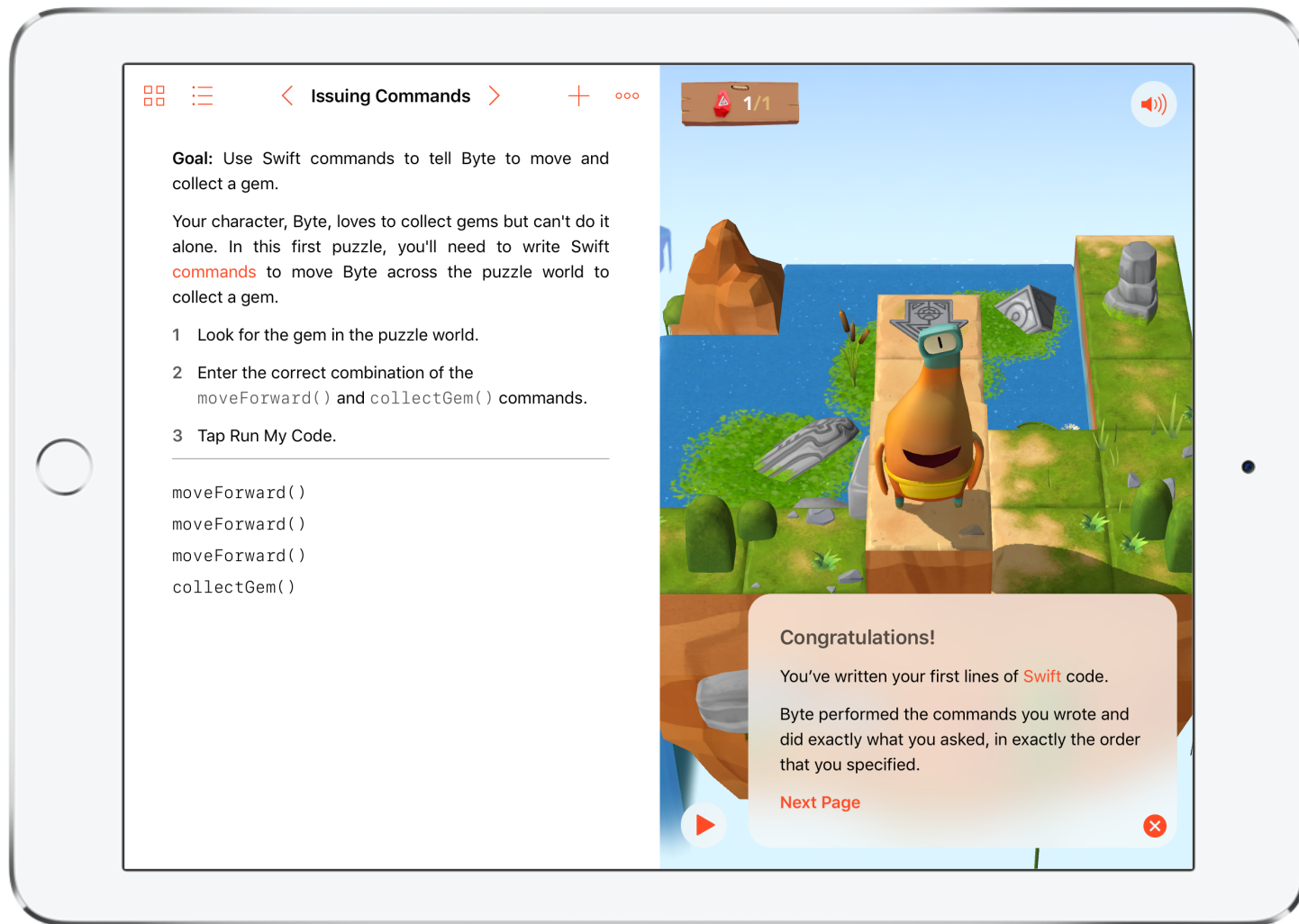




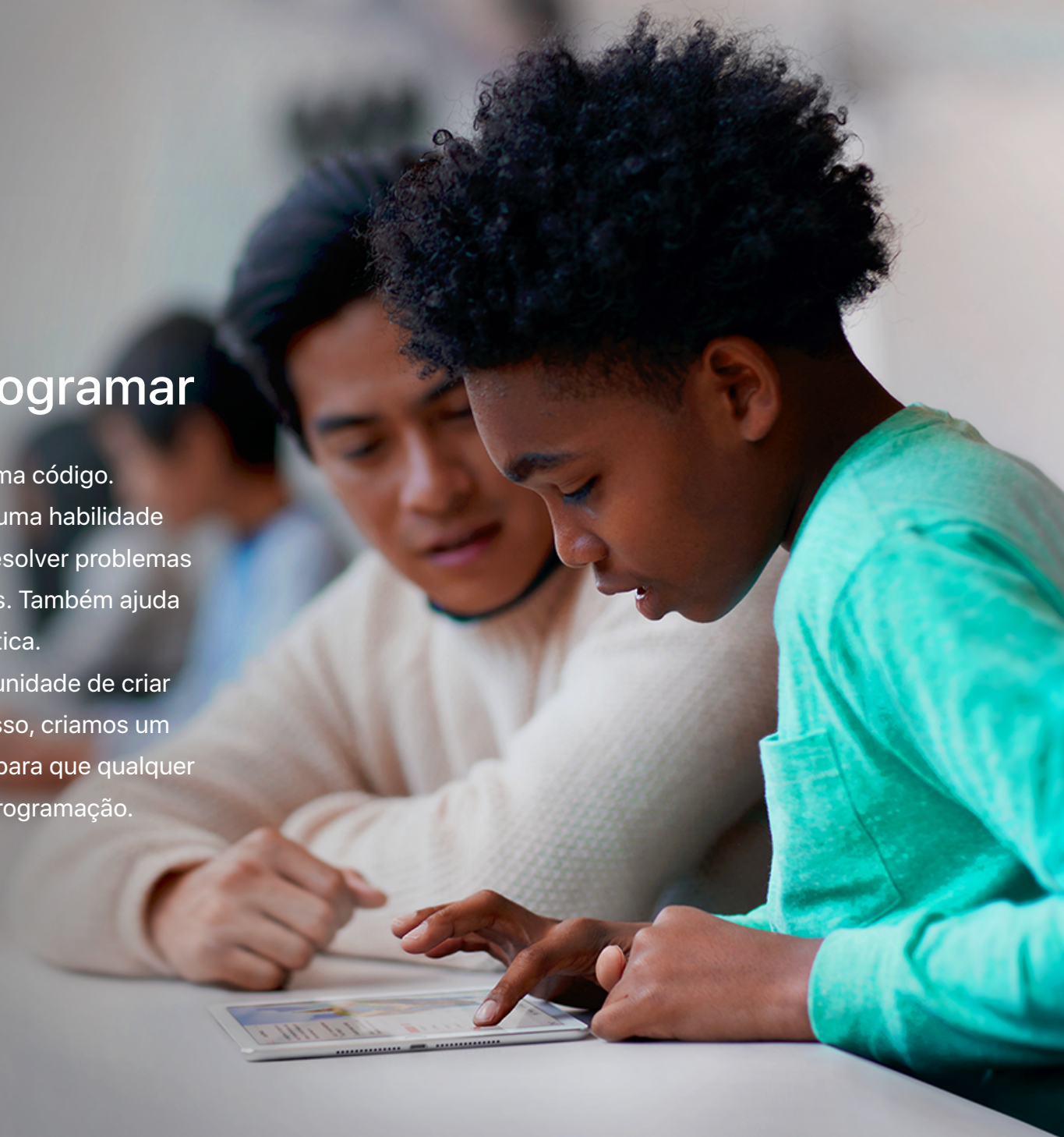
Guia Swift Playgrounds

Setembro de 2017






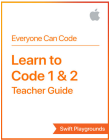




Todo mundo pode programar

A tecnologia tem uma linguagem. Ela se chama código. Acreditamos que dominar essa linguagem é uma habilidade muito importante. A programação ensina a resolver problemas e a trabalhar em equipe de maneiras criativas. Também ajuda a criar apps que colocam suas ideias em prática. Acreditamos que todos deveriam ter a oportunidade de criar algo com potencial de mudar o mundo. Por isso, criamos um novo programa com recursos e ferramentas para que qualquer pessoa possa aprender, formular e ensinar programação.



Currículo do programa Todo mundo pode programar

O programa Todo mundo pode programar conta com diversos recursos que habilitam os alunos a desenvolver seus primeiros apps, mesmo sem nenhuma experiência em programação. A tabela abaixo apresenta uma visão geral de todos os recursos de ensino e aprendizagem gratuitos disponíveis.

Currículo	Dispositivo	Público	Aplicativo	Pré-requisitos	Visão geral	Materiais de aprendizagem	Recursos de apoio	Horas-aula incluídas
		Do jardim de infância ao 2º ano		Nenhum	Usar apps visuais para explorar conceitos de programação de forma prática e começar a pensar como um programador.	<ul style="list-style-type: none"> Aulas do app codeSpark Academy Curso Tynker Space Cadet 	<ul style="list-style-type: none"> Comece a Programar 1: Guia do professor 	30 horas, incluindo o Guia do professor e as aulas dos apps
		Do 3º ao 5º ano		Nenhum	Explorar os conceitos básicos da programação e usar apps visuais para praticar o pensamento da programação.	<ul style="list-style-type: none"> Curso Tynker Dragon Spells 	<ul style="list-style-type: none"> Comece a Programar 2: Guia do professor 	36 horas, incluindo o Guia do professor e as aulas dos apps
		A partir do ensino fundamental II		Nenhum	Aprender os conceitos básicos de programação usando código Swift real.	<ul style="list-style-type: none"> App Swift Playgrounds Aulas dos cursos Aprenda a Programar 1 e 2 Curso do iTunes U 	<ul style="list-style-type: none"> Aprenda a Programar 1 e 2: Guia do professor 	Até 85 horas, incluindo o Guia do professor e as aulas dos cursos Aprenda a Programar 1 e 2
		A partir do ensino fundamental II		Aprenda a Programar 1 e 2	Expandir as habilidades de programação e começar a pensar como um desenvolvedor de apps.	<ul style="list-style-type: none"> App Swift Playgrounds Aulas do curso Aprenda a Programar 3 	<ul style="list-style-type: none"> Aprenda a Programar 3: Guia do professor 	Até 45 horas, incluindo o Guia do professor e as aulas dos cursos Aprenda a Programar 3
		Ensino médio e superior		Nenhum	Ganhar experiência prática com as ferramentas, técnicas e conceitos necessários para criar um app do iOS do zero.	Livro Introdução ao Desenvolvimento de Apps com Swift e arquivos de projeto	<ul style="list-style-type: none"> Introdução ao Desenvolvimento de Apps com Swift: Guia do professor 	90 horas
		Ensino médio e superior		Nenhum	Aprender conceitos básicos sobre Swift, UIKit e redes com laboratórios práticos e projetos orientados. Até o final do curso, os alunos serão capazes de criar um app idealizado por eles.	Livro Desenvolvimento de Apps com Swift e arquivos de projeto	<ul style="list-style-type: none"> Desenvolvimento de Apps com Swift: Guia do professor 	180 horas

Visão geral

O Swift Playgrounds é um app gratuito para iPad da Apple que torna o aprendizado e a experiência de programação mais interativos e divertidos. Os alunos podem resolver enigmas para dominar os conceitos básicos da Swift, uma linguagem de programação eficiente criada pela Apple e usada por profissionais para criar muitos dos apps mais populares da atualidade.

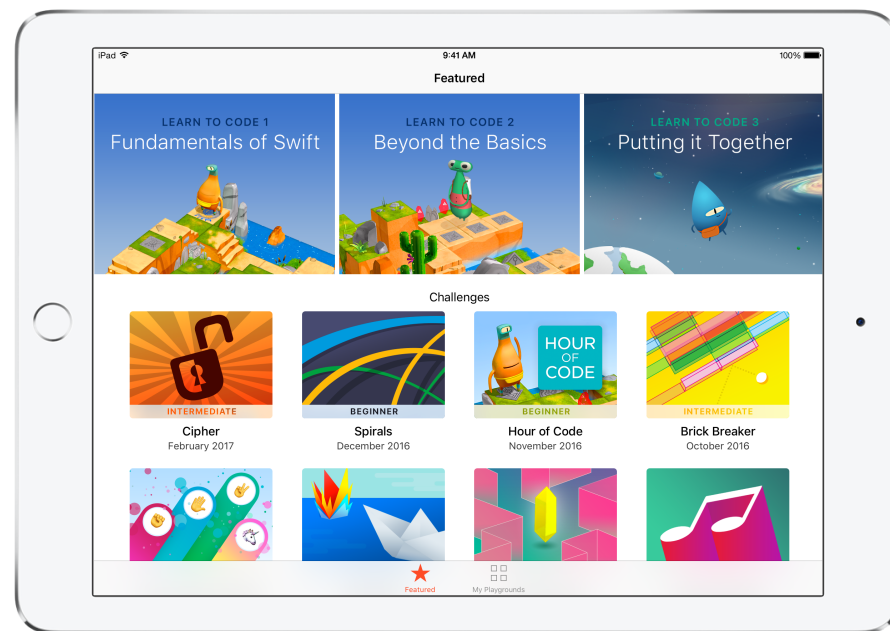
O app vem acompanhado de um conjunto completo de aulas desenvolvidas pela Apple chamado Aprenda a Programar. Usando o código Swift real, os alunos resolvem enigmas e conhecem personagens que podem controlar com apenas um toque. Ao explorar e resolver elaborados mundos de enigmas, os alunos desenvolvem habilidades de programação que se tornam a base desse conhecimento. Resolvendo outros desafios e playgrounds de dispositivos conectados que também estão incluídos, os alunos aplicam o que eles aprenderam a novos contextos.

Nos cursos Aprenda a Programar 1 e 2, os alunos aprendem conceitos como comandos, depuração, funções, loops, algoritmos e outros. Como não exigem experiência prévia, as aulas são ideais para programadores iniciantes. O Aprenda a Programar 3 ajuda os alunos a expandirem as habilidades de programação para começarem a pensar mais como um desenvolvedor de apps. Uma seção de design de app opcional do Guia do professor ajuda os professores a orientar os alunos por um processo de design de apps.

Na sala de aula

Os cursos Aprenda a Programar 1, 2 e 3 e as aulas dos Guias do professor podem ser usados com alunos a partir do ensino fundamental II. Os materiais são flexíveis e podem ser usados em qualquer ambiente de aprendizagem, seja em uma aula de programação independente, seja como parte de uma introdução à programação. As aulas foram desenvolvidas para períodos de 45 a 60 minutos, e algumas se estendem por vários períodos. A sugestão do tempo necessário para concluir cada seção de uma aula está incluída no conteúdo. Portanto, se você der uma aula menos estruturada, como uma atividade extracurricular, será possível dividi-la.

O Guia do professor oferece apoio para que professores com ou sem experiência de programação o utilizem para ensinar. É recomendável que os alunos e professores conheçam os conceitos básicos de programação ensinados nos

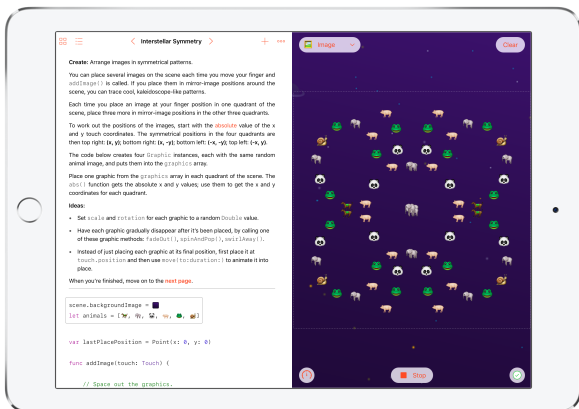


O Swift Playgrounds tem aulas de programação integradas e desafios independentes.

Principais características

Swift e iOS de verdade. No coração do Swift Playgrounds está a mesma linguagem de programação usada para criar muitos dos apps mais populares na App Store hoje. O que os alunos aprendem no Swift Playgrounds vai além do programa: são exatamente as habilidades de que eles precisam para criar apps.

Ambiente interativo. Crie código no lado esquerdo da tela e veja os resultados na hora à direita, com apenas um toque.



Acessibilidade. O Swift Playgrounds foi criado pensando na acessibilidade desde o início. Ele aproveita muitos recursos avançados de acessibilidade eficientes do iOS, como o Controle Assistivo e o VoiceOver, oferecendo até comentários adicionais sobre as ações de personagens à medida que os alunos os controlam com o código.

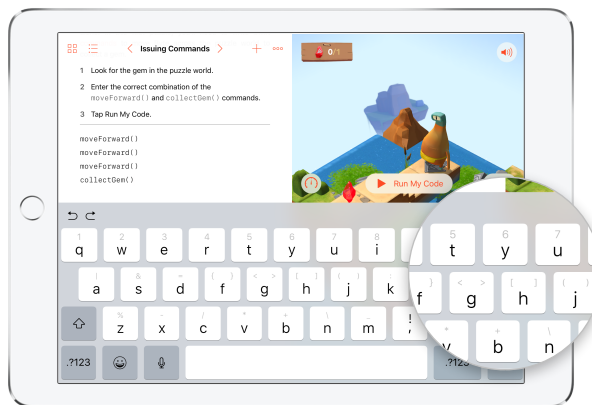
Animações envolventes. Cada seção começa com uma animação que relaciona conceitos de programação com a vida real, ajudando na compreensão do aluno.

Glossário integrado. As definições ajudam os alunos a compreender termos específicos.

Dicas úteis. Os alunos poderão obter ajuda se não souberem o que fazer. Em muitos casos, as dicas mudam dinamicamente à medida que digitam o código.

Barra de atalhos. As sugestões do QuickType para o código aparecem na parte inferior da tela e permitem que os alunos insiram o código de que precisam simplesmente tocando na barra de atalhos.

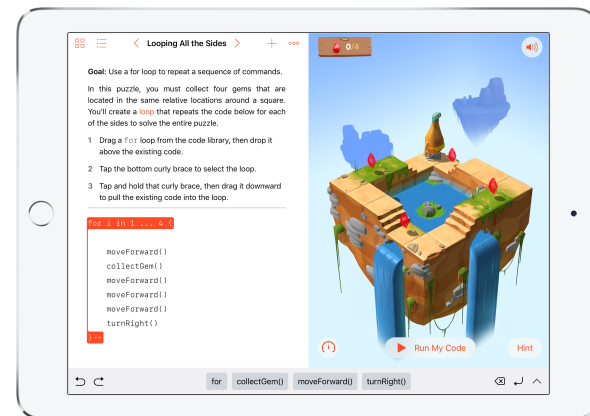
Teclado na tela. Um teclado específico dá acesso rápido aos números e símbolos usados com mais frequência na linguagem Swift.



Gravação e compartilhamento. Os alunos podem gravar o que fazem na tela para demonstrar seu trabalho.

Revisão de código. Execute o código com mais ou menos rapidez, ou percorra-o para destacar as linhas de código à medida que forem executadas, facilitando a identificação dos erros pelos alunos.

Toque para editar. Arraste estruturas complexas que encapsulam outro código, como loops e definições de função, em torno de um código existente. Basta tocar na palavra-chave (como "for") e os controles para arrastar serão exibidos na tela.



Edite no local. Edite valores numéricos, cores e operadores de maneira rápida e fácil usando o teclado popover.

Recursos de apoio

Aprenda a Programar 1 e 2: Guia do professor

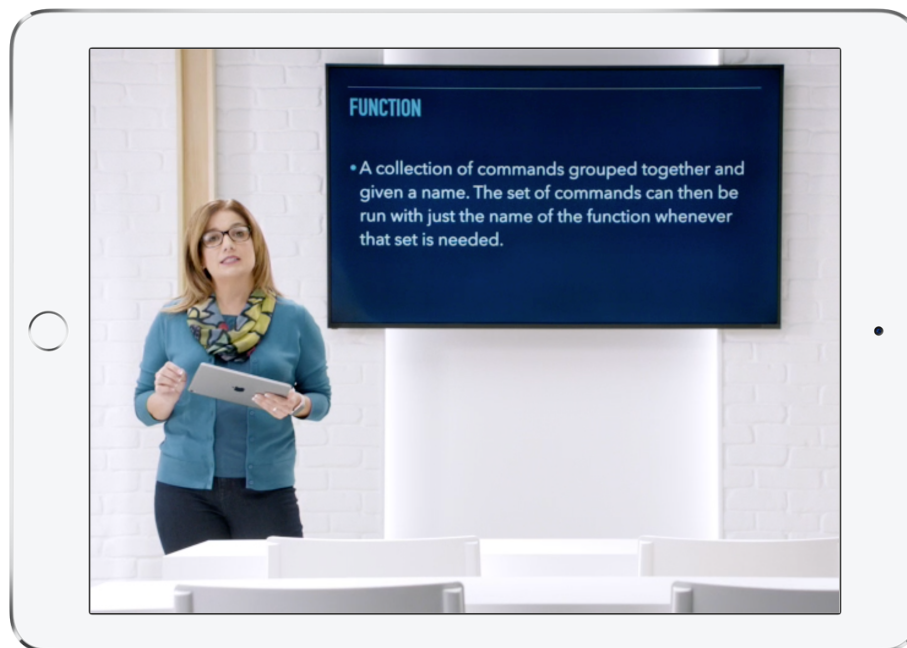
Criado para uso com alunos a partir do ensino fundamental II, este Guia do professor ajudará qualquer professor a levar os cursos Aprenda a Programar 1 e 2 para a sala de aula. As aulas destacam os principais conceitos de programação ao demonstrar como a programação é um modo de pensar que pode ser aplicado a outros temas e ao dia a dia. O conteúdo abrange atividades aprimoradas, atividades de revisão e reflexão, uma ficha de avaliação e apresentações do Keynote. O guia é voltado para alunos a partir do ensino fundamental II. Ele representa de 40 a 45 horas de aulas sobre conceitos principais de programação, com até 45 horas de atividades suplementares que ajudam os alunos a aplicar o que eles aprenderam e começar a desenvolver seu próprio app. As correlações de currículos estão incluídas com alinhamentos com vários padrões de currículos nacionais e internacionais para ciência da computação.

Aprenda a Programar 3: Guia do professor

O guia é voltado para alunos a partir do ensino fundamental II. Ele inclui 20 horas de aulas sobre conceitos principais de programação com até 25 horas de atividades suplementares que ajudam os alunos a aplicar o que eles aprenderam e começar a desenvolver seu próprio app. Com base nas habilidades de programação dos cursos Aprenda a Programar 1 e 2, ele conta com atividades de história, aulas de revisão de código, apresentações do Keynote, atividades de diário, uma ficha de avaliação e muitas atividades que ajudam os professores em sala de aula. As correlações de currículos estão incluídas com alinhamentos com vários padrões de currículos nacionais e internacionais para ciência da computação.

Aprenda a Programar 1 e 2: curso do iTunes U

Este curso do iTunes U traz o Guia do professor dos cursos Aprenda a Programar 1 e 2 em formato de videoaulas e recursos adicionais. Os vídeos também são uma excelente maneira para os professores verem como colocar o Guia do professor em prática em uma sala de aula.



Estrutura do curso

Aprenda a Programar 1

Para resolver enigmas em um mundo dinâmico em 3D, os alunos desenvolverão um conjunto de habilidades para formar um vocabulário básico de programação. A jornada da programação começa com comandos, funções e loops simples. Desde o início, eles programarão em Swift de verdade, usando o mesmo código que programadores profissionais.

Aula 0 — Introdução. Os alunos recebem uma introdução à ciência da computação e aos objetivos do curso.

Aula 1 — Pensar como um computador: comandos e sequências.

Os alunos aprendem sobre o uso de comandos e sequências em uma situação cotidiana e programam usando o que aprenderam na aula.

Aula 2 — Pensar como um detetive: depuração. Os alunos aprendem sobre o uso de depuração em uma situação cotidiana e depuram com código.

Aula 3 — Pensar com eficiência: funções e um pouco de loops.

Os alunos aprendem sobre o uso de funções e loops “for” em uma situação cotidiana e programam usando o que aprenderam na aula.

Revisão e reflexão. Os alunos revisam as aulas 1 a 3, analisam seus portfólios e criam uma comunidade com revisão por pares.

Aula 4 — Pensar com lógica: código condicional. Os alunos aprendem sobre a utilização de código condicional, operadores booleanos e lógicos e programam usando o que aprenderam na aula.

Aula 5 — Pensar sem parar: loops “while”. Os alunos aprendem sobre o uso de loops “while” em uma situação cotidiana e programam usando o que aprenderam na aula.

Aula 6 — Pensar a mesma ideia: algoritmos. Os alunos aprendem sobre o uso de algoritmos em uma situação cotidiana e programam usando o que aprenderam na aula.

Revisão e reflexão. Os alunos revisam os conceitos de programação das aulas 3 a 6, continuam refletindo sobre seus portfólios e dão prosseguimento à experiência em comunidade.

Aprenda a Programar 2

Os alunos ampliarão o conhecimento básico sobre a linguagem Swift. Eles irão além da simples solução de enigmas e criarão mundos próprios. Eles aprenderão sobre variáveis e tipos, os “constructors” de programação que permitem armazenar e acessar informações. Essas novas habilidades, junto com a inicialização e os parâmetros, proporcionarão outras maneiras de usar o código para interagir com os personagens e o mundo dos enigmas e permitirão que as regras do próprio mundo sejam alteradas.

Aula 7 — Pensar como um bot de notícias: variáveis. Os alunos aprendem sobre o uso de variáveis em uma situação cotidiana e programam usando o que aprenderam na aula.

Aula 8 — Pensar como um arquiteto: tipos. Os alunos aprendem sobre o uso de tipos em uma situação cotidiana e programam usando tipos e inicialização.

Aula 9 — Pensar especificamente: parâmetros. Os alunos aprendem sobre o uso de parâmetros em uma situação cotidiana e programam usando o que aprenderam na aula.

Aula 10 — Pensar com organização: matrizes. Os alunos aprendem sobre o uso de matrizes em uma situação cotidiana e programam usando o que aprenderam na aula.

Projeto-marco. Os alunos criam seus próprios mundos usando os conceitos aprendidos ao longo do programa, criando uma história para esse mundo. Eles refletirão sobre o que aprenderam usando os portfólios e a comunidade de revisão por pares.

Design de apps. Os alunos passam por um ciclo de design, cujo foco é criar um protótipo, semelhante ao processo que os desenvolvedores profissionais de apps passam.

Estrutura do curso (continuação)

Aprenda a Programar 3

O curso Aprenda a Programar 3 ajuda os alunos a expandirem as habilidades de programação aprendidas nas aulas anteriores para começarem a pensar mais como um desenvolvedor de apps. Os cursos Aprenda a Programar 1 e 2 são pré-requisitos recomendados para o curso Aprenda a Programar 3.

Ao encontrar o espaço interestelar do universo de Blu, os alunos assimilam um conjunto de ferramentas criativas à medida que exploram poderosos conceitos de programação usados por desenvolvedores profissionais. Enquanto aprendem sobre gráficos e coordenadas, eles poderão inserir e manipular imagens para depois combinar técnicas com eventos de toque para pintar formas artísticas no espaço.

Depois de aprender sobre os eventos de toque, os alunos passam para as strings, proporcionando uma forma de dar sua voz ao silencioso universo de Blu. Por fim, eles explorarão os manipuladores de evento ao usar eventos reais, como movimentos dos dedos ou toques para ativar o código. Com os manipuladores de eventos, eles poderão criar alienígenas animados ou transformar o universo em um instrumento musical gigante. Ao concluírem o curso, eles conseguirão combinar tudo o que aprenderam de uma forma muito metódica, com a capacidade de formular códigos cada vez mais avançados!

Aula 1 — Introdução ao curso Aprenda a Programar 3: coordenadas.

Os alunos aprendem sobre coordenadas, veem algoritmos, loops “for” e matrizes e, em seguida, programam usando uma combinação dos conceitos. Eles também debatem sobre o que faz uma história visual ser excelente.

Aula 2 — Pensar como um designer de apps: eventos de toque.

Os alunos veem variáveis, tipos e inicialização ao analisar seus apps favoritos e, em seguida, criam e inicializam suas próprias ferramentas de imagem no Swift Playgrounds. Eles também pesquisam sobre a influência das imagens em histórias visuais.

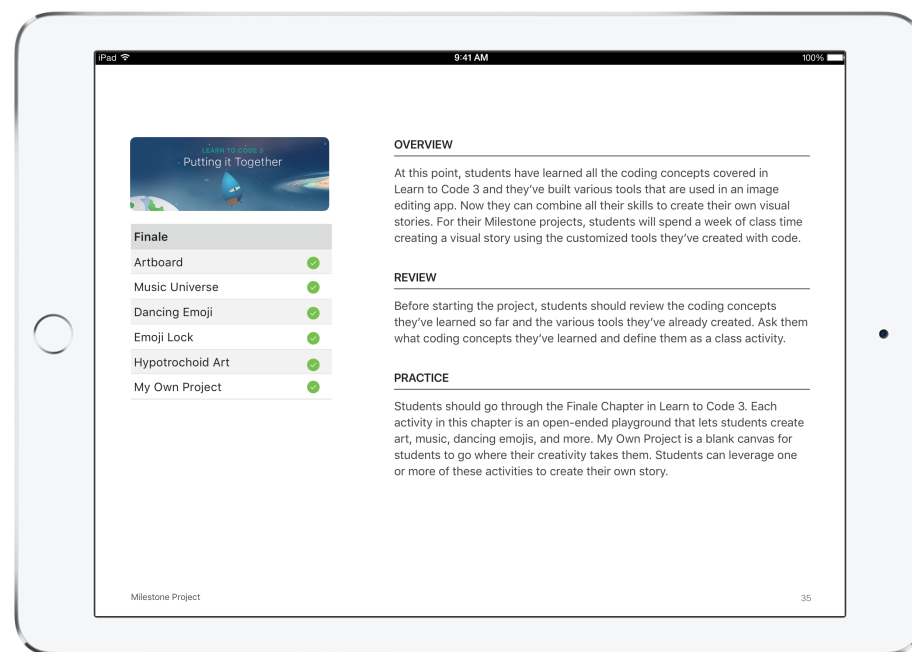
Aula 3 — Pensar como um editor: strings. Os alunos aprendem sobre strings em uma situação cotidiana e criam suas próprias ferramentas de texto no Swift Playgrounds. Eles também pesquisam sobre a influência de textos em histórias visuais.

Aula 4 — Pensar como um animador: manipuladores de evento.

Os alunos aprendem sobre manipuladores de eventos ao criar seus próprios jogos. Depois, criam as próprias ferramentas de ação no Swift Playgrounds e pesquisam sobre a influência da interatividade em histórias visuais.

Projeto-marco. Os alunos escrevem as próprias histórias visuais no Swift Playgrounds.

Design de apps. Os alunos passam por um ciclo de design, cujo foco é criar um protótipo, semelhante ao processo que os desenvolvedores profissionais de apps passam.



Outras informações

O Swift Playgrounds requer o iOS 10 e funciona nos seguintes modelos de iPad:

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

Baixe os recursos do Swift Playgrounds

- [Aprenda a Programar 1 e 2: curso do iTunes U](#)
- [Aprenda a Programar 1 e 2: Guia do professor](#)
- [Aprenda a Programar 3: Guia do professor](#)
- [App Swift Playgrounds](#)

Baixe os guias do curso Desenvolvimento de Apps com Swift

- [Introdução ao Desenvolvimento de Apps com Swift](#)
- [Introdução ao Desenvolvimento de Apps com Swift: Guia do professor](#)
- [Desenvolvimento de Apps com Swift](#)
- [Desenvolvimento de Apps com Swift: Guia do professor](#)

Baixe os recursos dos cursos Comece a Programar

- [Tynker](#)
- [codeSpark Academy](#)
- [Comece a Programar 1](#)
- [Comece a Programar 2](#)

Outros recursos

- Saiba mais sobre o [Swift Playgrounds](#).
- Saiba mais sobre o programa [Todo mundo pode programar](#).
- Saiba mais sobre a linguagem [Swift](#).
- Converse com outros educadores nos [Fóruns de desenvolvedores da Apple](#).

Alinhamento do currículo: Aprenda a Programar 1, 2 e 3

Este é o alinhamento preliminar dos Guias do professor do Swift Playgrounds Aprenda a Programar 1, 2 e 3 com o nível 2 dos Padrões de Serviços Provisórios de Ciência da Computação da Associação de Professores de Ciência da Computação (CSTA) K-12 de 2016. O alinhamento aborda os conceitos de algoritmos e programação dentro dos Padrões de Serviços Provisórios de Ciência da Computação da CSTA K-12 de 2016.

Padrões de Ciência da Computação da CSTA K-12 Nível 2 para o 6º ao 8º ano										
Padrão da CSTA	2-A-2-1 Comentários de colegas	2-A-7-2 Algoritmos: comparar	2-A-7-3 Atribuição	2-A-7-4 Algoritmos: interpretar	2-A-5-5 Artefatos de computação	2-A-5-6 Desenvolver programas	2-A-5-7 Variáveis	2-A-4-8 Parâmetros	2-A-3-9 Decomposição	2-A-6-10 Design iterativo
Capítulos do Swift Playgrounds	Comandos	●	●						●	●
	Funções	●	●		●	●		●	●	●
	Loops "for"	●	●		●	●		●	●	●
	Código condicional	●	●		●	●		●	●	●
	Operadores lógicos	●	●		●	●		●	●	
	Loops "while"	●	●		●	●		●	●	●
	Algoritmos	●	●		●	●		●	●	●
	Variáveis	●	●		●	●	●	●	●	●
	Tipos	●	●		●	●	●	●	●	●
	Inicialização	●	●					●	●	
	Parâmetros	●	●		●	●		●	●	
	Criação de mundo	●	●		●	●		●	●	
	Matrizes	●	●		●	●	●	●	●	●
	Coordenadas	●	●		●	●	●	●	●	●
	Eventos de toque	●	●	●	●	●	●	●	●	●
	Manipuladores de eventos	●	●	●	●	●	●	●	●	●
	Final	●	●		●	●	●	●	●	●

Legenda: ● Alinha-se ao padrão

Alinhamento do currículo: Aprenda a Programar 1 e 2

Este é o alinhamento preliminar dos Guias do professor do Swift Playgrounds Aprenda a Programar 1 e 2 com o nível 2 dos Padrões de Serviços Provisórios de Ciência da Computação da Associação de Professores de Ciência da Computação (CSTA) K-12 de 2016. O alinhamento aborda os conceitos de algoritmos e programação dentro dos Padrões de Serviços Provisórios de Ciência da Computação da CSTA K-12 de 2016.

Padrões de Ciência da Computação da CSTA K-12 Nível 2 para o 6º ao 8º ano										
Padrão da CSTA	2-A-2-1 Comentários de colegas	2-A-7-2 Algoritmos: comparar	2-A-7-3 Atribuição	2-A-7-4 Algoritmos: interpretar	2-A-5-5 Artefatos de computação	2-A-5-6 Desenvolver programas	2-A-5-7 Variáveis	2-A-4-8 Parâmetros	2-A-3-9 Decomposição	2-A-6-10 Design iterativo
Capítulos do Swift Playgrounds	Comandos	●	●		●				●	●
	Funções	●	●		●	●		●	●	●
	Loops "for"	●	●		●	●		●	●	●
	Código condicional	●	●		●	●		●	●	●
	Operadores lógicos	●	●		●	●		●	●	
	Loops "while"	●	●		●	●		●	●	●
	Algoritmos	●	●		●	●		●	●	●
	Variáveis	●	●		●	●	●	●	●	●
	Tipos	●	●		●	●	●	●	●	●
	Inicialização	●	●					●	●	
	Parâmetros	●	●		●	●		●	●	
	Criação de mundo	●	●		●	●		●	●	
	Matrizes	●	●		●		●	●	●	●

Legenda: ● Alinha-se ao padrão

Alinhamento do currículo: Aprenda a Programar 3

Este é o alinhamento preliminar do Guia do professor do Swift Playgrounds Aprenda a Programar 3 com o nível 2 dos Padrões de Serviços Provisórios de Ciência da Computação da Associação de Professores de Ciência da Computação (CSTA) K-12 de 2016. O alinhamento aborda os conceitos de algoritmos e programação dentro dos Padrões de Serviços Provisórios de Ciência da Computação da CSTA K-12 de 2016.

Padrões de Ciência da Computação da CSTA K–12 Nível 2 para o 6º ao 8º ano										
Padrão da CSTA	2-A-2-1 Comentários de colegas	2-A-7-2 Algoritmos: comparar	2-A-7-3 Atribuição	2-A-7-4 Algoritmos: interpretar	2-A-5-5 Artefatos de computação	2-A-5-6 Desenvolver programas	2-A-5-7 Variáveis	2-A-4-8 Parâmetros	2-A-3-9 Decomposição	2-A-6-10 Design iterativo
Capítulos do Swift Playgrounds	Coordenadas	●	●		●	●	●	●	●	●
	Eventos de toque	●	●	●	●	●	●	●	●	●
	Manipuladores de eventos	●	●	●	●	●	●	●	●	●
	Final	●	●		●	●	●	●	●	●

Legenda: ● Alinha-se ao padrão

Os recursos estão sujeitos a mudanças. Alguns recursos podem não estar disponíveis em todas as regiões ou idiomas.