Nick Morgan

JAVASCRIPT pour les KiDS

Dès 10 ans

no starch press

EYROLLES

JAVASCRIPT pour les KIDS

La programmation est un jeu d'enfants !

JavaScript est le langage de programmation du Web, la sauce secrète qui rend Internet si génial, vos sites web interactifs et vos jeux en ligne si amusants !

JavaScript pour les kids est une joyeuse introduction aux concepts essentiels de la programmation, dans laquelle les exemples sont expliqués étape par étape et accompagnés de drôles d'illustrations. Tu commenceras par les concepts de base, tels que les chaînes de caractères, les tableaux et les boucles. Tu poursuivras avec des sujets plus avancés, tels que l'interactivité avec jQuery et les graphismes dans le canvas. Au passage, tu créeras quelques jeux, tels que « À la recherche du trésor enfoui ! », le jeu du pendu et le jeu du serpent. Et tu verras comment :

- créer des fonctions pour organiser et réutiliser ton code ;
- écrire et modifier ton code HTML afin de rendre tes pages web plus dynamiques ;
- utiliser le DOM et jQuery pour permettre à tes pages web de réagir aux actions de l'utilisateur ;
- utiliser le canvas pour dessiner et animer des graphismes ;
- coder des jeux entièrement contrôlés par l'utilisateur avec détection des collisions et comptage des scores.

Avec des exemples visuels tels que des balles rebondissantes, des abeilles animées et des voitures de course, tu peux voir concrètement les résultats de tes programmes. Chaque chapitre s'appuie sur le précédent. Et les défis de programmation, à la fin de chaque chapitre, sont là pour te dégourdir les neurones et pour insuffler des idées incroyables dans tes programmes.

À propos de l'auteur

Nick Morgan est ingénieur *front end* chez Twitter. Il aime tous les langages de programmation, mais il a un faible pour le JavaScript. Nick vit à San Francisco (la partie embrumée) avec sa fiancée et leur chien peluche, Pancake. Il écrit pour le blog skilldrick.co.uk.

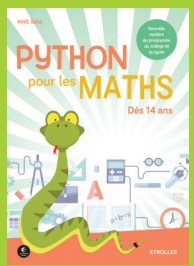
À qui s'adresse ce livre ?

Aux collégiens, lycéens, parents, enseignants et associations.

Sur editions-eyrolles.com/go/js4kids

Télécharge le code source des exemples et les solutions des défis du livre.

Dans la même collection



no starch press

www.editions-eyrolles.com

Nick Morgan

JAVASCRIPT pour les KIDS

no starch press

EYROLLES

ÉDITIONS EYROLLES
61, bd Saint-Germain
75240 Paris Cedex 05
www.editions-eyrolles.com

Traduction autorisée de l'ouvrage en langue anglaise intitulé
JavaScript for kids
de Nick Morgan (ISBN : 9781593274085),
publié par No Starch Press.

All Rights Reserved.

En application de la loi du 11 mars 1957, il est interdit de reproduire intégralement ou partiellement le présent ouvrage, sur quelque support que ce soit, sans l'autorisation de l'Éditeur ou du Centre Français d'exploitation du droit de copie, 20, rue des Grands Augustins, 75006 Paris.

© 2014 by Nick Morgan / No Starch Press pour l'édition en langue anglaise

© Groupe Eyrolles, 2017, pour la présente édition, ISBN : 978-2-212-11850-6

© Traduction française : Myriam Rakho

À PROPOS DE L'AUTEUR

Nick Morgan est ingénieur front-end chez Twitter. Il aime tous les langages de programmation, avec un faible pour le JavaScript. Il vit à San Francisco (la partie embrumée) avec sa fiancée et leur chien Pancake. Il écrit pour le blog skilldrick.co.uk.

À PROPOS DE L'ILLUSTRATEUR

Miran Lipovača est l'auteur de *Learn you a haskell for great good!*. Il aime boxer, jouer de la basse et, bien sûr, dessiner. Il est particulièrement fasciné par les squelettes dansants et le nombre 71. Et quand il passe des portes automatiques, il prétend les ouvrir avec son esprit !

À PROPOS DU RELECTEUR TECHNIQUE

Angus Croll est l'auteur d'*If Hemingway wrote JavaScript*. Il est autant obsédé par JavaScript que par la littérature. Il fait partie de l'équipe UI framework de Twitter où il a codéveloppé le framework Flight. Il est par ailleurs l'auteur du très influent blog « JavaScript, JavaScript… » (https://javascriptweblog.wordpress.com/) et il participe à des conférences internationales. Il tweete sous le pseudo @angustweets.

TABLE DES MATIÈRES

Avant-propos 1

À qui s'adresse ce livre ? 1
Comment lire ce livre ? 2
Que trouve-t-on dans ce livre ? 3
Amuse-toi bien ! 4

PARTIE 1
Fondamentaux

1
Qu'est-ce que le JavaScript ? 7

À la rencontre de JavaScript 8
Pourquoi apprendre le JavaScript ? 10
Écrire du JavaScript 11
La structure d'un programme JavaScript 12
    La syntaxe 14
    Les commentaires 14
Ce que tu as appris 15

2
Les types de données et les variables 17

Les nombres et les opérateurs 19
Les variables 21
    Nommer les variables 23
    Créer de nouvelles variables avec les maths 24
    Incrémenter et décrémenter 26
    += (plus-égal) et -= (moins-égal) 27
Les chaînes de caractères 28
    Joindre (concaténer) des chaînes de caractères 30
    Trouver la longueur d'une chaîne de caractères 30
    Extraire un caractère d'une chaîne de caractères 31
    Découper des chaînes de caractères 32
    Transformer des chaînes de caractères tout en majuscules
        ou tout en minuscules 33

Les booléens 35
Les opérateurs logiques 36
Comparer des nombres avec les opérateurs booléens 39
Les valeurs undefined et null 43
Ce que tu as appris 44

3
Les tableaux 45

Pourquoi s'intéresser aux tableaux ? 46
Comment créer un tableau ? 47
Comment accéder aux éléments d'un tableau ? 48
Comment attribuer des valeurs aux éléments d'un tableau et les modifier ? 49
Comment mélanger des éléments de type différent dans un tableau ? 51
Comment manipuler des tableaux ? 52
Connaître la longueur d'un tableau 52
Insérer des éléments dans un tableau 53
Supprimer des éléments d'un tableau 55
Concaténer des tableaux 57
Trouver la position d'un élément dans un tableau 58
Convertir un tableau en une chaîne de caractères 60
Des programmes utiles avec des tableaux 61
Retrouver le chemin de la maison 61
Le décideur 64
Un générateur d'insultes aléatoires 66
Ce que tu as appris 68

4
Les objets 71

Comment créer des objets ? 72
Utiliser des clés sans guillemets 73
Comment accéder aux valeurs d'un objet ? 74
Comment ajouter des valeurs dans un objet ? 75
Ajouter des valeurs dans un objet en utilisant la notation pointée 76
Comment combiner des tableaux et des objets ? 77
Un tableau d'amis 77
Comment explorer des objets dans la console ? 79
Des programmes utiles avec des objets 81
Tenir un registre de dettes 81
Cataloguer des films 82
Ce que tu as appris 84

5
Les bases du HTML 87

Les éditeurs de texte 88
La coloration syntaxique 89
Ton premier document HTML 89
Les balises et les éléments 90
Les éléments de titre 90
L'élément p (paragraphe) 91
L'espace dans HTML et les éléments de bloc 92
Les éléments en ligne 93
Un document HTML complet 94
La structure d'un document HTML 95
Ajouter des hyperliens dans le document HTML 96
Les attributs des liens 97
L'attribut title 98
Ce que tu as appris 99

6
Les conditions et les boucles 101

Comment intégrer du code JavaScript dans un document HTML ? 102
Les conditions 103
L'instruction if 104
L'instruction if... else 105
Comment créer des suites d'instructions if... else ? 106
Les boucles 109
Les boucles while 109
Les boucles for 112
Ce que tu as appris 116

7
Créer un jeu du pendu 121

Interagir avec un joueur 122
Utiliser prompt pour demander des informations à un joueur 122
Que se passe-t-il si tu cliques sur Annuler ? 124
Utiliser confirm pour poser une question oui/non à un joueur 124
Utiliser alert pour donner des informations à un joueur 125
Pourquoi utiliser alert au lieu de console.log ? 126
Concevoir ton jeu 126
Créer ton jeu en pseudo-code 127
Garder une trace de l'état du mot 127
Créer la boucle de jeu 129
Coder ton jeu 129

Choisir un mot au hasard . . . 130
Créer le tableau de réponses . . . 130
Coder la boucle de jeu . . . 131
Terminer la partie . . . 135
Le code du jeu . . . 135
Ce que tu as appris . . . 137

8
Les fonctions 141

L'anatomie de base d'une fonction . . . 142
Comment créer une fonction simple ? . . . 142
Comment appeler une fonction ? . . . 142
Comment passer des arguments à une fonction ? . . . 143
Dessiner des têtes de chats . . . 145
Comment passer plusieurs arguments à une fonction ? . . . 145
Comment renvoyer des valeurs à partir d'une fonction ? . . . 147
Comment utiliser des appels de fonctions comme des valeurs ? . . . 148
Comment simplifier ton code avec des fonctions ? . . . 149
Une fonction pour choisir un mot au hasard . . . 150
Un générateur d'insultes aléatoires . . . 150
Comment transformer ton générateur d'insultes en une fonction ? . . . 152
Comment quitter une fonction avant sa fin avec return ? . . . 153
Comment utiliser plusieurs fois return à la place d'instructions if... else ? . . . 154
Ce que tu as appris . . . 156

PARTIE 2
JAVASCRIPT AVANCÉ

9
Le DOM et jQuery 163

Sélectionner des éléments du DOM . . . 164
Identifier des éléments par id . . . 165
Sélectionner un élément avec getElementById . . . 165
Modifier le titre dans le DOM . . . 166
Manipuler l'arbre du DOM avec jQuery . . . 168
Charger jQuery dans la page HTML . . . 168
Modifier le titre avec jQuery . . . 168
Créer de nouveaux éléments avec jQuery . . . 170
Animer des éléments avec jQuery . . . 172

Enchaîner des animations avec jQuery 172
Ce que tu as appris 174

10
La programmation interactive 177

Différer l'exécution du code avec setTimeout 178
Annuler un setTimeout. 179
Exécuter du code plusieurs fois avec setInterval 180
Animer des éléments avec setInterval. 181
Réagir à des actions de l'utilisateur. 184
Réagir à des clics 184
L'événement mousemove 186
Ce que tu as appris 187

11
À la recherche du trésor enfoui ! 191

Concevoir le jeu 192
Créer la page web avec le HTML 193
Placer le trésor de manière aléatoire. 194
Sélectionner des nombres aléatoires 194
Déterminer les coordonnées du trésor 195
Le gestionnaire de clics. 195
Compter les clics 196
Calculer la distance entre un clic et le trésor 196
Utiliser le théorème de Pythagore 197
Signifier au joueur à quel point il est près de gagner 199
Vérifier si le joueur a gagné 200
Assembler le tout 201
Ce que tu as appris 203

12
La programmation orientée objet 205

Un objet simple 206
Ajouter des méthodes aux objets 206
Le mot-clé this. 207
Attribuer une même méthode à différents objets 207
Créer des objets grâce aux constructeurs 209
L'anatomie d'un constructeur. 210
Un constructeur Voiture 210
Dessiner des voitures 212
Tester la fonction dessinerVoiture 214
Personnaliser les objets grâce aux prototypes. 215

Ajouter une méthode dessiner au prototype de l'objet voiture 215
Ajouter une méthode deplacerADroite. 217
Ajouter les méthodes de déplacement à gauche, en avant et en arrière 218
Ce que tu as appris 220

PARTIE 3 CANVAS

13 L'élément canvas 225

Créer un simple canvas 226
Dessiner dans l'élément canvas 226
Sélectionner et enregistrer l'élément canvas. 227
Récupérer l'environnement de dessin 227
Dessiner un carré 227
Dessiner plusieurs carrés 228
Changer la couleur du dessin. 230
Dessiner les contours d'un rectangle 231
Dessiner des lignes ou tracés 233
Remplir des tracés 234
Dessiner des arcs et des cercles 236
Dessiner un quart de cercle ou un arc 238
Dessiner un demi-cercle 238
Dessiner un cercle complet 239
Dessiner beaucoup de cercles avec une fonction 239
Ce que tu as appris 242

14 Déplacer des objets dans le canvas 245

Comment se déplacer dans la page ? 246
Effacer complètement le canvas. 247
Dessiner le rectangle. 247
Changer de position 247
Visualiser l'animation dans le navigateur. 247
Comment animer la taille d'un carré ? 248
Une abeille aléatoire. 249
Une nouvelle fonction cercle. 250
Dessiner l'abeille 251
Mettre à jour la position de l'abeille. 253
Animer l'abeille bourdonnante 254

Comment faire rebondir une balle ? 256
Le constructeur Balle 257
Dessiner la balle 257
Déplacer la balle 258
Faire rebondir la balle 259
Animer la balle 260
Ce que tu as appris 262

15
Contrôler des animations avec le clavier 265

Les événements du clavier 266
Mettre en place le fichier HTML 266
Ajouter le gestionnaire d'événements keydown 267
Utiliser un objet pour convertir des codes de touches en noms . . . 268
Déplacer une balle avec le clavier 269
Mettre en place le canvas 270
Définir la fonction cercle 271
Créer le constructeur Balle 271
Définir la méthode deplacer 271
Définir la méthode dessiner 272
Créer une méthode definirDirection 273
Réagir au clavier 274
Animer la balle 275
Assembler le tout 276
Exécuter le code 278
Ce que tu as appris 279

16
Le jeu du serpent : partie 1 281

La structure du jeu 283
Animer la partie avec setInterval 284
Créer les objets du jeu 284
Mettre en place les contrôles au clavier 285
Préparer le code du jeu 285
Créer le code HTML 285
Définir les variables canvas, ctx, width et height 286
Diviser le canvas en blocs 286
Définir la variable score 288
Dessiner les murs 288
Afficher le score 290
Fixer la ligne de base du texte 291
Régler la taille et la police du texte 293
Écrire la fonction afficherScore 294

Terminer la partie . . . 295
Ce que tu as appris . . . 296

17
Le jeu du serpent : partie 2 299

Écrire le constructeur Bloc . . . 300
Ajouter la méthode dessinerCarre . . . 301
Ajouter la méthode dessinerCercle . . . 302
Ajouter la méthode egal . . . 304
Créer le serpent . . . 305
Écrire le constructeur Serpent . . . 306
Dessiner le serpent . . . 307
Déplacer le serpent . . . 307
Ajouter la méthode deplacer . . . 308
Ajouter la méthode verifierCollision . . . 312
Diriger le serpent avec le clavier . . . 314
Ajouter le gestionnaire d'événements keydown . . . 314
Ajouter la méthode definirDirection . . . 315
Créer la pomme . . . 317
Écrire le constructeur Pomme . . . 317
Dessiner la pomme . . . 317
Déplacer la pomme . . . 318
Assembler le tout . . . 319
Ce que tu as appris . . . 325

Pour aller plus loin 329

Plus de JavaScript . . . 330
La programmation web . . . 330
Le langage HTML . . . 330
Le langage CSS . . . 331
La programmation côté serveur avec Node.js . . . 331
La programmation graphique . . . 331
L'élément canvas . . . 332
Le SVG avec la bibliothèque Raphaël . . . 332
La programmation 3D . . . 332
La programmation de robots . . . 333
La programmation audio . . . 333
La programmation de jeux . . . 333
Partager son code avec JSFiddle . . . 334

Glossaire 335

Index 341

AVANT-PROPOS

Bienvenue dans JavaScript pour les kids ! Avec ce livre, tu apprendras à programmer en JavaScript, le langage du Web. Plus encore, tu vas devenir un programmeur : une personne qui non seulement utilise les ordinateurs, mais les contrôle également. Une fois que tu auras appris à programmer, tu pourras obtenir tout ce que tu souhaites des ordinateurs !

JavaScript est un langage de programmation intéressant à apprendre, car il est utilisé partout. Les navigateurs web tels que Chrome, Firefox et Internet Explorer s'en servent tous. Avec la puissance de JavaScript, les développeurs web peuvent transformer leurs plus simples pages web en véritables applications interactives et en jeux.

Et il n'y a pas que les pages web. JavaScript peut aussi être exécuté sur un serveur web pour créer des sites web complets. Il peut même être utilisé pour contrôler des robots et autres machines.

À qui s'adresse ce livre ?

Ce livre est destiné à toute personne qui souhaite apprendre le JavaScript ou qui veut débuter en programmation. Il est adapté aux enfants, mais il peut aussi être lu comme une introduction à la programmation pour les débutants de tout âge.

Ainsi, tu développeras tes connaissances en JavaScript étape par étape. Tu commenceras par les types de données simples de JavaScript, pour continuer ensuite avec des types plus complexes, des structures de contrôle et des fonctions. Tu pourras alors apprendre à écrire du code qui interagira avec l'utilisateur lorsque celui-ci déplace la souris ou lorsqu'il appuie sur une touche du clavier, par exemple. Enfin, tu découvriras l'élément `canvas`, qui permet de dessiner et d'animer tout ce que tu veux avec ton code JavaScript.

Tout au long de ton apprentissage, tu créeras quelques jeux afin d'étendre tes compétences en programmation tout en mettant en application ce que tu auras appris.

Comment lire ce livre ?

Tout d'abord, lis-le dans l'ordre ! Cela peut te sembler évident, mais beaucoup de personnes veulent passer directement aux parties les plus amusantes, comme celles où on crée des jeux. Chaque chapitre de ce livre s'appuie sur ce qui a été enseigné précédemment. Donc, si tu commences par le début, tu seras plus à l'aise lorsque tu arriveras à la création de jeux.

Les langages de programmation sont comme des langues maternelles : tu dois apprendre leur grammaire et leur vocabulaire, et cela prend du temps. La seule manière de progresser est d'écrire (et de lire) beaucoup de code. Plus tu écriras du JavaScript, plus certaines parties du langage deviendront une seconde nature pour toi, jusqu'au jour où tu écriras le JavaScript couramment.

En parallèle de ta lecture, je te conseille de copier et de tester le code des exemples proposés dans ce livre. Si tu ne comprends pas tout ce qu'ils font, effectue de petites modifications pour voir ce que ça change. Si le résultat n'est pas celui que tu espérais, tente de trouver pourquoi.

Par-dessus tout, essaie de résoudre les problèmes proposés dans les sections « Un petit test ! » et « Défis de programmation ». Taper le code des exemples est un bon premier pas, mais tu ne comprendras la programmation de manière plus approfondie que lorsque tu commenceras à écrire tes propres programmes. Si un des défis te plaît, alors persévère ! Et invente-toi tes propres défis pour développer encore plus les programmes que tu auras écrits.

Des exemples de solutions aux défis de programmation (ainsi que les fichiers du code des jeux et autres exemples) sont disponibles à l'adresse `http://www.editions-eyrolles.com/dl/0011850`. Regarde les solutions seulement après avoir résolu les défis, ainsi tu pourras comparer ton approche à la mienne. Si tu es bloqué, jette un œil aux solutions pour y trouver des astuces. Mais n'oublie pas qu'il ne s'agit que d'exemples de solutions. En JavaScript, la même chose peut être accomplie avec un grand nombre de manières différentes. Alors si tu te retrouves avec une solution qui est complètement différente de la mienne, ne t'inquiète surtout pas !

Si tu rencontres un terme que tu ne comprends pas, cherche-le dans le glossaire à la page 335 qui contient les définitions de la

plupart des mots du livre qui appartiennent au domaine de la programmation.

Que trouve-t-on dans ce livre ?

- Le chapitre 1 est une introduction rapide au langage JavaScript. Tu y apprends à écrire du JavaScript dans Google Chrome.
- Le chapitre 2 examine les variables et les types de données de base de JavaScript : les nombres, les chaînes de caractères et les booléens.
- Le chapitre 3 est dédié aux tableaux, utilisés pour stocker des listes de données.
- Le chapitre 4 est réservé aux objets, qui sont constitués de paires clé-valeur.
- Le chapitre 5 est une introduction au langage HTML, utilisé pour créer des pages web.
- Le chapitre 6 te montre comment contrôler encore plus le code avec des instructions `if`, des boucles `for`, et d'autres structures de contrôle.
- Le chapitre 7 rassemble tout ce que tu as appris jusqu'ici pour créer un jeu du pendu, où tu dois deviner des mots.
- Le chapitre 8 t'apprend à écrire tes propres fonctions, de manière à regrouper des bouts de code pour ensuite les réemployer.
- Le chapitre 9 est une introduction à jQuery, un outil qui permet de contrôler plus facilement les pages web avec JavaScript.
- Au chapitre 10, tu apprends à créer des pauses, des intervalles et des gestionnaires d'événements de manière à rendre ton code encore plus interactif.
- Au chapitre 11, tu utilises les fonctions, jQuery et les gestionnaires d'événements pour réaliser un jeu appelé « À la recherche du trésor enfoui ! ».
- Le chapitre 12 enseigne un style de programmation appelé « programmation orientée objet ».
- Le chapitre 13 introduit l'élément `canvas`, qui permet d'ajouter des éléments graphiques aux pages web avec ton code JavaScript.

- Le chapitre 14 s'appuie sur les techniques d'animation que tu as découvertes au chapitre 10 pour créer des animations avec canvas.
- Le chapitre 15 t'apprend à contrôler ces animations dans canvas avec ton clavier.
- Les chapitres 16 et 17 reviennent sur tout ce que tu as appris précédemment pour coder un jeu du serpent complet.
- L'épilogue fournit quelques pistes pour approfondir ce que tu as appris.
- Enfin, le glossaire contient les définitions de la plupart des mots nouveaux rencontrés en lisant ce livre.

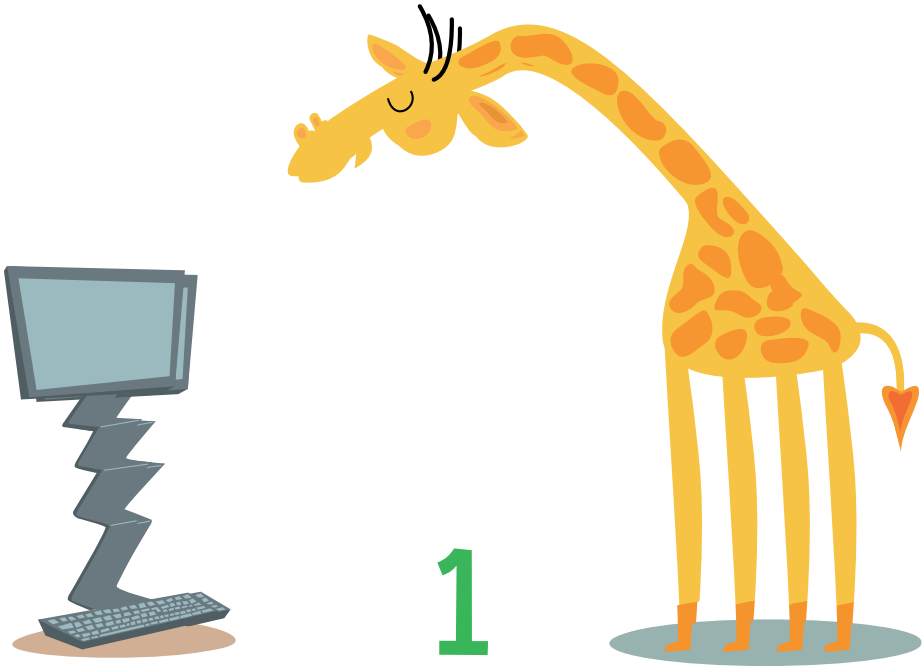
Amuse-toi bien !

Une dernière chose à ne pas oublier : amuse-toi ! La programmation peut être une activité amusante et créative, tout comme dessiner ou jouer (d'ailleurs, dans ce livre, tu vas beaucoup dessiner et jouer avec JavaScript). Une fois que tu auras compris comment coder, ta seule limite sera ton imagination.

Bienvenue dans le monde fantastique de la programmation informatique – j'espère que tu vas t'éclater !

PARTIE 1

FONDAMENTAUX



1

QU'EST-CE QUE LE JAVASCRIPT ?

Les ordinateurs sont des machines incroyablement puissantes, capables d'accomplir des exploits fantastiques tels que participer à des concours d'échecs, publier des milliers de pages web ou réaliser des millions de calculs complexes en moins de quelques secondes. Mais, en réalité, les ordinateurs sont aussi très bêtes : ils exécutent uniquement ce que les humains leur ordonnent de faire. Pour leur transmettre un ordre, nous utilisons des programmes informatiques – un ensemble d'instructions que les ordinateurs doivent suivre. Sans eux, ils ne peuvent absolument rien faire ! Pire, les ordinateurs ne comprennent ni l'anglais, ni aucune autre langue orale.

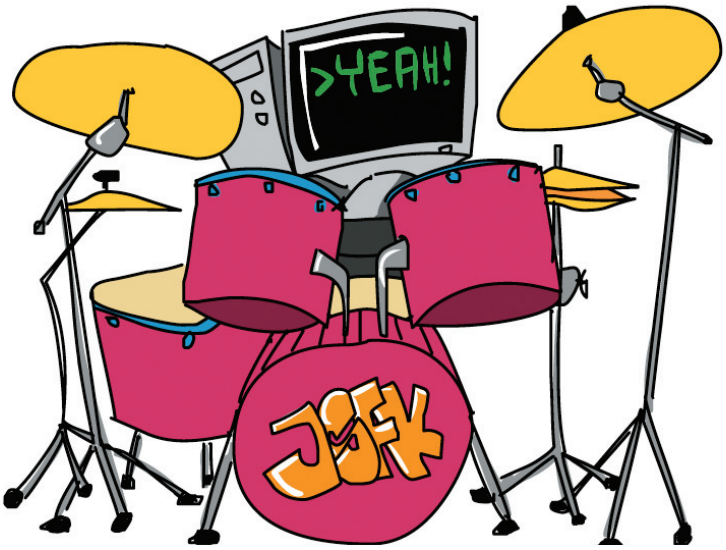
À la rencontre de JavaScript

Les programmes informatiques sont écrits dans des langages de programmation, tels que le JavaScript – tu l'as certainement utilisé même si tu n'en as peut-être jamais entendu parler auparavant.

Le JavaScript sert à écrire des programmes qui sont exécutés dans des pages web. Il permet de contrôler l'apparence de ces pages et de les faire réagir lorsqu'un visiteur clique sur un bouton ou bouge sa souris. Des sites tels que Gmail, Facebook et Twitter le mettent en œuvre pour que les utilisateurs puissent envoyer des e-mails, poster des commentaires ou parcourir des sites web, etc. Par exemple, lorsque tu lis sur Twitter un tweet de `@Eyrolles`, si tu fais défiler la page vers le bas, tu peux voir le lien « Plus de tweets » s'afficher. C'est du JavaScript en action !

Il te suffit de visiter quelques sites web pour comprendre à quel point le JavaScript est passionnant.

- Tu peux jouer de la musique et créer des effets visuels extraordinaires. Par exemple, tu peux piloter une vidéo musicale interactive créée par HelloEnjoy pour la chanson *Lights* (« lumières » en français) d'Ellie Goulding (`http://lights.helloenjoy.com/`). La figure 1-1 t'en montre un exemple.



- Tu peux aussi construire des outils que d'autres utiliseront pour créer leurs propres œuvres d'art. Patatap (http://www.patatap.com/), par exemple, est une sorte de « batterie » virtuelle qui émet toutes sortes de bruits associés à des animations (figure 1-2).

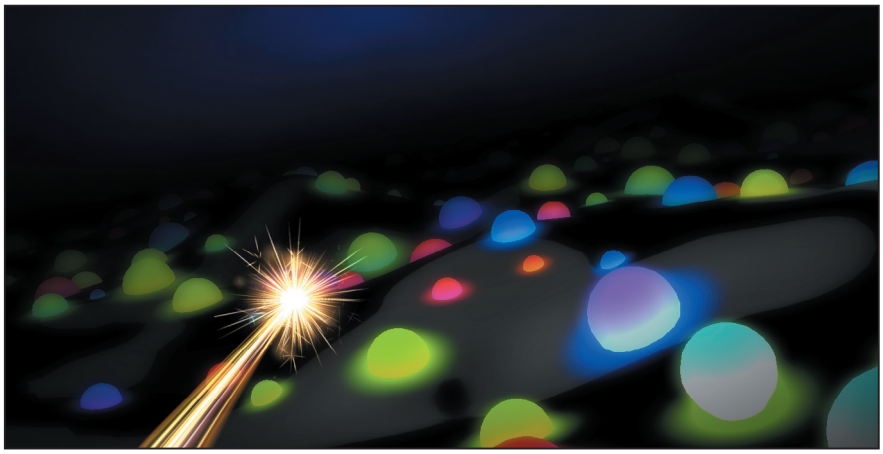


Figure 1-1. Dans la vidéo Lights de HelloEnjoy, tu peux contrôler le curseur scintillant.

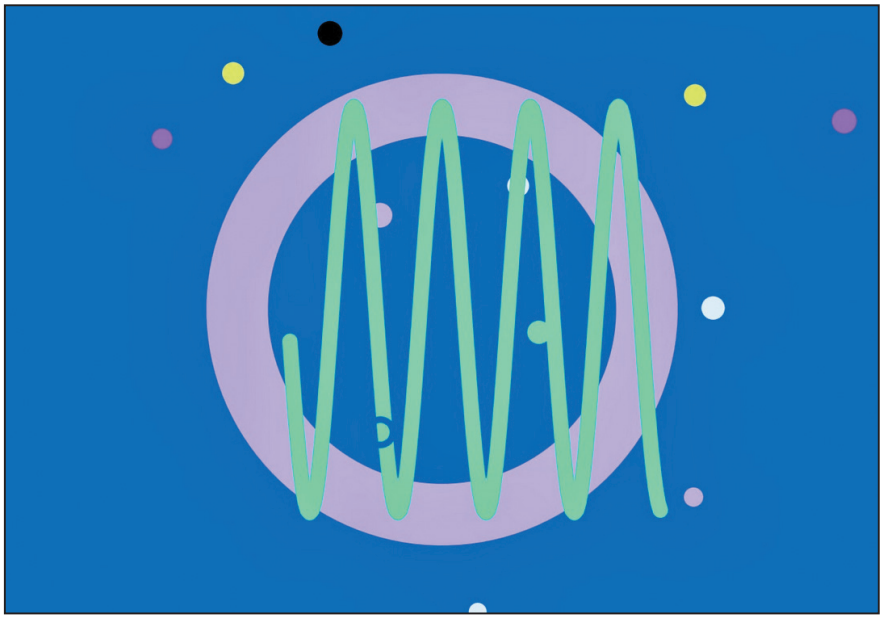


Figure 1-2. Visite le site Patatap et appuie sur quelques touches pour créer différents bruits.

- Tu peux t'amuser avec des jeux. Par exemple, Cube Slam (https://www.cubeslam.com) est une recréation en 3D du jeu Pong, bien connu de tes parents, qui ressemble un peu à un jeu de tennis de table. Mesure-toi à un de tes amis ou à un ours généré par l'ordinateur (figure 1-3).

Figure 1-3. Le jeu Cube Slam est entièrement programmé en JavaScript.

Pourquoi apprendre le JavaScript ?

Le JavaScript n'est pas le seul langage de programmation : il y en a des centaines d'autres. Il existe bien des raisons de l'apprendre ; il est en effet beaucoup plus facile (et plus amusant) que beaucoup d'autres langages de programmation ! Mais la meilleure raison est peut-être que pour écrire et exécuter des programmes JavaScript, tout ce qu'il te faut c'est un navigateur web (Internet Explorer, Mozilla Firefox ou Google Chrome). Chaque navigateur web contient un interpréteur JavaScript qui sait lire les programmes JavaScript.

Et après avoir écrit un programme en JavaScript, tu peux envoyer un lien à qui tu le souhaites à partir duquel ils pourront l'exécuter dans un navigateur sur leur propre ordinateur ! (Regarde la section « Partager son code avec JSFiddle », page 334.)

Écrire du JavaScript

À présent, écrivons un peu de JavaScript très simple dans Chrome (http://www.google.com/chrome/). Pour cela, installe Chrome sur ton ordinateur (si ce n'est pas déjà fait), puis ouvre-le. Tape `about:blank` dans la barre d'adresse et appuie sur la touche **Entrée**. Tu verras alors une page blanche apparaître.

Tu vas tout d'abord coder dans la console JavaScript de Chrome. C'est un outil secret que les développeurs utilisent pour tester de courts programmes JavaScript. Sous Windows et Linux, appuie simultanément sur les touches **Ctrl + Maj + J** ; sous Mac OS, appuie sur les touches **⌘ + Alt + J**.

Si tu as tout fait correctement, tu devrais voir une page blanche, ainsi qu'une fenêtre dans laquelle un curseur (|) clignote à droite d'un chevron (>) (figure 1-4). C'est là que tu écriras ton code JavaScript !

NOTE *La console Chrome applique des couleurs au code. Par exemple, le texte que tu saisis en entrée est bleu, et celui qui est affiché en sortie est coloré en fonction de son type. Dans ce livre, nous utiliserons les mêmes couleurs dans notre code, partout où nous nous servirons de la console.*

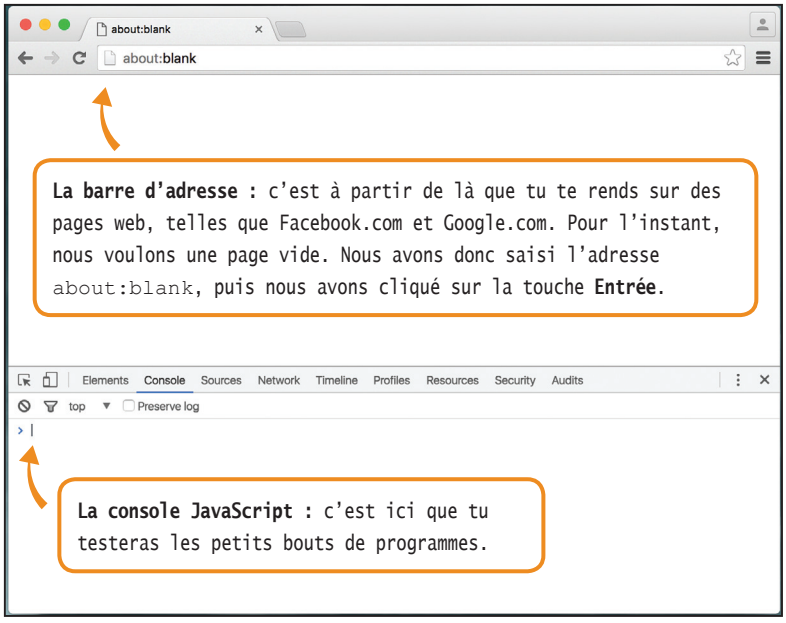


Figure 1-4. La console JavaScript de Google Chrome

Quand tu tapes du code au niveau du curseur et que tu appuies sur la touche **Entrée**, JavaScript lance, ou « exécute », ton code. Puis il affiche le résultat (s'il y en a un) sur la ligne suivante. Par exemple, tape cette ligne dans ta console :

```
3 + 4;
```

Maintenant, appuie sur **Entrée**. JavaScript devrait afficher la réponse (7) à cette simple addition sur la ligne suivante.

```
3 + 4;
7
```

Ce n'est pas si difficile que ça ! Mais JavaScript n'est-il pas plus qu'une simple calculatrice ? Essayons quelque chose d'autre.

La structure d'un programme JavaScript

Créons quelque chose d'un peu plus fou : un programme JavaScript qui affiche une série de têtes de chats, comme ceci :

```
=^.^=
```

À la différence du programme d'addition, celui-ci prendra plusieurs lignes. Pour écrire ton programme dans la console, tu devras insérer des retours chariot en appuyant sur **Maj + Entrée** à la fin de chaque ligne. (Si tu appuies seulement sur **Entrée**, Chrome essaie d'exécuter ce que tu as écrit, et bien sûr, le programme échouera. Je t'avais prévenu, les ordinateurs sont bêtes !)

Écris ceci dans la console de ton navigateur :

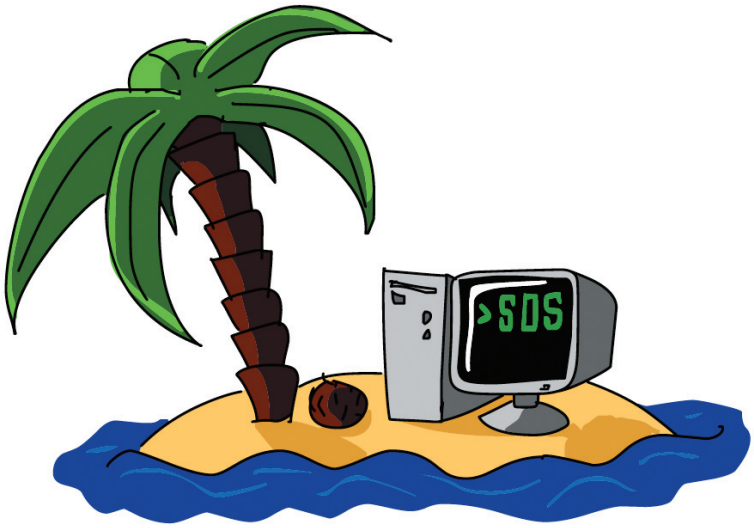
```
// Dessine autant de chats que tu le souhaites !
var dessinerChats = function (combienDeFois) {
    for (var i = 0; i < combienDeFois; i++) {
        console.log(i + " =^.^=");
    }
};

dessinerChats(10); // Tu peux mettre n'importe quel nombre ici, à la
                   // place de 10.
```

À la fin, appuie sur **Entrée** au lieu de **Maj + Entrée**. Tu devrais alors voir :

```
0 =^.^=
1 =^.^=
2 =^.^=
3 =^.^=
4 =^.^=
5 =^.^=
6 =^.^=
7 =^.^=
8 =^.^=
9 =^.^=
```

Si tu fais la moindre erreur de frappe, ton affichage peut être complètement différent, ou alors un message d'erreur apparaît. C'est ce que je voulais dire quand j'affirmais que les ordinateurs étaient bêtes ! Même un tout petit bout de code doit être absolument parfait pour que l'ordinateur puisse comprendre ce que tu veux qu'il fasse.



Pour le moment, je ne vais pas t'expliquer en détail comment ce code fonctionne (nous reverrons ce programme au chapitre 8). Mais voyons tout de même certaines fonctionnalités de ce programme et des programmes JavaScript en général.

La syntaxe

Notre programme contient pas mal de symboles, dont les parenthèses `()`, les points-virgules `;`, les accolades `{}` et le signe +. Et nous y avons utilisé quelques mots qui pourraient sembler mystérieux au premier coup d'œil (tels que `var` et `console.log`). Tous ces éléments font partie de la syntaxe de JavaScript : ce sont les règles de JavaScript qui disent comment combiner les symboles et les mots pour créer des programmes qui fonctionnent.

Lorsque nous apprenons un nouveau langage de programmation, une des choses les plus difficiles est de s'habituer aux règles qu'il faut respecter pour transmettre différents types d'instructions à l'ordinateur. En tant que débutant, tu oublieras facilement où mettre les parenthèses, ou l'ordre dans lequel il faut inclure certaines valeurs. Mais, avec de la pratique, tu vas t'y habituer !

Dans ce livre, nous y allons doucement mais sûrement, en introduisant petit à petit de nouvelles syntaxes afin que tu puisses construire des programmes de plus en plus puissants.

Les commentaires

La première ligne de notre programme des chats est celle-ci :

```
// Dessine autant de chats que tu le souhaites !
```

Cela s'appelle un commentaire. Les développeurs les incluent dans leurs programmes pour que leur code soit plus facile à lire et à comprendre des autres développeurs. Mais l'ordinateur ignore complètement les commentaires.

En JavaScript, les commentaires commencent par deux barres obliques (//). Tout ce qui suit ces deux barres obliques (sur la même ligne) est ignoré par l'interpréteur de JavaScript. Les commentaires n'ont donc aucun effet sur la manière dont un programme est exécuté ; ils sont seulement là comme description du programme.

Dans ce livre, tu trouveras des commentaires décrivant ce que fait le code. Et lorsque tu écriras tes propres programmes, ajoutes-y des commentaires. Ainsi, lorsque, plus tard, tu reliras ton code, ils te rappelleront comment celui-ci fonctionne et ce qu'il fait à chaque étape.

Dans notre programme des chats, il y a un second commentaire à la dernière ligne. Souviens-toi que tout ce qui se trouve après ces barres obliques // n'est pas exécuté par l'ordinateur.

```
dessinerChats(10); // Tu peux mettre n'importe quel nombre ici, à la
                   //  place de 10.
```

Les commentaires de code se placent soit sur une ligne à part, soit après ton code. Si tu mets les barres obliques // au tout début de la ligne, comme ceci :

```
// dessinerChats(10);
```

… rien ne se passe ! Chrome considère la ligne entière comme un commentaire, même si c'est du JavaScript.

Lorsque tu commenceras à lire du code JavaScript, tu verras aussi des commentaires qui ressemblent à ceci :

```
/*
Dessine autant de chats
que tu le souhaites !
*/
```

C'est un style différent, qui s'utilise spécialement pour les commentaires qui font plus d'une ligne. Mais c'est exactement la même chose : tout ce qui se trouve entre /* et */ est un commentaire que l'ordinateur n'exécutera pas.

Ce que tu as appris

Dans ce chapitre, tu as découvert ce qu'est le JavaScript et ce que tu peux faire avec. Tu as appris à exécuter du code JavaScript dans le navigateur Google Chrome, et tu y as testé un exemple de programme JavaScript. Tous les exemples de code de ce livre peuvent (et devraient !) être utilisés dans la console JavaScript de Chrome, à moins que je ne dise le contraire. Tu ne dois pas seulement lire les

programmes, tu dois aussi essayer de les écrire, car c'est le seul moyen pour toi d'apprendre à programmer.

Dans le prochain chapitre, tu débuteras l'apprentissage des fondamentaux de JavaScript, en commençant par les trois types de données de base avec lesquels tu peux travailler : les nombres, les chaînes de caractères et les booléens.

APPRENEZ À CODER À VOS ÉLÈVES DÈS 5 ANS !

NOUVELLE MATIÈRE DU PROGRAMME, DU CE1 À LA 3E !

SCRATCH

Dès 5 ans
160 pages – 15,90 €

Dès 8 ans
160 pages – 15,90 €

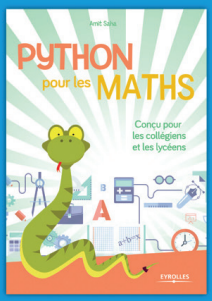
Dès 8 ans
64 pages – 12 €

Dès 10 ans
288 pages – 25 €

PYTHON

Dès 10 ans
64 pages – 12 €

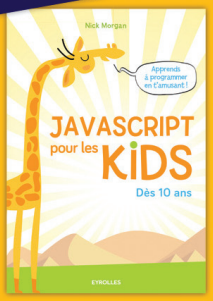
Dès 10 ans
332 pages – 22,90 €



Dès 10 ans
300 pages – 22,90 €

Dès 10 ans
350 pages – 24,90 €

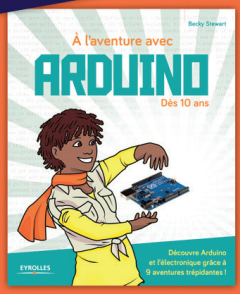
WEB



Dès 10 ans
364 pages – 22,90 €
À paraître en 2016

Dès 10 ans
152 pages – 19,90 €
À paraître en 2016

MAKERS



Dès 10 ans
352 pages – 25 €

Dès 8 ans
64 pages – 12 €
À paraître en 2016

EYROLLES

Pour en savoir plus : www.editions-eyrolles.com/go/coderaveceyrolles