



Guide du programme Swift Playgrounds

Septembre 2017

The screenshot shows a tablet displaying the Swift Playgrounds app. The interface includes a navigation bar at the top with icons for orientation, split screen, back, creation mode, plus, and more. The main area is divided into two sections: a text-based challenge on the left and a visual game world on the right.

Challenge Text:

Objectif : Utiliser les commandes Swift pour dire à Octet de se déplacer et de collecter une gemme.

Ton personnage, Octet, aime collecter les gemmes, mais il ne peut pas y arriver tout seul. Dans ce premier puzzle, tu devras écrire des **commandes** Swift pour déplacer Octet à travers le puzzle et collecter une gemme.

- 1 Cherche la gemme dans le monde du puzzle.
- 2 Saisis la combinaison correcte de commandes `moveForward()` — *avancer* — et `collectGem()` — *collecter une gemme*.
- 3 Touche Exécuter mon code.

```
moveForward()  
moveForward()  
moveForward()  
collectGem()
```

Game World:

A 3D environment featuring a character named Octet, a yellow, orange, and red blob-like creature, standing on a path. The background shows a blue sky, green grass, and some rocks. A wooden sign above the character displays "1/1". A speaker icon in the top right corner indicates audio is available.

Completion Message:

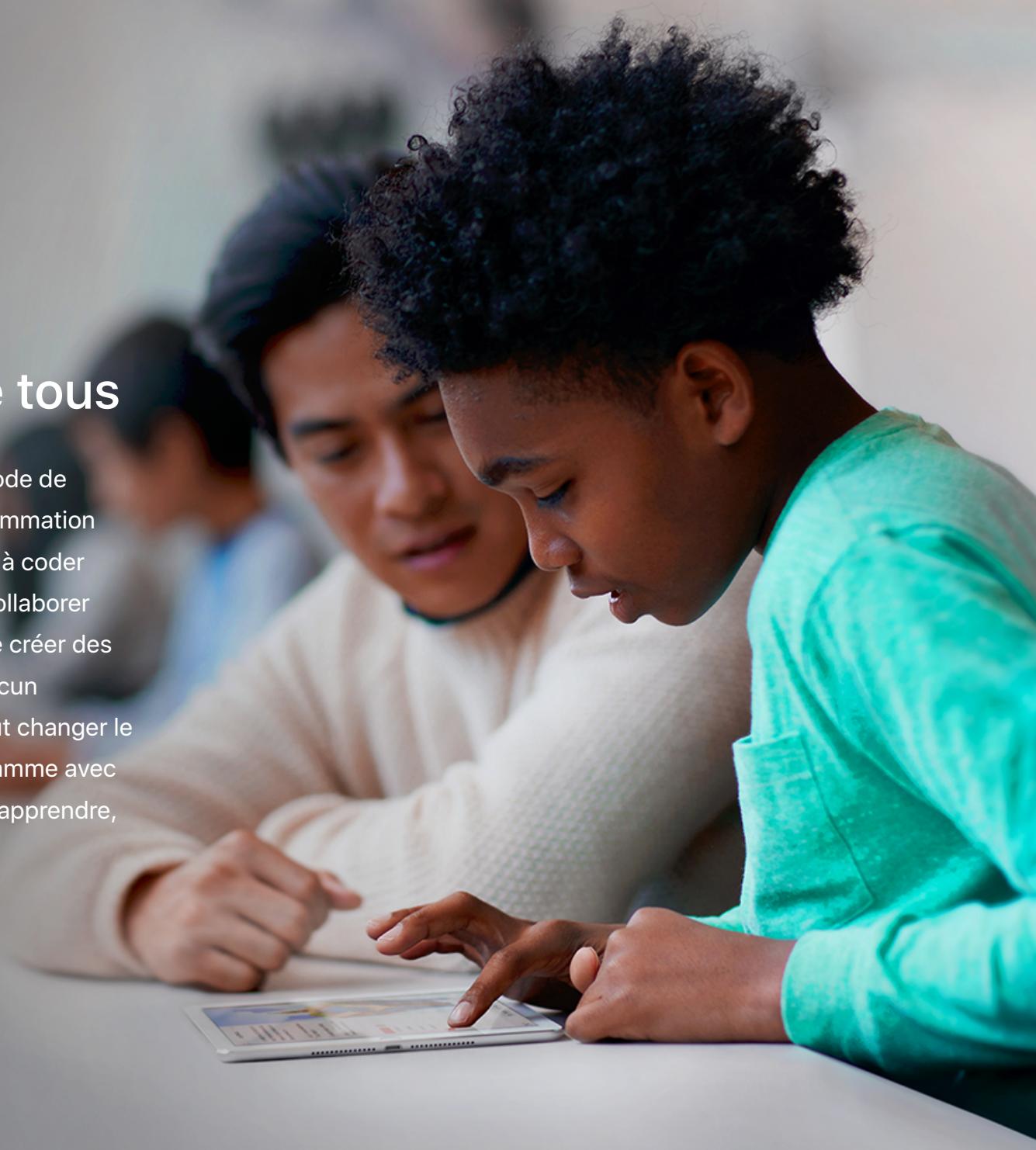
Félicitations !

Tu as rédigé tes premières lignes de code **Swift**. Octet a effectué les commandes que tu as rédigées, et a fait exactement ce que tu lui as indiqué, dans l'ordre exact que tu lui avais signifié.

[Page suivante](#) ✖

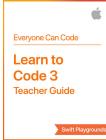
Le code à la portée de tous

Les technologies ont leur propre langage : le code de programmation. Et nous pensons que la programmation est une compétence fondamentale. Apprendre à coder vous apprend à résoudre des problèmes et à collaborer de manière créative. Cela vous permet aussi de créer des apps qui donnent vie à vos idées. D'offrir à chacun l'opportunité d'imaginer quelque chose qui peut changer le monde. Nous avons élaboré un nouveau programme avec les outils et ressources permettant à chacun d'apprendre, d'écrire et d'enseigner la programmation.



Programme Le code à la portée de tous

Le programme Le code à la portée de tous inclut toute une gamme de ressources qui guident les élèves, de leur initiation à la programmation à la conception de leurs premières apps. Le tableau ci-dessous offre un aperçu de toutes les ressources pédagogiques disponibles gratuitement.

Programme	Appareil	Public	App	Prérequis	Présentation	Supports pédagogiques	Ressources d'assistance	Nombre d'heures de cours incluses
		De la maternelle au CE1		Aucun	Commencer à penser comme des programmeurs, avec découverte pratique des concepts de programmation à l'aide d'apps visuelles.	<ul style="list-style-type: none"> Leçons de l'app codeSpark Academy Cours Cadet de l'espace sur Tynker 	<ul style="list-style-type: none"> Débuter en programmation 1 : Guide d'enseignement 	30 heures, en comptant le Guide d'enseignement et les leçons de l'app
		Du CE2 au CM2		Aucun	Explorer les concepts de base de la programmation et s'entraîner à penser comme des programmeurs à l'aide d'apps visuelles.	<ul style="list-style-type: none"> Cours Sorts Dragon sur Tynker 	<ul style="list-style-type: none"> Débuter en programmation 2 : Guide d'enseignement 	36 heures, en comptant le Guide d'enseignement et les leçons de l'app
		À partir du collège		Aucun	Apprendre les bases de la programmation à l'aide du code Swift.	<ul style="list-style-type: none"> App Swift Playgrounds Leçons Apprendre à coder 1 et 2 Cours iTunes U 	<ul style="list-style-type: none"> Apprendre à coder 1 et 2 : Guide d'enseignement Badges Swift Playgrounds Apple Teacher Learning Center 	Jusqu'à 85 heures, en comptant le Guide d'enseignement et les leçons Apprendre à coder 1 et 2
		À partir du collège		Apprendre à coder 1 et 2	Approfondir ses compétences en programmation et apprendre à penser comme un développeur d'apps.	<ul style="list-style-type: none"> App Swift Playgrounds Leçons Learn to Code 3 	<ul style="list-style-type: none"> Learn to Code 3: Teacher Guide 	Jusqu'à 45 heures, en comptant le Guide d'enseignement et les leçons Learn to Code 3
		Lycée et université		Aucun	Acquérir une expérience pratique grâce aux outils, techniques et concepts nécessaires pour concevoir une app iOS simple de A à Z.	Manuel Introduction au développement d'apps avec Swift : Guide d'enseignement	<ul style="list-style-type: none"> Introduction au développement d'apps avec Swift : Guide d'enseignement 	90 heures
		Lycée et université		Aucun	Acquérir des connaissances solides sur Swift, UIKit et les technologies réseau grâce à des travaux pratiques et des projets guidés. À l'issue du cours, les élèves sont capables créer une app issue de leur imagination.	Livre App Development with Swift et fichiers de projet	<ul style="list-style-type: none"> App Development with Swift: Teacher Guide 	180 heures

Présentation

Swift Playgrounds est une app Apple gratuite pour iPad qui rend l'apprentissage de la programmation interactif et ludique. Les élèves apprennent à résoudre des énigmes pour acquérir les bases du codage avec Swift, un langage de programmation puissant créé par Apple et utilisé par les pros pour créer certaines des apps les plus appréciées du moment.

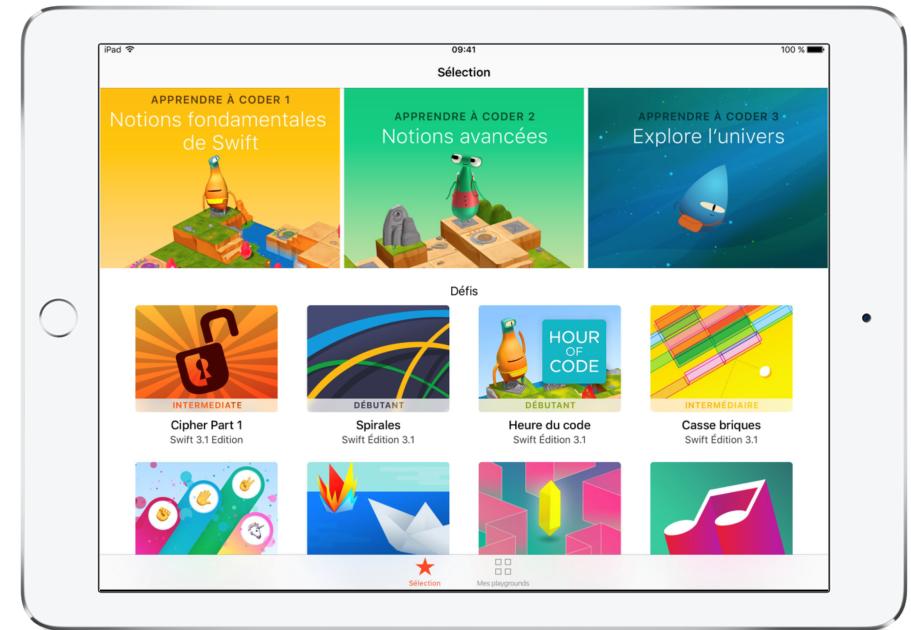
Cette app s'accompagne d'une série complète de cours de programmation développés par Apple, intitulée Apprendre à coder. À l'aide d'un vrai code Swift, les élèves résolvent des énigmes et rencontrent des personnages à contrôler d'un simple geste. En explorant un monde foisonnant et en y résolvant des puzzles, les élèves développent des compétences de programmation fondamentales. Des défis supplémentaires et des playgrounds pour appareils connectés sont également fournis : ils permettent aux élèves d'appliquer dans d'autres contextes les connaissances qu'ils ont acquises.

Au cours des leçons Apprendre à coder 1 et 2, les élèves apprennent des concepts comme les commandes, le débogage, les fonctions, les boucles et les algorithmes, entre autres. Ces leçons ne nécessitent aucune expérience en programmation et conviennent donc tout particulièrement aux débutants. La leçon Learn to Code 3 aide les élèves à déployer leurs compétences en programmation pour apprendre à penser comme un développeur d'apps. Une section facultative Conception d'apps figurant dans le Guide d'enseignement aide les enseignants à guider les élèves tout au long d'un processus de conception d'apps.

En classe

Les leçons Apprendre à coder 1 et 2 et Learn to Code 3, ainsi que celles des Guides d'enseignement, s'adressent aux élèves à partir du collège. Les ressources pédagogiques sont modulables et s'adaptent à tous les environnements d'apprentissage : elles peuvent être utilisées dans un cours de programmation unique ou être intégrées à un programme d'initiation plus général. Les leçons sont conçues pour des temps de classe de 45 à 60 minutes, bien que certaines leçons nécessitent plusieurs temps de classe. Le guide comporte des indications sur la durée requise pour réaliser les sections d'une leçon. Si vous enseignez dans un cadre moins structuré, vous pouvez scinder les leçons en plusieurs parties.

Les Guides d'enseignement constituent une aide précieuse qui permet aux enseignants avec ou sans expérience en programmation d'enseigner cette matière. Idéalement, les élèves et les enseignants doivent maîtriser les rudiments de programmation enseignés dans les leçons Apprendre à coder 1 et 2 avant de passer à la leçon Learn to Code 3.

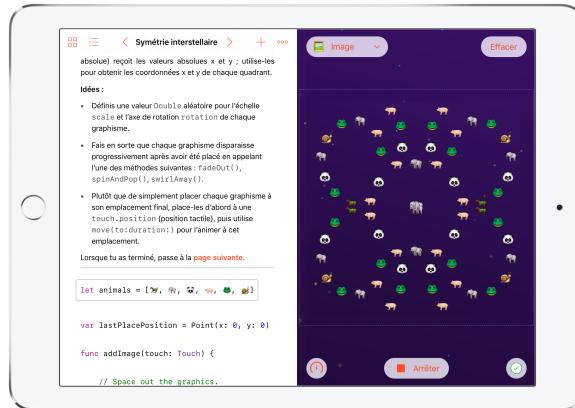


Swift Playgrounds comprend des leçons de programmation intégrées ainsi que des défis autonomes.

Principales caractéristiques

Véritable code Swift et iOS. Au cœur de Swift Playgrounds se trouve le même langage de programmation Swift utilisé pour concevoir de nombreuses applications actuellement à l'affiche sur l'App Store. Les compétences que les élèves assimilent dans Swift Playgrounds ne se limitent pas à des compétences utiles dans d'autres domaines ; ce sont précisément les compétences dont ils ont besoin pour créer des apps.

Environnement interactif. Créez un code sur la partie gauche de l'écran et visualisez instantanément le résultat sur la partie droite, d'un simple toucher.



Accessibilité. Dès le départ, l'app Swift Playgrounds a été conçue dans un souci d'accessibilité. Elle tire parti des nombreuses et puissantes fonctionnalités d'accessibilité iOS, notamment Switch Control et VoiceOver, et fournit même des commentaires vocaux supplémentaires sur les actions des personnages lorsque les élèves les contrôlent avec le code.

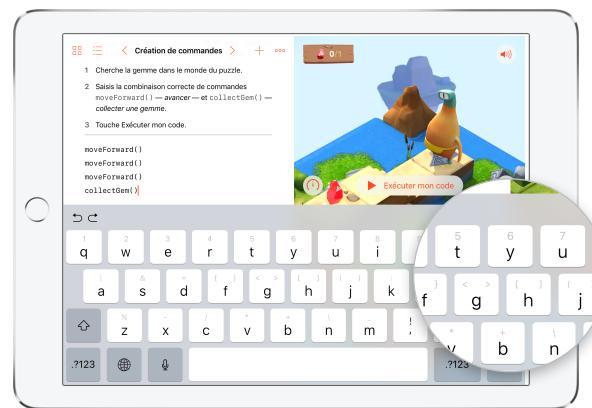
Animations immersives. Chaque section commence par une animation immersive qui établit un lien entre des concepts de programmation et le monde réel, favorisant ainsi la compréhension des élèves.

Glossaire intégré. Les définitions aident les élèves à comprendre des termes spécifiques.

Conseils utiles. Les élèves peuvent obtenir de l'aide en cours de route s'ils sont bloqués. Dans la plupart des cas, ces conseils changent de façon dynamique à mesure que les élèves saisissent du code.

Barre de raccourcis. Des suggestions QuickType pour le code apparaissent en bas de l'écran et permettent aux élèves de saisir le code dont ils ont besoin en touchant simplement la barre de raccourcis.

Clavier à l'écran. Un clavier dédié à Swift offre un accès rapide aux numéros et symboles les plus fréquemment utilisés dans Swift.



Enregistrez et partagez. Les élèves peuvent enregistrer ce qu'ils font à l'écran pour présenter leur travail.

Vérifiez le code. Exécutez le code plus rapidement ou plus lentement, ou parcourez-le pour mettre en évidence les lignes de code au fur et à mesure de leur exécution, permettant ainsi aux élèves d'identifier plus facilement les erreurs.

Touchez pour modifier. Faites glisser des structures complexes qui enveloppent d'autres codes, comme des boucles et des définitions de fonction, autour du code existant. Il suffit de toucher le mot-clé (par exemple « for ») et les commandes de glissement apparaissent à l'écran.



Modifiez sur place. Modifiez les valeurs numériques, les couleurs et les opérateurs rapidement et facilement à l'aide d'un clavier flottant.

Ressources d'assistance

Apprendre à coder 1 et 2 : Guide d'enseignement

Conçu pour être utilisé avec des élèves à partir du collège, ce Guide d'enseignement aidera chaque enseignant à présenter les cours Apprendre à coder 1 et 2. Les leçons abordent des notions de programmation essentielles, tout en démontrant que coder correspond à une manière de penser que l'on peut appliquer à d'autres domaines et dans la vie quotidienne. Elles incluent des activités améliorées, des activités de révision et de réflexion, une grille de notation et des présentations Keynote. Le guide est adapté à l'enseignement en collège et lycée. Il représente 40 à 45 heures de cours sur les fondamentaux de la programmation et jusqu'à 45 heures d'activités supplémentaires pour aider les élèves à mettre en application ce qu'ils ont appris et à commencer à concevoir leur propre app. Les correspondances avec les programmes scolaires sont incluses dans la conformité aux divers programmes nationaux et internationaux d'enseignement de l'informatique.

Learn to Code 3: Teacher Guide

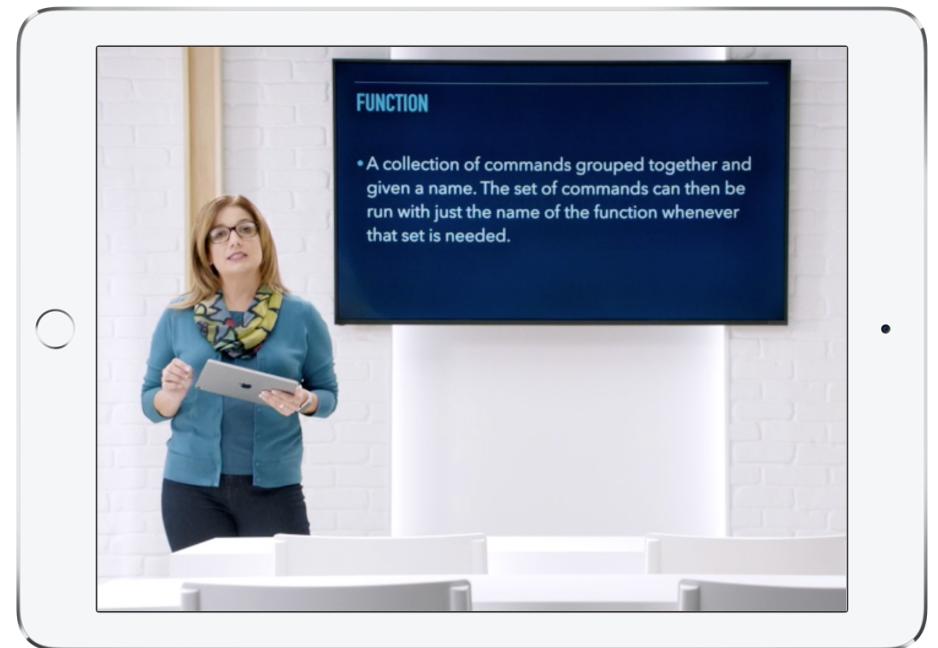
Le guide est adapté à l'enseignement en collège et lycée. Il propose 20 heures de cours sur les fondamentaux de la programmation et jusqu'à 25 heures d'activités supplémentaires pour aider les élèves à mettre en application ce qu'ils ont appris et à commencer à concevoir leur propre app. S'appuyant sur les compétences de programmation acquises dans les cours Apprendre à coder 1 et 2, il inclut, entre autres, des activités scénarisées, des leçons de révision de code, des présentations Keynote, des invitations à tenir un journal ainsi qu'une grille de notation pour aider les enseignants à animer les leçons en classe. Les correspondances avec les programmes scolaires sont incluses dans la conformité aux divers programmes nationaux et internationaux d'enseignement de l'informatique.

Apprendre à coder 1 et 2 : cours iTunes U

Ce cours iTunes U permet de suivre le Guide d'enseignement Apprendre à coder 1 et 2 à travers des leçons vidéo et autres ressources. Ces vidéos constituent également, pour les enseignants, un moyen formidable de voir comment les leçons du Guide d'enseignement peuvent être mises en application dans une salle de classe.

Programme Apple Teacher : gagner des badges Swift Playgrounds

Le programme Apple Teacher est un programme gratuit de formation professionnelle conçu pour aider et encourager les enseignants. Il propose des ressources d'apprentissage à un rythme personnel, des astuces, des sources d'inspiration et des actualités. Les Apple Teachers peuvent accéder à l'Apple Teacher Learning Center pour répondre à des questionnaires sur l'apprentissage et l'enseignement avec Swift Playgrounds et gagner ainsi quatre nouveaux badges. Ils recevront ensuite un nouveau logo Apple Teacher indiquant qu'ils ont acquis la compétence Swift Playgrounds.



Plan du cours

Apprendre à coder 1

En résolvant des puzzles dans un monde dynamique en 3D, les élèves développeront un ensemble de compétences de programmation qui leur permettront d'enrichir leur vocabulaire de programmation de base.

Leur parcours de programmation commence par de simples commandes, fonctions et boucles. Dès le départ, ils utiliseront du vrai code Swift, le même qu'utilisent les vrais programmeurs.

Leçon 0 — Premiers pas. Les élèves suivent une initiation à l'informatique et prennent connaissance des objectifs du cours.

Leçon 1 — Penser comme un ordinateur : Commandes et séquences. Les élèves découvrent l'utilisation des commandes et des séquences dans la vie de tous les jours, puis programment à l'aide de commandes et de séquences.

Leçon 2 — Penser comme un détective : Débogage. Les élèves découvrent l'utilisation du débogage dans la vie de tous les jours, puis effectuent un débogage à l'aide du code.

Leçon 3 — Penser efficacement : Fonctions et un soupçon de boucles. Les élèves découvrent l'utilisation des fonctions et des boucles for dans la vie de tous les jours, puis programment à l'aide de fonctions et de boucles for.

Revoir et réfléchir. Les élèves passent en revue les leçons 1 à 3, examinent leurs portfolios et créent une communauté avec évaluation par les pairs.

Leçon 4 — Penser logiquement : Code conditionnel. Les élèves découvrent l'utilisation du code conditionnel, des booléens et des opérateurs logiques, puis programment à l'aide du code conditionnel, de booléens et d'opérateurs logiques.

Leçon 5 — Penser encore et encore : Boucles while. Les élèves découvrent l'utilisation des boucles while dans la vie de tous les jours, puis programment en utilisant ce type de boucle.

Leçon 6 — Penser une même idée : Algorithmes. Les élèves découvrent l'utilisation des algorithmes dans la vie de tous les jours, puis programment à l'aide d'algorithmes.

Revoir et réfléchir. Les élèves passent en revue les concepts de programmation des leçons 3 à 6, poursuivent leur réflexion sur leurs portfolios et continuent leur expérience communautaire.

Apprendre à coder 2

Les élèves s'appuieront sur leurs connaissances fondamentales de Swift. Ils évolueront de la simple résolution d'énigmes à la création de leurs propres mondes. Ils découvriront les variables et les types, les structures de code qui permettent de stocker des informations et de les récupérer. Ces nouveaux acquis, complétés par l'initialisation et les paramètres, leur fourniront encore plus de façons de manier le code pour interagir avec leurs personnages et le monde de casse-tête, en leur permettant de modifier les règles mêmes de ce monde.

Leçon 7 — Penser comme un InfoMate : Variables. Les élèves découvrent l'utilisation des variables dans la vie de tous les jours, puis programment à l'aide de variables.

Leçon 8 — Penser comme un architecte : Types. Les élèves découvrent l'utilisation des types dans la vie de tous les jours, puis programment à l'aide de types et d'une initialisation.

Leçon 9 — Penser spécifiquement : Paramètres. Les élèves découvrent l'utilisation des paramètres dans la vie de tous les jours, puis programment à l'aide de paramètres.

Leçon 10 — Penser organisé : Tableaux. Les élèves découvrent l'utilisation des tableaux dans la vie de tous les jours, puis programment à l'aide de tableaux.

Projet Milestone. Les élèves bâtissent leurs propres univers en utilisant les concepts appris tout au long du programme, créant ainsi leur propre histoire. Ils réfléchissent à ce qu'ils ont appris à l'aide de leurs portfolios et des commentaires recueillis auprès des pairs de la communauté.

Conception d'apps. Les élèves vont appliquer un cycle de conception axé sur la réalisation de prototypes, très proche du processus suivi par les développeurs professionnels.

Plan du cours (suite)

Learn to Code 3

Le cours Learn to Code 3 aide les élèves à déployer les compétences en programmation qu'ils ont acquises lors des leçons précédentes pour apprendre à penser comme un développeur d'apps. Il est recommandé de suivre les leçons Apprendre à coder 1 et 2 avant de passer au volet Learn to Code 3.

En parcourant l'espace interstellaire de l'univers Blu, les élèves construisent un ensemble d'outils créatifs en explorant les puissants concepts de programmation utilisés par les développeurs professionnels. Après avoir étudié les graphismes et les coordonnées, ils pourront placer et manipuler des images, puis combiner toutes ces techniques avec des événements tactiles pour dessiner des figures artistiques dans l'espace.

Après les événements tactiles, les élèves se plongent dans le concept des chaînes et font ainsi entendre leur voix dans l'univers silencieux de Blu. Enfin, ils explorent les gestionnaires d'événements en utilisant des événements réels, comme des mouvements de doigts ou des touchers, pour déclencher leur code. Grâce aux gestionnaires d'événements, ils créent des aliens animés ou transforment l'univers en un gigantesque instrument de musique. À la fin du cours, ils auront appris à combiner leurs compétences avec brio en écrivant un code plus avancé que jamais !

Leçon 1 — Introduction to Learn to Code 3: Coordinates. Les élèves découvrent les coordonnées en examinant des algorithmes, des boucles « for » ainsi que des tableaux, puis programment en utilisant une combinaison de ces concepts. Ils discutent également des éléments qui composent une histoire illustrée réussie.

Leçon 2 — Think Like an App Designer: Touch Events. Les élèves examinent les variables, les types et l'initialisation en analysant leurs apps préférées, puis créent et initialisent leurs propres outils d'image dans Swift Playgrounds. Ils étudient également l'impact des images sur les histoires illustrées.

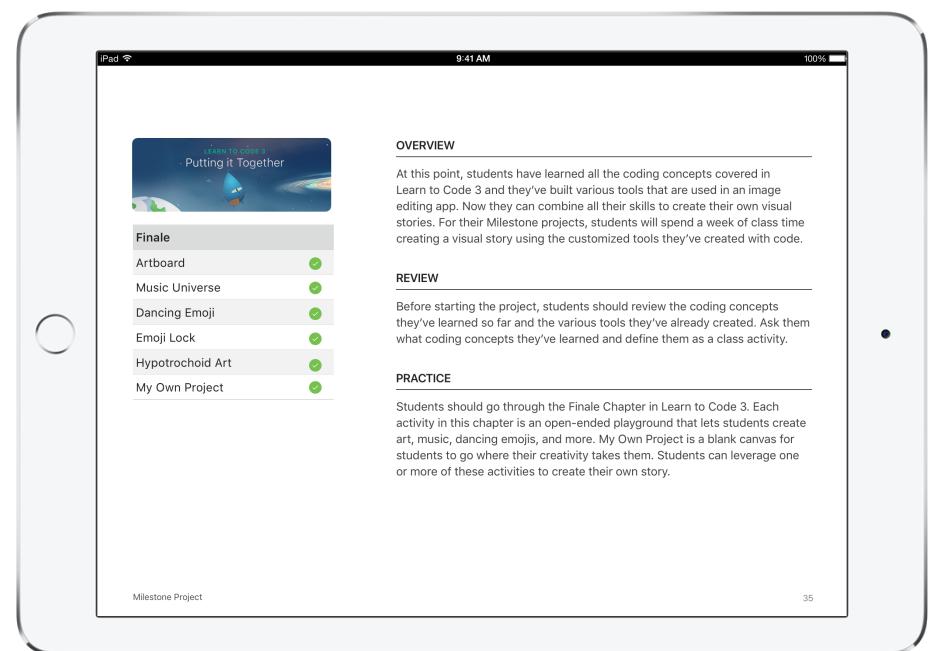
Leçon 3 — Think Like an Editor: Strings. Les élèves découvrent l'utilisation des chaînes dans la vie de tous les jours, puis créent leurs propres outils de texte dans Swift Playgrounds. Ils étudient également l'impact du texte sur les histoires illustrées.

Leçon 4 — Think Like an Animator: Event Handlers.

Les élèves découvrent les gestionnaires d'événements en concevant leurs propres jeux. Ils créent ensuite leurs propres outils d'action dans Swift Playgrounds et étudient l'impact de l'interactivité sur les histoires illustrées.

Milestone Project. Les élèves codent leurs propres histoires illustrées dans Swift Playgrounds.

App Design. Les élèves vont appliquer un cycle de conception axé sur la réalisation de prototypes, très proche du processus suivi par les développeurs professionnels.



Informations complémentaires

Swift Playgrounds nécessite iOS 10 et fonctionne sur les modèles d'iPad suivants :

- iPad Pro (9,7 pouces)
- iPad Pro (12,9 pouces)
- iPad
- iPad Air 2
- iPad Air
- iPad mini 4
- iPad mini 3
- iPad mini 2

Téléchargez les ressources Swift Playgrounds :

- [Learn to Code 1 & 2: iTunes U Course](#)
- [Apprendre à coder 1 et 2 : Guide d'enseignement](#)
- [Learn to Code 3: Teacher Guide](#)
- [App Swift Playgrounds](#)

Télécharger les guides App Development with Swift

- [Introduction au développement d'apps avec Swift](#)
- [Introduction au développement d'apps avec Swift : Guide d'enseignement](#)
- [App Development with Swift](#)
- [App Development with Swift: Teacher Guide](#)

Les fonctionnalités sont sujettes à modification. La disponibilité des fonctionnalités peut varier en fonction des zones géographiques et des langues.

© 2017 Apple Inc. Tous droits réservés. Apple, le logo Apple, iPad, iPad Air, iPad Pro, QuickType et Xcode sont des marques d'Apple Inc., déposées aux États-Unis et dans d'autres pays. iPad mini et Swift sont des marques d'Apple Inc. App Store et Genius Bar sont des marques de service d'Apple Inc., déposées aux États-Unis et dans d'autres pays. Les autres noms de produits et de sociétés mentionnés dans ce document appartiennent à leurs propriétaires respectifs. Les caractéristiques des produits sont susceptibles d'être modifiées sans préavis. Les informations contenues dans ce document sont fournies à titre indicatif uniquement ; Apple n'assume aucune responsabilité quant à leur utilisation. Septembre 2017