

Nom :	Prénom :	Classe :
-------	----------	----------

SNT — Programmation (TP) - Initiation, avec Pyrates

Objectifs :

- Découvrir et savoir utiliser la syntaxe d'un langage de programmation (Python).

Introduction

Savoir programmer permet de **comprendre** et de **créer** les **outils numériques** qui façonnent notre monde, plutôt que de simplement les utiliser. Cela développe aussi la **logique**, la **créativité** et la **capacité à résoudre des problèmes** de manière structurée.

Agriculture, technologies de l'information, santé et services médicaux, éducation et formation, commerce, construction, tourisme, finance, énergie ... Programmer est un savoir-faire transversal : on le retrouve désormais dans tous les domaines.

Pyrates

« Pyrates » est un **jeu éducatif** avec lequel nous allons découvrir quelques bases de la programmation.

Pour commencer, utilisez votre **tablette** ou un **ordinateur** et **ouvrez la page Web** suivante :

<https://py-rates.fr>

Choisissez votre personnage puis cliquez sur le bouton « Je commence! »

ou saisissez un code secret puis cliquez sur « Je reprends! » si vous avez déjà joué à Pyrates.

Lors de votre première partie, un code secret (identifiant) vous est communiqué. **Notez-le**, par exemple ici ↓

Ce code vous permet de reprendre le jeu au dernier niveau atteint.

Première partie

Lorsque vous débutez, il est conseillé de cliquer sur le bouton vert « Guide de démarrage », situé sous le logo.

Lisez attentivement ce guide pour comprendre ce que vous devez faire.

Généralement, votre **objectif** est de faire déplacer votre personnage jusqu'à la clé, puis jusqu'au trésor.

Mais cet objectif peut varier, selon les niveaux — Il existe huit niveaux différents.

Quelques recommandations :

- Lisez l'**objectif** du niveau, il est indiqué sous le logo, en haut à gauche de l'écran.
Ex. « Ramasser la clé puis ouvrir le coffre. »
- Prenez connaissance des **contraintes** du niveau. Certains niveaux vous empêchent de dépasser un certain nombre de lignes. Ex. « Dans ce niveau votre programme ne doit pas dépasser 10 lignes. »
- Observez les **fonctions de contrôle** du personnage que vous pouvez utiliser.
Ces fonctions changent, selon les niveaux !

Niveau 1

Lorsque nous découvrons un niveau, ayons **toujours le réflexe** de prendre connaissance de l'objectif, des contraintes et des fonctions de contrôle du personnage.

Objectif du niveau	Ramasser la clé puis ouvrir le coffre.
Contraintes	Le programme ne doit pas dépasser 10 lignes.
Fonctions de contrôle du personnage	avancer() gauche() droite() ouvrir()

Voici le code fourni pour ce niveau 1 :

```
1 # Ecrire le code de son programme ci-dessous
2 avancer()
3
```

Pour exécuter ce code, et voir comment notre personnage se déplace, appuyons sur le bouton « Exécuter ».

Zut, le code fourni ne permet que de faire avancer le personnage que d'une case vers l'avant !

En programmation, on dit qu'une ligne de code est une **instruction**.

Et lorsqu'on exécute un programme, tout recommence à zéro, puis les instructions sont exécutées les unes après les autres.



Essayons de modifier ce programme, pour essayer d'atteindre au moins la clé.

Quelles sont les **fonctions de contrôle** du personnage dont nous disposons pour ce niveau ?
avancer(), gauche(), droite(), et ouvrir().

Ok, essayons cela :

```
1 # Ecrire le code de son programme ci-dessous
2 avancer()
3 droite()
4 avancer()
5 gauche()
6 avancer()
7
```

Puis exécutons ce programme (bouton vert « Exécuter »).

Super ! Nous parvenons à déplacer notre personnage jusqu'à la clé !

Il ne reste plus qu'à faire déplacer le personnage jusqu'au trésor :

1	# Ecrire le code de son programme ci-dessous	<p>Tiens, c'est étrange, les lignes supérieures à la ligne 10 sont coloriées en rouge ... Pourquoi ?</p> <p>Relisons les contraintes du niveau : <i>Dans ce niveau votre programme ne doit pas dépasser 10 lignes.</i></p> <p>Voilà notre explication !</p> <p>Mais comment faire pour écrire notre programme en moins de 10 lignes, alors que nous avons besoin de faire avancer notre personnage de 16 cases !?</p>
2	avancer()	
3	droite()	
4	avancer()	
5	gauche()	
6	avancer()	
7	droite()	
8	avancer()	
9	avancer()	
10	avancer()	
11	avancer()	
12	avancer()	
13	avancer()	
14	avancer()	

Mémo programmation

Heureusement le jeu Pyrates intègre un « Mémo programmation » à gauche de l'écran. Ce mémo contient les chapitres suivants :

Titre	Description
Notions de base	Notions essentielles pour débiter
Variable	Garder des informations en mémoire
Conditionnelle	Exécuter selon certaines conditions
Boucle for	Répéter un certain nombre de fois
Boucle while	Répéter selon certaines conditions

Répéter un certain nombre de fois ? Voilà qui peut nous être utile pour ce niveau 1 !

1	# Ecrire le code de son programme ci-dessous
2	avancer()
3	droite()
4	avancer()
5	gauche()
6	avancer()
7	droite()
8	for _ in range(16):
9	avancer()

Ça marche ! Il ne reste plus qu'à ouvrir le coffre, après cette « Boucle for »...

Boucle for

Une boucle for permet d'exécuter une ou plusieurs instructions, un nombre de fois déterminé.

Exemple 1 :

```
1 for _ in range(3):
2     sauter()
3 avancer()
```

Dans l'exemple ci-dessus, l'instruction `sauter()` est exécutée **trois fois**.

Puis, **une fois que la boucle est terminée**, l'instruction `avancer()` est exécutée une fois.

Exemple 2 :

```
1   for _ in range(3):
2       sauter()
3       avancer()
```

Dans l'exemple ci-dessus, l'instruction `sauter()` puis l'instruction `avancer()` sont exécutées **trois fois**.

La différence entre l'exemple 1 et l'exemple 2 réside dans le nombre d'espaces placés devant une instruction.



Lorsqu'une instruction possède **4 espaces** devant elle, elle fait alors parti de la boucle.

Aides supplémentaires pour les différents niveaux de Pyrates

Niveau	Aide
2	Même principe qu'au niveau 1. Et si on utilise une boucle, pourquoi ne pas en utiliser une autre, un peu plus bas dans le code !?
3	Il va falloir garder une information en mémoire... Une variable ! <code>message = lire_nombre()</code> peut être utile, suivi d'un <code>sauter_hauteur(message)</code> ?
4	Cette fois, il faut exécuter une instruction selon une certaine condition . Si le message contient "gau", alors on se tourne à gauche... Sinon, on se tourne à droite ...
5	Si la hauteur est de 1, alors on saute, sinon si la hauteur est de 2, alors on saute_haut, sinon : on avance.
6	
7	
8	

