

Sean McManus

MON COURS VISUEL DE PROGRAMMATION SCRATCH ET PYTHON PAS À PAS

Traduit de l'anglais (Royaume-Uni) par
Christine Eberhardt

DUNOD

L'édition originale de cet ouvrage a été publiée en 2020
au Royaume-Uni par Dorling Kindersley sous le titre
Beginner's Step-By-Step Coding Course

This edition published by arrangement
with Dorling Kindersley Limited,
a Penguin Random House Company.
All rights reserved.

Original Title : *Beginner's Step-by-Step Coding Course*
Copyright © Dorling Kindersley Limited, 2020
DK, One Embassy Gardens, 8 Viaduct Gardens
London SW11 7BW
A Penguin Random House Company

Pour l'édition française :
© Dunod, 2021
11 rue Paul Bert, 92240 Malakoff
www.dunod.com
978-2-10-082801-2

Traduction : Christine Eberhardt

Interdit à la vente au Québec

Code de la propriété intellectuelle n'autorisant, aux termes de l'article 122-5, 2° et 3° a), d'une part, que les « copies ou reproductions strictement servées à l'usage privé du copiste et non destinées à une utilisation collective » d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause est illicite » (art. L. 122-4).

La représentation ou reproduction, par quelque procédé que ce soit, constitue donc une contrefaçon sanctionnée par les articles L. 335-2 et suivants du Code de la propriété intellectuelle.

SOMMAIRE

À propos de ce livre

6

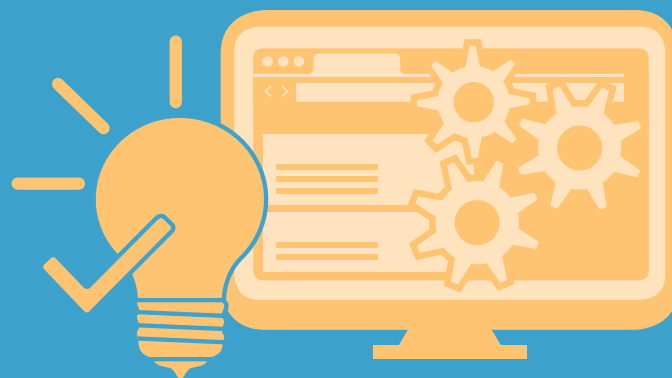
INTRODUCTION

Qu'est-ce que la programmation ?

8

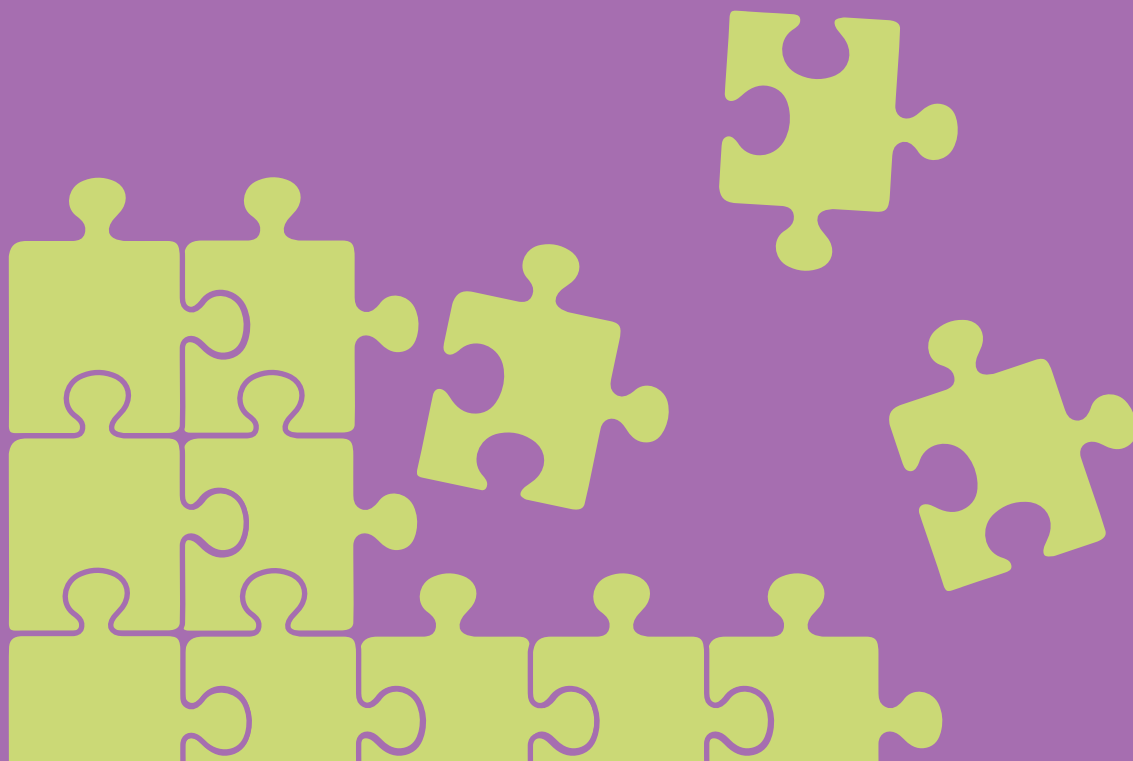
Les langages de programmation

10



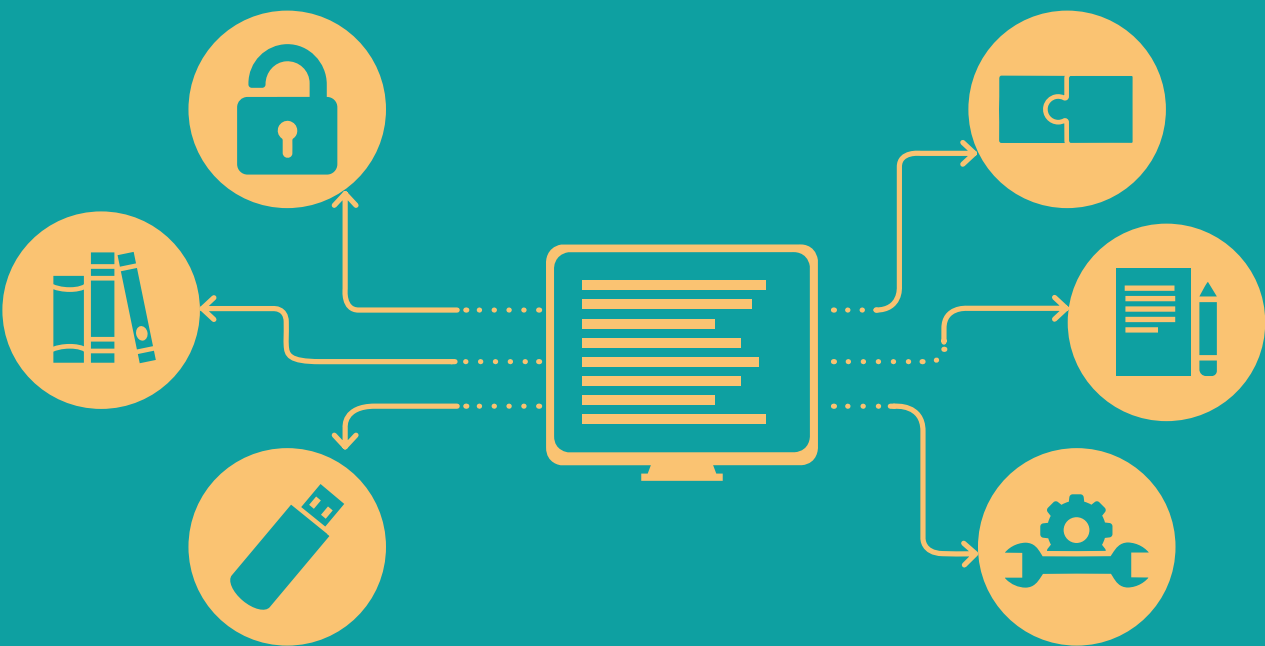
SCRATCH

Qu'est-ce que Scratch ?	16	Logique et décisions	32
L'interface de Scratch	18	Les entrées	34
Les sprites	20	Envoyer des messages	36
Blocs de couleur et scripts	22	Utiliser les fonctions	38
Le déplacement des sprites	24	Traducteur de voyage	40
Apparences et sons	26	Casse-tête	52
Gérer le déroulement du programme	28	Esquive d'astéroïdes	68
Manipuler les données	30		



PYTHON®

Qu'est-ce que Python?	82	Les bibliothèques	104
Installation de Python	84	Générateur d'équipes	106
Utilisation de IDLE	86	Débogage	118
Les variables dans Python	88	Planificateur de projet	122
Les données dans Python	90	Objets et classes	144
Opérateurs logiques et branchements	92	Gestionnaire de budget	146
Entrée et sortie	94	Pygame Zero	164
Les boucles dans Python	96	Knight's Quest	166
Les fonctions	100		



À propos de ce livre

Organisation de l'ouvrage

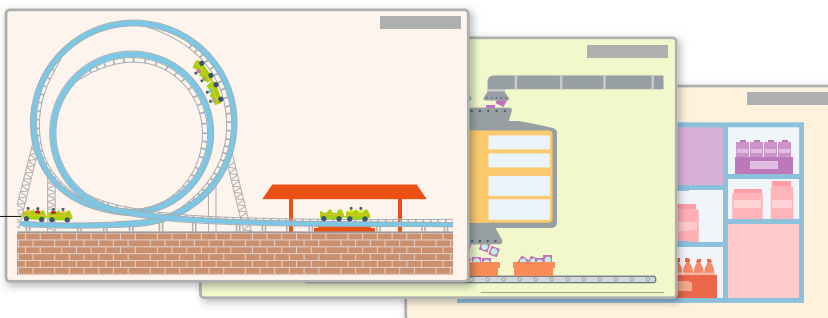
Deux chapitres exposent les principes fondamentaux des deux langages de programmation Scratch et Python. Pour chaque langage les principes de base

sont présentés puis des projets vous sont proposés afin que vous puissiez tester par vous-même les notions abordées.

Concepts

Chaque chapitre présente les concepts de programmation de base du langage à l'aide d'exemples de code pratiques que vous pouvez tester pour mieux comprendre le concept.

Les illustrations vous aident à comprendre les concepts



Projets

Les projets de cet ouvrage vont vous apprendre à créer des jeux, des planificateurs et des applications. Chacun d'eux débute par un rapide aperçu de ce que vous allez apprendre, de comment le planifier et de quoi vous aurez besoin pour le créer. Des instructions simples étape-par-étape vous guident en expliquant chaque caractéristique du code, à l'aide d'annotations détaillées.

Cet encadré présente les concepts utilisés dans le projet

VOUS ALLEZ APPRENDRE



 **Durée:**
1 heure

Lignes de code: 58

Niveau de difficulté


Indique le temps estimé pour réaliser un projet

Indique le nombre de lignes de code estimé pour un projet. Ce nombre varie selon l'éditeur de code utilisé

VOUS ALLEZ APPRENDRE

Indique le niveau de difficulté d'un projet, 1 étant le plus facile

Les projets sont découpés en sections avec des étapes bien définies afin de faciliter l'apprentissage

1.1


1.2


```
disponible = 2500.00  
budgets = {}
```

ÉTAPE PAR ÉTAPE

Fenêtres de code Python

Le code Python est édité dans deux fenêtres différentes ; la fenêtre de commande et l'éditeur. Dans cet ouvrage, des couleurs différencient ces fenêtres à grille afin que vous sachiez dans laquelle vous devez saisir le code.

```
>>> input = 2
>>> score = input * 3
>>> print(score)
6
```

FENÊTRE DE COMMANDE

Ces chevrons n'apparaissent que dans la fenêtre de commande. Saisissez le code après l'invite >>>

```
def reset_jeu():
    global score, charmes
    score = 0
    charmes = 0
```

FENÊTRE DE L'ÉDITEUR

Chaque indentation (espace au début d'une ligne) représente quatre cases de grille vides. Les indentations supplémentaires sont toujours en multiples de quatre cases

Chaque rectangle de la grille représente un espace dans le code

Icône « Enregistrer »

L'icône « Enregistrer » vous rappelle à quel moment vous devez sauvegarder le programme au cours du projet.



ENREGISTRER

Trucs et astuces

La section Trucs et astuces en fin de projet donne quelques astuces pour améliorer le code existant ou ajouter de nouvelles fonctionnalités.



Trucs et astuces



Code sur le site dunod.com

Le pack de ressources des projets de cet ouvrage (à l'exception des sections « Trucs et astuces » et des projets créés avec Scratch) est disponible sur **www.dunod.com**. Vous y trouverez le code au format original (.py) et les images pour tous les jeux.



Rendez-vous à cette adresse pour télécharger le pack de ressources de cet ouvrage

Qu'est-ce que la programmation ?

Les ordinateurs et appareils électroniques ont besoin de logiciels (ou programmes) pour fonctionner. La programmation, ou codage, est l'art d'écrire des instructions. Il existe des programmeurs professionnels mais le codage peut aussi être un passe-temps.

Les programmes informatiques sont partout

La programmation ne concerne plus uniquement les systèmes informatiques conventionnels. Le monde est devenu de plus en plus numérique et presque tout fonctionne avec des logiciels. Les programmes sont désormais intégrés dans des appareils tels que les téléphones portables et les tablettes, les équipements domestiques et même dans les systèmes de transport.

DEVENIR PROGRAMMEUR

Une fois que vous maîtrisez les bases, voici comment développer vos compétences en développement.

- **Pratiquez** : écrivez et testez du code
- **Lisez du code** : vous apprendrez beaucoup en étudiant les programmes des autres
- **Apprenez plusieurs langages** : en étudiant les concepts sur lesquels reposent les autres langages, vous saurez lequel choisir pour votre projet
- **Publiez des projets** : en publiant votre travail, vous bénéficierez des commentaires des autres programmeurs et vous améliorerez votre code

Centre de données

Un centre de données est similaire à une installation informatique à l'échelle industrielle. Ses nombreux serveurs accessibles sur Internet stockent des données ou exécutent un logiciel « dans le cloud ». Le « cloud » est un réseau mondial de serveurs qui fournissent des services via Internet.

Voiture

Un logiciel peut être utilisé pour contrôler la vitesse, la température ou la consommation d'une voiture. Le système de navigation nécessite également des programmes.

Ordinateur de bureau

Les ordinateurs sont largement utilisés dans les bureaux et les foyers pour gérer les activités administratives et créatives comme la musique, le design, le traitement de texte, les comptes, etc.

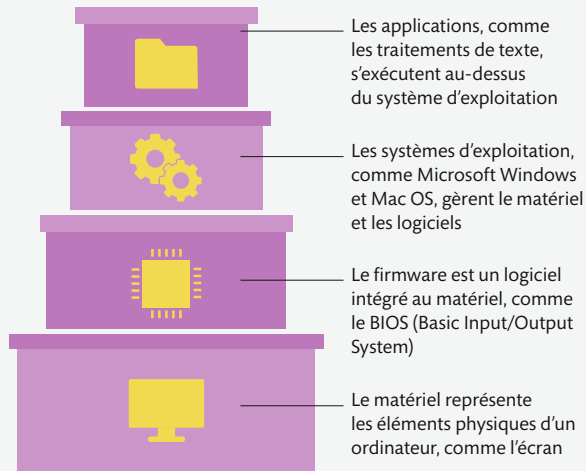
Machine à laver

De nombreux appareils ménagers sont équipés de programmes. Les cycles de lavage, la température de l'eau et les programmations d'une machine à laver, par exemple, sont gérés par un logiciel.



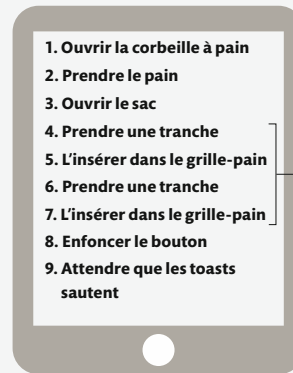
Qu'est-ce qu'un programme ?

Un programme est un ensemble d'instructions qu'un ordinateur suit pour effectuer une tâche. Les programmes peuvent être extrêmement complexes. Microsoft Windows, par exemple, se compose de millions de lignes d'instructions.

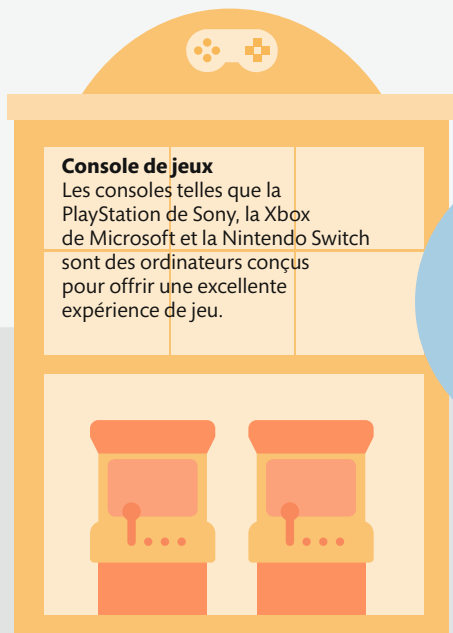


Penser comme l'ordinateur

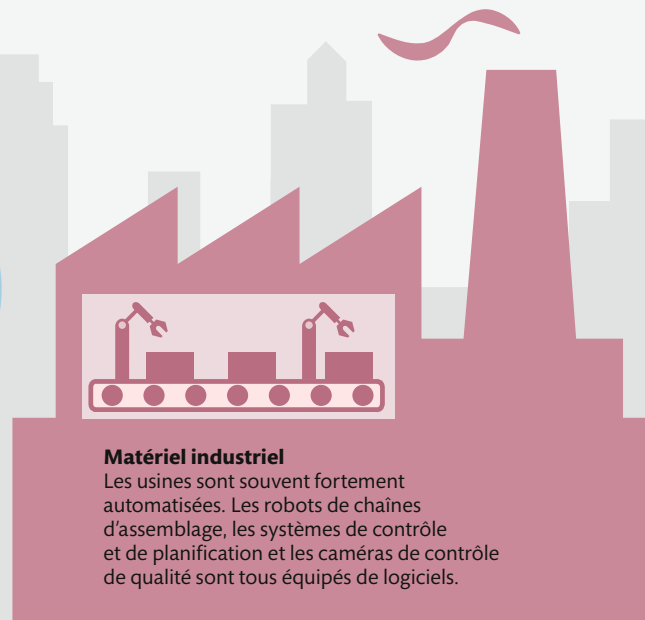
Pour écrire un programme, il faut comprendre comment l'ordinateur traite les instructions. Chaque tâche doit être décomposée. On ne peut pas simplement demander à un robot de « faire griller des toasts », par exemple. Il faut programmer des instructions précises et détaillées pour chaque étape du processus.



Au lieu de répéter deux fois la même instruction, il sera plus rapide et plus simple de dire dans le programme : « Répéter deux fois : prendre la tranche, l'insérer dans le grille-pain »



Appareil photo
Sur les appareils photo d'aujourd'hui, c'est un logiciel qui permet de changer les réglages, de revoir les photos et de les supprimer au besoin.



Les langages de programmation

Un langage de programmation est un ensemble de mots et de symboles qui permet d'écrire des instructions pour un ordinateur. Il faut parfois choisir entre la facilité d'utilisation du langage et sa puissance.

Langages de haut et de bas niveaux

Les langages de programmation de haut niveau n'exigent pas de connaître le matériel informatique. Ils emploient souvent des mots proches du langage humain et gèrent automatiquement certaines fonctions de l'ordinateur. Souvent, un programme écrit dans ce

langage peut s'exécuter sur différents matériels. Les langages de bas niveau offrent un contrôle plus pointu de l'ordinateur, mais ils exigent une compréhension plus approfondie de son fonctionnement. Les programmes écrits dans ces langages peuvent ne pas fonctionner sur d'autres matériels.

LANGAGES DE HAUT NIVEAU

- Écriture assez rapide
- Facile à comprendre
- Rapidité d'exécution acceptable
- Programmes compatibles avec différents matériels
- Aucune connaissance informatique requise

Instruction pour afficher du texte

```
print("Bonjour!")
```

Python

Langage populaire de haut niveau, Python est facile à lire et à écrire. Cette instruction affiche « Bonjour! » sur l'écran.

LANGAGES DE BAS NIVEAU

- Contrôle direct des fonctionnalités matérielles
- Vitesse d'exécution supérieure avec du code bien écrit
- Nécessite une bonne connaissance du matériel
- Programme uniquement compatible avec le même processeur ou un processeur similaire

Nombre hexadécimal, un système d'écriture en base 16 courant en informatique

```
MOV AX, 66H
```

Assembleur

Ce langage est très proche du code machine du processeur. Cet exemple prend un nombre et le place dans l'accumulateur du processeur.

