



Clube de Programação com Swift



Kit do Swift Playgrounds

Bem-vindos ao Clube de Programação com Swift!

Aprender a programar ensina a resolver problemas e a trabalhar em conjunto de maneiras criativas. Também ajuda a criar apps que colocam suas ideias em prática.

Os Clubes de Programação com Swift são uma maneira divertida de aprender a programar e criar apps. Atividades criadas usando Swift, a linguagem de programação da Apple, ajudam você a colaborar à medida que aprende a programar, criar protótipos de apps e pensar sobre como a programação pode fazer a diferença no mundo à sua volta.

Não é necessário ser professor ou especialista em programação para organizar um Club de Programação com Swift. Os materiais são individualizados, então, todos os integrantes do clube podem aprender juntos, mas cada um no seu próprio ritmo. E todos vocês poderão celebrar as ideias e criações do clube organizando um evento de demonstração de apps para a sua comunidade.

Esse kit está organizado em três seções:



Primeiros passos

Tudo que é necessário para iniciar um Clube de Programação com Swift.



Aprendizagem e criação

Dicas e atividades para desenvolver sessões do clube.



Comemoração

Recursos úteis para planejar e conduzir a apresentação de um app em sua comunidade.

Clubes de Programação com Swift

Programação em blocos | Entre 8 e 11 anos de idade

Aprenda o básico em programação usando apps visuais no iPad.



Swift Playgrounds | A partir de 11 anos de idade

Use a linguagem Swift para aprender os fundamentos da programação com o Swift Playgrounds no iPad.



Xcode | A partir de 14 anos de idade

Aprenda a desenvolver apps no Xcode no Mac.



Primeiros passos



1. Baixe os materiais do clube.

Use o AirDrop para compartilhar esses dois guias com integrantes do clube na primeira reunião de vocês. Eles também são incluídos neste documento.



Atividades de programação

Aprenda conceitos de programação com estas atividades divertidas de colaboração e resolva puzzles com o app Swift Playgrounds no iPad.

[Baixe Atividades de programação do Swift Playgrounds >](#)



Diário de design de apps

Explore o processo de design de apps com este diário do Keynote. Troque ideias, planeje, crie protótipos e avalie as ideias de app do seu clube.

[Baixe o Diário de design de apps do Swift Playgrounds >](#)



2. Confira os recursos de tecnologia.

Antes da primeira reunião, confira se vocês têm o seguinte:

- **iPad.** iPad mini 2 ou posterior, iPad Air ou posterior ou iPad Pro com iOS 11 ou posterior. É melhor se cada um tiver seu próprio dispositivo, mas também é possível compartilhar dispositivos para programação em conjunto.
- **App Swift Playgrounds.** [Baixe o Swift Playgrounds >](#)
- **Playgrounds do Aprenda a Programar 1 e 2.** Baixe esses playgrounds pelo app Swift Playgrounds.
- **Keynote.** Você usará o app Keynote no iPad para seus protótipos de apps.
- **Materiais do Clube de Programação com Swift.**



3. Planeje.

Aqui estão alguns pontos a serem considerados:

- Quem são os integrantes do clube? Quais são os interesses deles? Eles têm experiência em programação ou são totalmente novatos?
- Com que frequência seu clube vai se reunir? Se vocês estiverem planejando uma colônia de férias, quantas horas de atividades de programação vocês terão?
- Qual tecnologia está disponível para o clube?
- Quais são os objetivos do clube?



4. Faça o trabalho de divulgação.

Divulgue o Clube de Programação com Swift. Aqui estão algumas ideias e recursos para atrair novos integrantes para o clube:

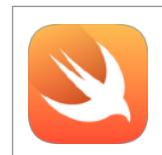
- **Apresente seu clube.** Use e-mail, redes sociais, internet, folhetos ou divulgação “boca a boca” para divulgar seu clube à comunidade.
- **Organize uma reunião informativa.** Pergunte a possíveis integrantes do clube quais são os interesses deles e que tipos de apps eles gostariam de criar. Fale sobre ideias para fazer a apresentação do design de apps e como envolver os integrantes. Você também pode compartilhar na internet um vídeo curto sobre o clube.

Esses itens podem ajudar você a promover e personalizar seu Clube de Programação com Swift:

- **Pôsteres.** [Baixe este modelo gratuito](#) e personalize-o para criar seu próprio pôster. Imprima-o e faça a divulgação, ou crie um pôster digital para compartilhar na internet. Não se esqueça de incluir detalhes de quando e onde o clube se reunirá e como participar.
- **Adesivos e camisetas.** Use estes [adesivos do Clube de Programação com Swift](#) para ajudar a promover seu clube. Camisetas são uma excelente maneira de reconhecer os integrantes que participam de eventos de demonstração de apps. Baixe o [modelo de camiseta do Clube de Programação com Swift](#) para criar camisetas para os integrantes.



Pôster do Clube de Programação com Swift



Adesivo do Clube de Programação com Swift



Camiseta do Clube de Programação com Swift

Aprendizagem e criação



Os materiais do clube foram desenvolvidos para você combinar atividades de programação e criação de apps. Você também pode adicionar sessões que atendam aos interesses dos integrantes. Veja um exemplo de programação para 30 sessões de uma hora do clube.



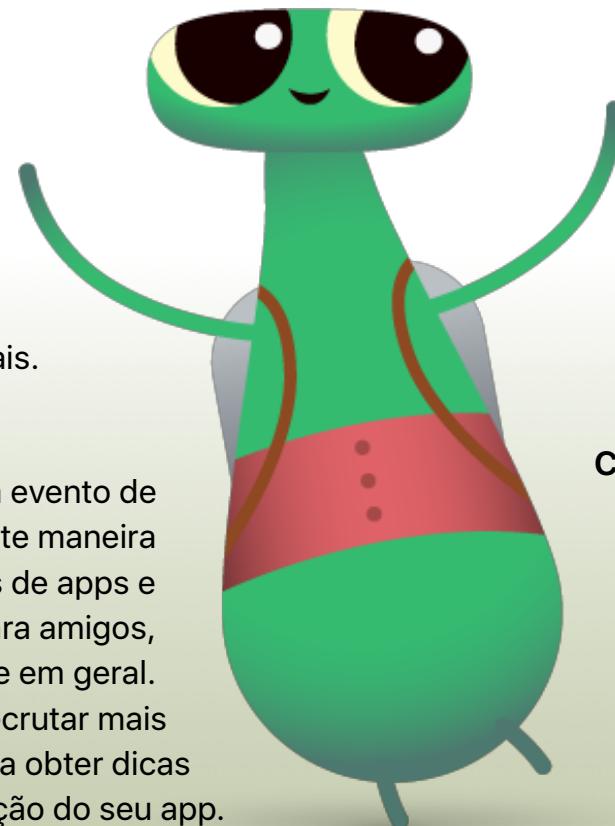
Considere adicionar sessões para expandir atividades de programação e design de apps, como criar um percurso de obstáculos para drones ou criar um desafio de resgate para um robô. Para dar início à troca de ideias sobre design de apps, você pode até mesmo adicionar palestrantes convidados ou visitas.



Crie uma equipe de liderança. Ter um grupo que ajude na liderança do clube pode torná-lo muito mais fácil e divertido. Quais integrantes do clube têm potencial de liderança? Pense em nomear no clube representantes para eventos, programação, design de apps e muito mais.

Aprendam juntos. Os líderes do clube não precisam saber tudo. Ajude os integrantes a desenvolver suas habilidades de pesquisa e solução de problemas e os incentive a ajudar os demais.

Mostre seu conhecimento. Um evento de apresentação do app é uma excelente maneira de promover o clube, ideias de apps e habilidades de programação para amigos, familiares, professores e a comunidade em geral. Ele pode até mesmo ajudar você a recrutar mais integrantes. Veja a [página 13](#) para obter dicas sobre como fazer a apresentação do seu app.



Compartilhe ideias. Alguns integrantes estarão interessados na criação de jogos. Outros podem querer criar apps para ajudar pessoas, aprender a linguagem Swift ou controlar robôs. Pense em como os integrantes podem trabalhar juntos em projetos que considerem importantes.

Combine. E recombine. Às vezes, os integrantes que estão mais avançados podem deixar os outros para trás. Veja se eles podem ajudar os iniciantes para que a programação fique no mesmo nível. Ensinar a outra pessoa é uma excelente maneira de aprender!



Atividades de programação do Swift Playgrounds

Clube de Programação com Swift

Swift Playgrounds

Atividades de programação

Pensar como um computador —
Comandos e sequências

Introdução (11 minutos)

Introdução

Prática (30 minutos)

Pra pensar:

Vá mais além

Dash (10 minutos)

Pra pensar:

Atividades de programação: criadas em torno do Swift Playgrounds, essas atividades de colaboração apresentam habilidades e conceitos fundamentais de programação.

Conceitos de programação: em cada atividade, os integrantes do clube aprenderão um conceito fundamental de programação e o explorarão em um contexto do cotidiano. Depois, aplicarão o conceito de programação para resolver puzzles no Swift Playgrounds.

Vá mais além: cada conceito de programação tem duas atividades “Vá mais além”. A primeira atividade aprofunda a compreensão do conceito de programação e estimula a comunicação e o trabalho em equipe. Os integrantes usam o iPad para aplicar sua compreensão em um projeto criativo.

A segunda atividade opcional desafia os integrantes do clube a aplicar o conceito em um playground das seções Desafios, Pontos de Partida e Assinaturas do Swift Playgrounds. Algumas atividades exigem dispositivos conectados específicos.

Todo mundo pode programar

Aprenda a Programar 1 e 2
Guia do professor

Baixe o Swift Playgrounds:
Aprenda a Programar 1 e 2:
Guia do professor >

Swift Playgrounds

Dicas para aprender com o Swift Playgrounds



Explore os puzzles primeiro. Incentive os integrantes do clube a dar um zoom e girar o mundo de Byte na visualização dinâmica para que possam ter uma boa ideia do que precisam fazer. Eles também podem visualizar em tela cheia ao tocar e manter pressionada a partição entre as duas janelas e depois arrastando para a esquerda.

Explore os puzzles. Eles ficam mais complexos. Os integrantes do clube podem dividir um puzzle em partes para ajudá-los a pensar em todas as etapas para resolvê-lo. Eles podem usar o Pages ou Notas para planejar e escrever as etapas antes de programar.

Monte um help desk. Mantenha um espaço no qual especialistas do clube possam oferecer suporte a seus colegas.



Resolva de várias maneiras. Cada puzzle tem muitas soluções. Incentive quem terminar antes a pensar de diferentes maneiras para resolver os puzzles. Pensar com flexibilidade e comparar diferentes soluções podem ajudá-los a melhorar suas habilidades de pensamento crítico.

Programe em duplas. Peça para os integrantes do clube tentarem trabalhar juntos em um iPad. Eles podem trocar ideias sobre como resolver os puzzles e revezar a vez na programação.

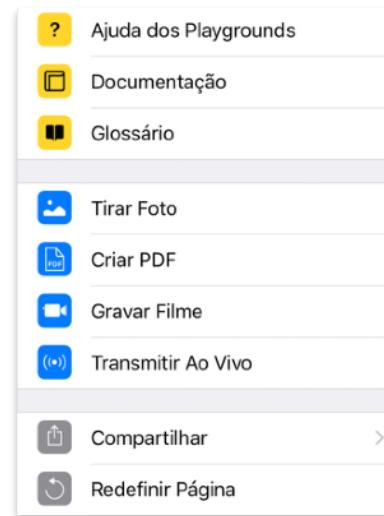
Use recursos de acessibilidade. O Swift Playgrounds funciona bem com os recursos de acessibilidade integrados do iOS, assim qualquer pessoa pode aprender a programar. Por exemplo, os programadores podem inverter as cores, ativar a escala de cinzas e aumentar o zoom para ajustar a visibilidade.

Explore o Swift Playgrounds



Biblioteca de snippets.

Para reduzir a digitação,
toque na barra de
ferramentas para acessar
a biblioteca de snippets
e arrastar rapidamente
os fragmentos de código
mais usados.



Escolha um personagem. Personalize sua experiência ao tocar no personagem para escolher outro.

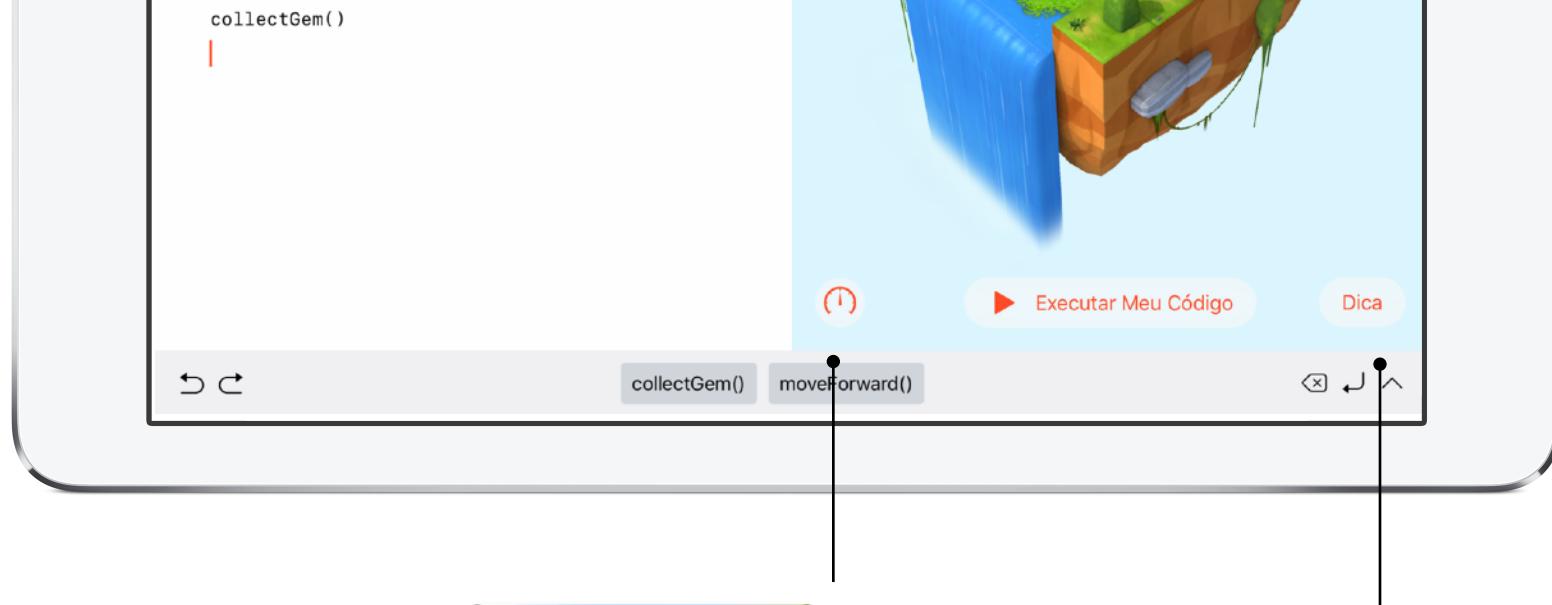
Atribuindo Comandos < > + ...

Objetivo: usar os comandos Swift para ordenar que o Byte se move e coleта uma joia.

O seu personagem, Byte, adora coletar joias, mas ele não consegue fazer isso sozinho. Neste primeiro puzzle, você terá que escrever **comandos** Swift para movimentar o Byte pelo mundo do puzzle e coletar uma joia.

- 1 Procure a joia no mundo do puzzle.
- 2 Digite a combinação certa dos comandos `moveForward()` (ir para frente) e `collectGem()` (coletar joia).
- 3 Toque em Executar Meu Código.

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



Dica. Esse recurso contém sugestões para ajudar os alunos. Ele também revela, dentro do possível, a solução de um puzzle, embora os programadores não possam simplesmente copiar e colar a solução. Para continuar, eles ainda precisam concluir as etapas e formular os códigos para poder avançar.

- » Executar Mais Rápido
- » Executar Rápido
- » **Executar Meu Código**
- » Passo a Passo
- » Passo a Passo Devagar
- » X

Controle a velocidade. Aumente ou reduza a velocidade do código.

Destaque o código à medida que ele é executado. Use a função “Passo a Passo” para destacar cada linha de código durante a execução a fim de compreender melhor o que o código está fazendo.



Diário de design de apps



Os programadores usam esse diário no Keynote para aprender sobre recursos do app e para criar um app que resolva um problema da comunidade.

Eles trabalham em pequenas equipes para trocar ideias e planejar a solução do app; depois, criam um protótipo do app no Keynote.

O diário orienta os programadores no processo de avaliação de seus designs e iteração de seus protótipos, exatamente como fazem os designers de apps profissionais.

Os integrantes do clube podem criar um vídeo ou apresentação do app de três minutos e celebrar o trabalho em uma demonstração do design do app.

Comemoração



Apresentação do design de apps

O processo de design de apps e apresentação são excelentes oportunidades para envolver a comunidade mais ampla e explorar o potencial de apps para resolver problemas atuais. A apresentação também é a maneira perfeita de exibir os talentos dos integrantes do clube!

1. Planeje o grande evento. Defina uma data para a apresentação e convide alunos, professores, pais e integrantes da comunidade.

Reserve alguns minutos para que cada equipe apresente seu app e conduza uma breve sessão de perguntas e respostas. Se o grupo for grande, pode ser interessante dividir o clube em duas rodadas nas quais os integrantes podem assistir às apresentações uns dos outros.

Considere concluir o evento com uma apresentação de slides divertida de fotos tiradas durante as sessões do clube.



2. Crie prêmios. A competição amigável pode ser um excelente motivador. Inspire integrantes do clube ao oferecer prêmios que reconheçam pontos fortes específicos no design de apps. Considere prêmios para:

- Melhor engenharia
- Melhor inovação
- Melhor design
- Melhor apresentação

Você também pode incentivar a participação do público com um prêmio “Escolha do Público”.



Você pode baixar e modificar este [certificado](#) para diferentes prêmios.

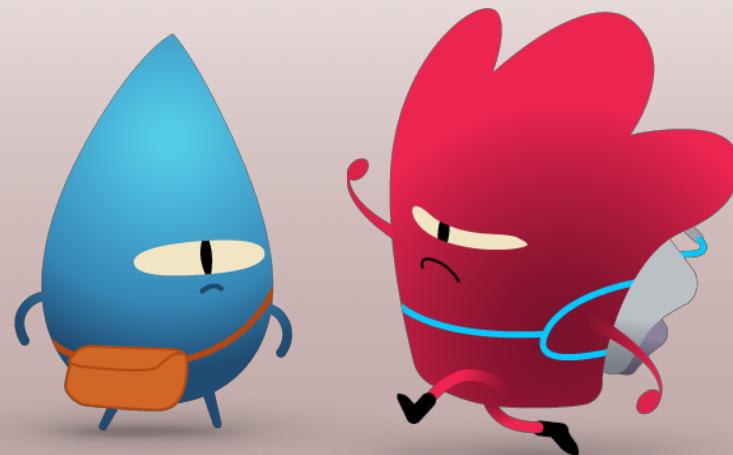


3. Recrute jurados e orientadores. Jurados e orientadores podem ser professores ou funcionários da escola, alunos com experiência em programação, especialistas do setor de desenvolvimento ou design, integrantes da diretoria da escola, líderes da comunidade local ou pessoas que se beneficiariam da ideia do app.

Os jurados não precisam esperar até a apresentação para conhecer o clube. Considere convidá-los como palestrantes para compartilharem sua experiência quando os alunos estiverem na fase de troca de ideias ou planejamento do design de apps.

4. Escolha um vencedor. Os jurados podem usar a ficha de avaliação na próxima página para ajudá-los a avaliar as apresentações dos apps e fazer comentários. Você também pode compartilhar a ficha de avaliação com os programadores antes da apresentação como parte da fase de avaliação do processo de design de apps.

5. Compartilhe e inspire. Gravar as apresentações é uma boa ideia. Compartilhe-as com a comunidade e crie um vídeo com os destaques para inspirar futuros integrantes do clube.



Ficha de avaliação

[Baixe >](#)

| Categoria | Básico (1 ponto) | Intermediário (2 pontos) | Proficiência (3 pontos) | Domínio (4 pontos) | Pontos |
|--|--|--|--|--|-----------------------------|
| Conteúdo da apresentação | Compartilha informações básicas, como objetivo e público-alvo. | Fornece explicação clara do objetivo e design do app, e como ele atende às necessidades do usuário. | Fornece explicação clara e convincente do problema a ser resolvido, da demanda de mercado, do público-alvo e de como o app foi criado para atender às necessidades do usuário. | Faz uma apresentação persuasiva sustentada por fatos que demonstram como o app atende às necessidades do usuário, vai além delas ou as redefine. | |
| Execução da apresentação | Informativa. Apresentação feita por um integrante da equipe. | Confiante e animada. Apresentação feita por mais de um integrante da equipe. | Envolvente e com bom uso de recursos visuais para sustentar a história. A equipe destaca as contribuições de cada integrante. | História contada de forma criativa e memorável. Auxílio visual envolvente e alternâncias harmoniosas entre os integrantes da equipe. | |
| Interface do usuário (UI) | Telas coerentes com o objetivo do app. | Design claro e funcional com elementos conhecidos. O protótipo sustenta as tarefas básicas de usuário. | Design elegante, conciso e agradável com uso consciente de cores, layouts e facilidade de leitura. O protótipo facilita a navegação do usuário pelo app. | O design permite que o usuário interaja facilmente com o conteúdo. O protótipo usa animações, cores e layouts para criar uma experiência transparente e envolvente. | |
| Experiência do usuário (UX) | Intenção clara. Os usuários conseguem atingir um ou mais objetivos. | Navegação padrão e estável. Caminho intuitivo pelo conteúdo do app. | Adaptável às necessidades do usuário. Trata acessibilidade, privacidade e segurança. | Inovador, encantador e surpreendente. Oferece aos usuários uma nova experiência que diferencia o app de seus concorrentes. | |
| Conceitos de programação | Algum nível de ligação entre a funcionalidade do app e o código por trás dele. | Explicação de como os conceitos gerais de programação, como tipos de dados, lógica condicional ou eventos de toque, se relacionam com o app. | Descrição de tarefas específicas de programação necessárias para criar o app. Demonstração de como o código é a força motriz da funcionalidade do app. | Explicação da arquitetura, estrutura de dados, algoritmos e recursos do app. Discussão da tomada de decisões ao desenvolver esta abordagem. | |
| Avaliação técnica (opcional) <i>Para protótipos funcionais de apps no Xcode. Os jurados devem conhecer a linguagem Swift e as melhores práticas de desenvolvimento de apps para iOS.</i> | A linguagem Swift é executada em exemplos específicos. O código é básico, sem abstração. | O código é executado sem erros em todos os casos. O código é básico, com algum grau de abstração. | O código é organizado seguindo as convenções de nomenclatura da linguagem Swift. Alto grau de abstração. As diretrizes de desenvolvimento para iOS são seguidas. | O código é bem documentado com comentários. Uso eficiente dos recursos da linguagem Swift. Emprega organização, como o uso do Model-View-Controller (modelo-visualização-controlador). | |
| Comentários: | | | | | |
| | | | | | 0 Pontuação total |



Clube de Programação com Swift

Swift Playgrounds

Certificado de Conclusão

Concedido a

Por



Assinatura

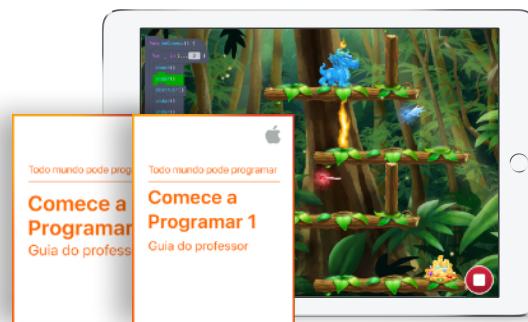
Data

Vá mais além

O Clube de Programação com Swift é apenas o início da sua jornada em programação. O currículo do Todo mundo pode programar oferece recursos divertidos e de apoio para que os programadores passem da aprendizagem das noções básicas no iPad para a criação de apps reais no Mac.

E você não precisa parar nas atividades do clube. Os abrangentes Guias do professor também possibilitam que os professores levem a programação para a sala de aula, com aulas passo a passo e alinhadas ao currículo para alunos do jardim de infância à faculdade.

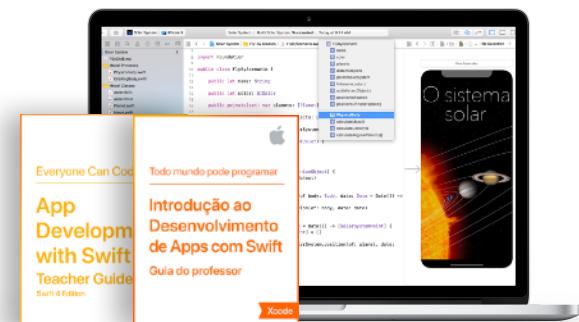
[Veja todos os recursos do Todo mundo pode programar >](#)



Saiba mais sobre o currículo
[Comece a Programar >](#)



Saiba mais sobre o currículo do
[Swift Playgrounds >](#)



Saiba mais sobre o currículo
[Desenvolvimento de Apps com Swift >](#)



© 2018 Apple Inc. Todos os direitos reservados. Apple, o logotipo da Apple, AirDrop, iPad, iPad Air, iPad mini, iPad Pro, Keynote, Mac, Pages e Xcode são marcas comerciais da Apple Inc., registradas nos EUA e em outros países. Swift e Swift Playgrounds são marcas comerciais da Apple Inc. Os demais produtos e nomes de empresas aqui mencionados provavelmente são marcas comerciais das suas respectivas empresas. Novembro de 2018