
#print(choisi)
```

```
pierre (p), feuille (f) ou
ciseaux (c) ? c
c vs p
```

- Joue au jeu quelques fois en cliquant sur Run et en faisant un choix.

Pour l'instant tu devras déterminer toi-même qui a gagné. Ensuite tu ajouteras du code Python pour résoudre ce problème.

## Étape 4 Vérifier le résultat

Maintenant, ajoutons le code pour voir qui a gagné.

- Tu devras comparer les variables `joueur` et `ordinateur` pour voir qui a gagné.

S'ils sont identiques, alors on procède à un tirage au sort:

```
print(ordinateur)
if joueur == ordinateur:
    print("TIRAGE AU SORT !")
```

pierre (p), feuille (f)  
ou ciseaux (c) ? c  
c vs c  
TIRAGE AU SORT !

- Teste ton code en jouant plusieurs fois le jeu jusqu'à obtenir un tirage au sort.

Tu devras cliquer plusieurs fois sur `Run` pour lancer un nouveau jeu.

- Maintenant regardons les cas où le joueur a choisi "p" (Pierre) mais pas l'ordinateur.

Si l'ordinateur choisi "c" (ciseaux) alors le joueur gagne (la pierre bat les ciseaux).

Si l'ordinateur choisi "f" (feuille) alors l'ordinateur gagne (le papier bat la pierre).

Nous pouvons vérifier le choix du joueur *et* le choix de l'ordinateur en utilisant `et`.

```
if joueur == ordinateur:
    print("TIRAGE AU SORT !")
elif joueur == 'p' and ordinateur == 'c':
    print('Le joueur gagne!')
elif joueur == 'p' and ordinateur == 'f':
    print("L'ordinateur gagne!")
```

- Ensuite regardons les cas où le joueur a choisi "f" (feuille) mais pas l'ordinateur:

```
elif joueur == 'p' and ordinateur == 'c':
    print('Le joueur gagne!')
elif joueur == 'p' and ordinateur == 'f':
    print("L'ordinateur gagne!")
elif joueur == 'f' and ordinateur == 'p':
    print('Le joueur gagne!')
elif joueur == 'f' and ordinateur == 'c':
    print("L'ordinateur gagne!")
```

- Et enfin tu peux ajouter le code pour déterminer le gagnant quand le joueur choisi "c" (ciseaux) et l'ordinateur choisi pierre ou feuille?
- Maintenant joue au jeux pour vérifier ton code.

```
pierre (p), feuille (f)
ou ciseaux (c) ? c
c vs f
Le joueur gagne!
```

Clique sur **Run** pour lancer un nouveau jeu.



## Étape 5 Défi: L'art ASCII

Au lieu d'utiliser les lettres p, f et c pour représenter pierre, feuille et ciseaux, peux-tu utiliser l'art ASCII?

Par exemple:

```
pierre (p), feuille (f)
ou ciseaux (c) ? f
___ vs >8
L'ordinateur gagne!
```

Où:

```
pierre: 0
feuille: ___
ciseaux: >8
```

- Au lieu de dire `print ordinateur` tu devras ajouter une nouvelle ligne pour chaque option dans le `if` pour afficher le bon art ASCII.

Astuces:

```
if choisi == 1:
    ordinateur = 'p'
    print('0') ← Art ASCII pour la pierre

#print(ordinateur)
```

- Au lieu de dire `print joueur` tu devras ajouter une nouvelle instruction conditionnelle `if` pour vérifier quel issue le joueur a choisi et afficher le bon art ASCII:

Indice:

```
joueur = input('pierre (p), feuille (f) ou ciseaux (c) ?')

if joueur == 'p':
    print('0', 'vs', end=' ')
```

Rappelle-toi qu'en ajoutant `end=' '` à la fin des paramètres de la fonction `print` la fera se terminer par un espace à la place d'une nouvelle ligne.

## Étape 6 Défi: Créer un nouveau jeu

---

Peux tu créer un nouveau jeu comme le Pierre, Feuille, Ciseaux avec différents objets?

Clique sur le bouton "Dupliquer" pour créer une copie de ton projet Pierre, Feuille, Ciseaux.

Par exemple, utilise Feu, Arbres et Eau:

---

Ce projet a été traduit par des bénévoles:

Simon Gleises

Mathias Pierre

Grâce aux bénévoles, nous pouvons donner aux gens du monde entier la chance d'apprendre dans leur propre langue. Vous pouvez nous aider à atteindre plus de personnes en vous portant volontaire pour la traduction - plus d'informations sur [rpf.io/translate](https://rpf.io/translate) (<https://rpf.io/translate>).

---

Publié par [Raspberry Pi Foundation](https://www.raspberrypi.org) (<https://www.raspberrypi.org>) sous un [Creative Commons license](https://creativecommons.org/licenses/by-sa/4.0/) (<https://creativecommons.org/licenses/by-sa/4.0/>).

**Voir le projet et la licence sur GitHub** (<https://github.com/RaspberryPiLearning/rock-paper-scissors>).