



Swift Coding Club

Develop in Swift



Swift Coding Clubへようこそ

コーディングを学ぶと、創造力を発揮しながら協力し合って問題を解決する力が身につきます。また、自分のアイデアを形にするアプリケーションを作れるようになります。

Swift Coding Clubでは、コーディングとアプリケーションデザインについて楽しく学ぶことができます。Appleのプログラミング言語であるSwiftを使い、チームで協力しながらアクティビティに取り組み、コードの書き方やアプリケーションのプロトタイプ(試作品)を作る方法を学びます。また、コードによって自分たちの暮らしをどのように変えることができるのか、考えるきっかけにもなります。

教師でなくても、コーディングに詳しくなくても、Swift Coding Clubの運営は可能です。教材は各自が自分のペースで学べるようになっており、指導する立場の人ともメンバーと一緒に学びながら進めることができます。そして、地域でアプリケーションコンテストを行い、メンバーのアイデアやデザインを讃えましょう。

このキットには3つのセクションがあります。



開始準備

Swift Coding Clubを
スタートするために
必要な準備について
説明します。



指導計画とヒント

活動を計画するための
ヒントやアクティビティの内容
について紹介します。



作品発表

地域でアプリケーション
コンテストを企画・開催
するのに役立つ資料を
紹介します。

コーディングに関する リソース

Swift Coding Clubでは、コーディングを教えるための様々なリソースをご用意しています。これらのリソースを使えば、iPadでの基礎学習からMacでの本格的なアプリケーション開発へと学習を進めていくことができます。



Everyone Can Code | 10歳程度から

iPadのSwift PlaygroundsでSwiftを使って
プログラミングの基礎を学びます。[さらに詳しく >](#)



Develop in Swift | 15歳程度から

MacのXcodeでアプリケーション開発を
学びます。[さらに詳しく >](#)

開始準備



1. クラブの教材をダウンロードする

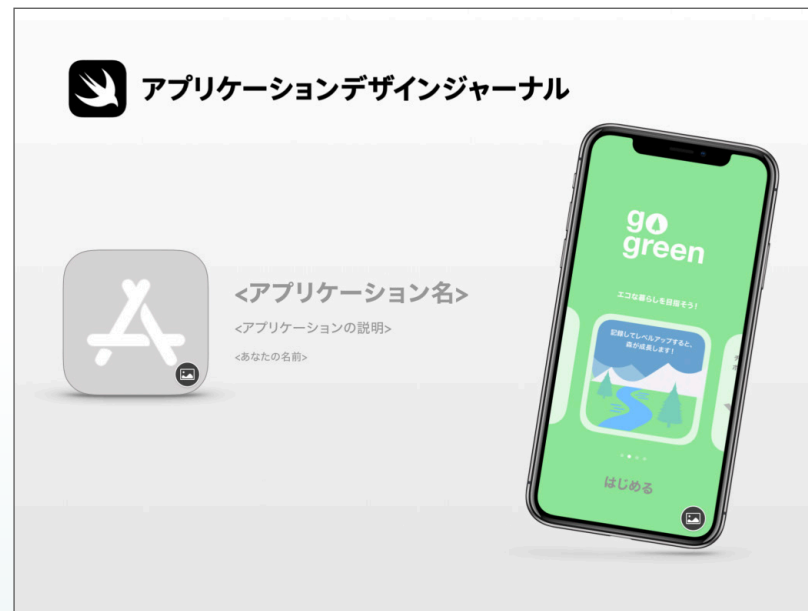
クラブの初回のミーティングで、AirDropを使って以下の2つの教材をメンバーに配布します。
このガイドにも付属しています。



アプリケーション開発プロジェクト

アプリケーション開発プロジェクトガイドでは、プロが使うのと同じツール、テクニック、概念が自主的に学べます。
MacのXcodeプレイグラウンドでコードを書くことを通して、プログラミングの概念を学びましょう。

[アプリケーション開発プロジェクトをダウンロード >](#)



アプリケーションデザインジャーナル

アプリケーションデザインのプロセスを詳しく学ぶためのKeynoteジャーナルです。クラブのメンバーが考えたアプリケーションのアイデアをブレインストーミングし、計画し、プロトタイプを作り、評価します。

[アプリケーションデザインジャーナルをダウンロード >](#)



2. テクノロジーを確認する

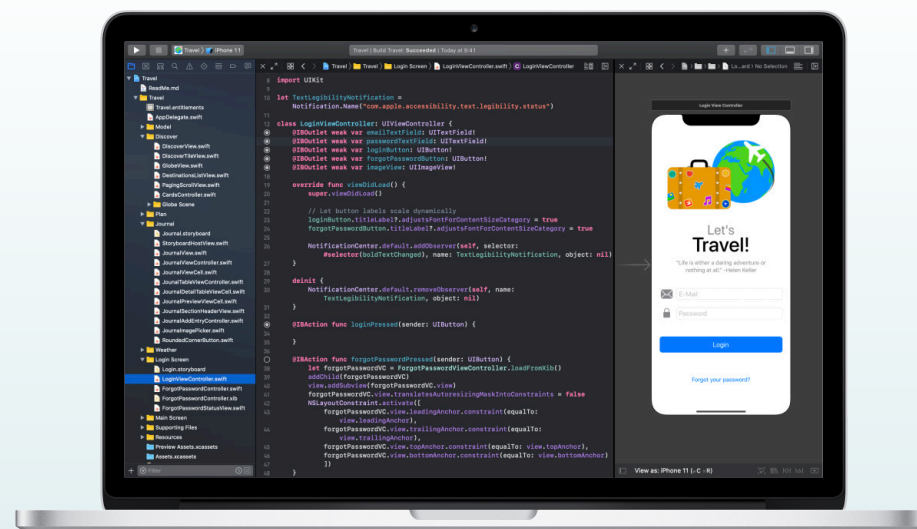
初回のミーティングまでに、必ず以下のものを用意してください。

- **Mac**. macOS MojaveまたはHigh Sierraを搭載するMacが必要です。1人1台ずつ用意するのが理想的ですが、何人かでMacを共有して一緒にコーディングに取り組むこともできます。
- **Xcode**. Appleが提供する無料のMacアプリケーションで、あらゆるMacアプリケーションやiOSアプリケーションの開発に使用されています。素晴らしいアプリケーション体験を生み出すためのあらゆるツールが揃っています。Xcode 9はSwift 4に、Xcode 10はSwift 4.2に対応しています。
- 「**Swiftによるアプリケーション開発：入門編**」。Appleが提供する無料の教材で、初心者向けのコーディングプロジェクトが8つ用意されています。
- **Keynote**. アプリケーションのプロトタイプ作成には、iPadもしくはMacのKeynoteアプリケーションを使います。
- **Swift Coding Clubの教材**。

3. 計画を立てる

次の点を考慮してください。

- クラブにはどのようなメンバーが所属していますか？メンバーは何に興味をもっていますか？コーディングの経験はありますか？それとも完全な初心者ですか？
- どのくらいの頻度でクラブ活動を行いますか？夏合宿やイベントを計画している場合、コーディングアクティビティに何時間確保できますか？
- クラブではどのようなテクノロジーを利用できますか？
- クラブの目標は何ですか？





4. 情報発信する

Swift Coding Clubのことを多くの人に知らせましょう。クラブの新メンバーを募集するためのアイデアや資料を紹介します。

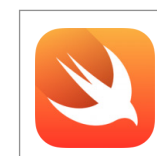
- **クラブ発足の告知。**Eメール、ソーシャルメディア、ウェブサイト、チラシ、口コミなどを利用して、地域社会にクラブの紹介をします。
- **説明会を開く。**クラブへの参加を考えている人に、興味のあることや、作ってみたいアプリケーションの種類などを聞いてみましょう。アプリケーションコンテストの開催予定や、参加方法についても説明しましょう。クラブの簡単な紹介ビデオをオンラインで共有するのもよいでしょう。

Swift Coding Clubの宣伝やカスタマイズには、以下のアイテムも活用してください。

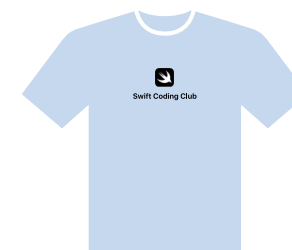
- **ポスター。**[無料テンプレートをダウンロード](#)し、カスタマイズして独自のポスターを作成してください。印刷して掲示するか、デジタル版ポスターをオンラインで見られるようにします。クラブ活動の日程や場所、参加方法などの詳細も必ず記載しましょう。
- **ステッカーとTシャツ。**[Swift Coding Clubステッカー](#)をクラブの宣伝に利用してください。Tシャツもおすすめです。おそろいのTシャツを着れば、アプリケーションコンテストのときにクラブのメンバーを見分けやすくなります。[Swift Coding ClubのTシャツテンプレート](#)をダウンロードして、メンバー用のTシャツを作ってください。



Swift Coding Clubのポスター



Swift Coding Clubのステッカー



Swift Coding ClubのTシャツ

指導計画とヒント



クラブの教材は、コーディングとアプリケーションデザインのプロジェクトを交互に実施するデザインになっています。メンバーの興味に合わせてセッションを追加することもできます。1時間のセッションを30回行うスケジュール例を以下に紹介します。



アプリケーションデザインやコーディングのプロジェクトをさらに発展させたい場合は、AR（拡張現実）やVR（仮想現実）について調べるセッションや、身近で活動しているアプリケーションデザイナーから話を聞くセッションなどを追加してみてもいいかもしれません。また、アプリケーションデザインのブ레인ストーミングを活発にするため、ゲストスピーカーを招待したり、課外授業を実施するのもいいでしょう。



リーダーチームを編成する。クラブの指導を手伝ってくれるメンバーを選んでリーダーチームを作っておくと、活動が円滑になり、さらに盛り上がります。どのメンバーがリーダーに向いていそうですか？イベント、コーディング、アプリケーションデザインなどをそれぞれ担当する係を決めておくのもよいでしょう。

一緒に学ぶ。クラブの指導者が何もかも知っている必要はありません。メンバーが自分で調べたり問題解決するスキルを伸ばせるようサポートし、お互いに助け合って活動するよう促しましょう。

披露する。アプリケーションコンテストは、クラブの活動内容や、アプリケーションのアイデア、コーディングスキルを、メンバーの友達、家族、教師、地域の人たちにアピールする絶好のチャンスです。これをきっかけに新しいメンバーが増えるかもしれません。コンテストを開催するためのヒントについては、[11ページ](#)を参照してください。



アイデアを共有する。ゲームを作りたいメンバーもいれば、人の役に立つアプリケーションを作りたいメンバー、Swiftについて学びたいメンバー、ロボットを動かしたいメンバーもいるでしょう。みんなが関心をもてるプロジェクトに、協力し合って取り組める方法を考えてみましょう。

混合チームを結成する。先に進んでいるメンバーが、ほかのメンバーを置いてけぼりにしてしまうこともあります。進みの早いメンバーと初心者メンバーをペアにして、ペアプログラミングを試してみましょう。誰かに教えることで、理解も深まります。



アプリケーション開発プロジェクト



8つのプロジェクトを通して、初心者向けにコーディングの概念とアプリケーション開発の基本を紹介します。プロジェクトごとにそれぞれ異なるアプリケーションの要素を取り上げており、実際に動くアプリケーションを1つずつ作成しながら、段階的に学んでいけるようになっていますので、最終的に、自分独自のアプリケーションを開発するスキルが身につきます。



8つのプロジェクトをすべて完了する必要はありません。ここでのねらいは、アプリケーションの様々な側面を紹介し、自分のアプリケーションをデザインしてみたいという気持ちを起こさせることです。



プロジェクトに取り組むには、「Swiftによるアプリケーション開発：入門編」コースが必要です。必修レッスンを完了してからプロジェクトをスタートさせましょう。



「Swiftによるアプリケーション開発：入門編」をダウンロード >



内容の詳細や発展させるためのヒントを得る
「Swiftによるアプリケーション開発：入門編 教師用ガイド」をダウンロード >

Xcodeで学習する際のヒント

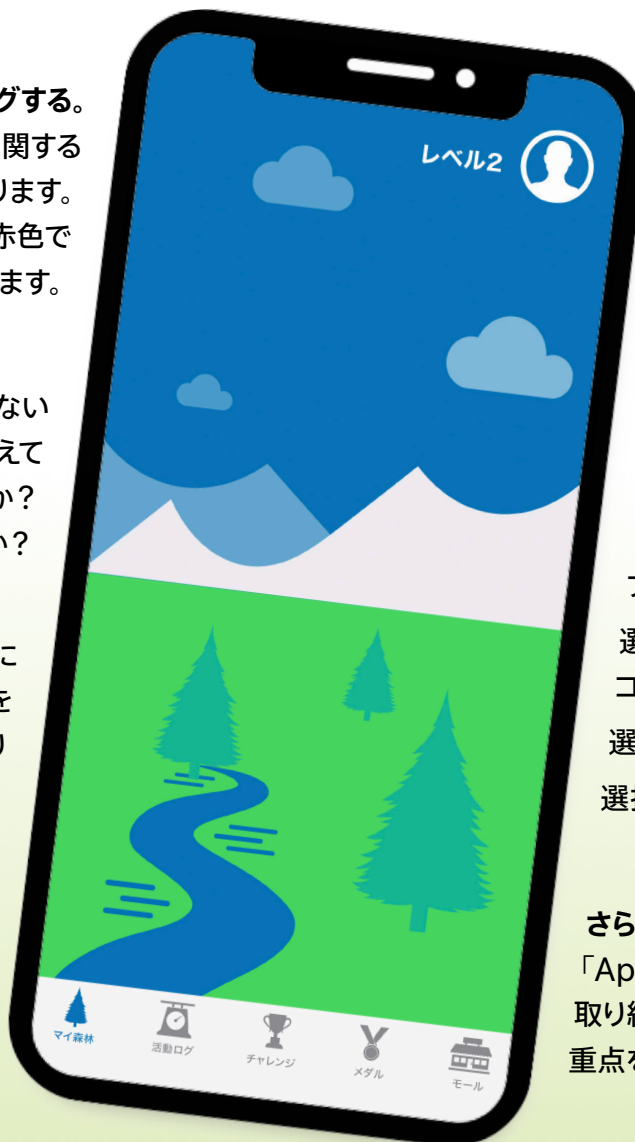


コンソールを確認してデバッグする。
コンソールを見ると、エラーに関する有益な情報が見つかることがあります。エラーが発生したコード行は赤色でハイライトされます。

手を止めて考える。バグは避けられないものです。手を止めて、問題について考えてみましょう。どのような症状が発生しましたか？どの時点まで正常に動いていましたか？

コードの書き方は人それぞれ。お互いにコードを見せ合って気付いたことを指摘したり、協力してデバッグしたりするとよいでしょう。

ヘルプデスクを作る。進みの早いメンバーが、ほかのメンバーをサポートできる場所を確保します。



Xcodeの環境設定を詳しく調べる。メニューバーで「Xcode」>「Preferences」と選択して、テキスト編集などの環境設定を設定します。デベロッパアカウントの追加、ナビゲーションやフォントのカスタマイズ、イベント発生時の動作設定などが可能です。

キーボードショートカットをマスターする。

プロジェクトをビルドする： $\text{⌘} + B$

プロジェクトをビルドして実行する： $\text{⌘} + R$

選択したコード行をコメントアウトまたはコメント解除する： $\text{⌘} + /$

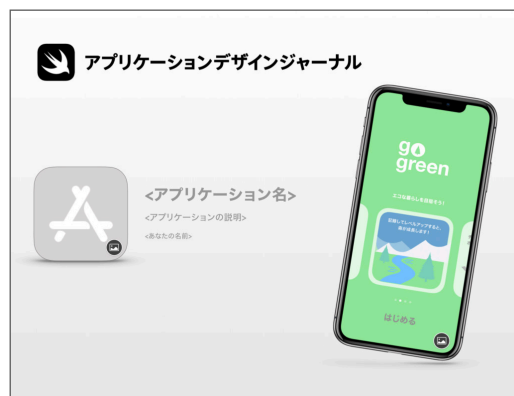
選択したコード行のインデントを左にずらす： $\text{⌘} + [$

選択したコード行のインデントを右にずらす： $\text{⌘} +]$

さらに進める。入門コースを完了したメンバーは「App Development with Swift」コース(英語)に取り組んで、Xcodeを使ったプロトタイプのコーディングに重点を置いて学習することもできます。



アプリケーションジャーナル



このKeynoteジャーナルを使ってアプリケーションの機能について学び、身近な問題を解決するアプリケーションをデザインします。



少人数のチームでアプリケーションについてブレンストーミングし、計画を立ててから、プロトタイプをKeynoteで作成します。



アプリケーションジャーナルでは、アプリケーションデザインを評価したり、プロトタイプを改善するプロセスが一通り紹介されています。これはプロのアプリケーションデザイナーも使っている手法です。

アプリケーションを3分でアピールするプレゼンテーションまたはビデオを作成し、アプリケーションコンテストで披露します。



作品発表



成果を讀えましょう

アプリケーションデザインのプロセスとその成果を見せるアプリケーションコンテストは、地域社会の幅広いメンバーを巻き込んで、現代の社会問題を解決することができる、アプリケーションの可能性について一緒に考える絶好のチャンスです。また、クラブのメンバーたちの才能を披露する、またとない機会でもあります。

1. 盛大なイベントを企画する。コンテストの日程を決め、参加者、家族、地域のメンバーを招待しましょう。

チームごとに自分たちのアプリケーションのプレゼンテーションを行って、質疑応答の時間も取ります。クラブの規模が大きい場合は、前半と後半に分けて、メンバーがお互いのプレゼンテーションを見ることができるようにしてください。

クラブのセッション中に撮った写真を楽しいスライドショーにして、最後に上映するのもおすすめです。

2. 賞を設定する。楽しく競い合えば、学習意欲が高まります。アプリケーションデザインで特に力を発揮したメンバーを表彰して、やる気を刺激しましょう。次のような賞を検討します。

- 最優秀エンジニアリング賞
- 最優秀アイデア賞
- 最優秀デザイン賞
- 最優秀プレゼン賞

特別賞を追加し、観客に投票してもらうこともできます。



この賞状をダウンロードし、カスタマイズして様々な賞状を作ってください。



3. 審査員やメンターを依頼する。審査員やメンターには、アプリケーション開発やデザイン業界のエキスパート、自治会長、アプリケーションの対象ユーザーなどが適任です。審査員にはアプリケーションコンテストよりも前にクラブ活動を見学してもらうこともできます。審査員をゲストスピーカーとして招き、アプリケーションデザインのブレインストーミングや計画の段階で、プロの視点からの助言をお願いしてもいいでしょう。

4. 優勝者を選ぶ。審査員がアプリケーションのプレゼンテーションを評価し、フィードバックを提供する時は、次のページにあるルーブリックを活用してください。このルーブリックは、アプリケーションコンテストより前に、アプリケーションデザインプロセスの「評価」の段階でメンバーに説明しておいてもよいでしょう。

5. シェアしてインスピレーションを広げる。プレゼンテーションを録画してもよいでしょう。ビデオを幅広いコミュニティで共有しましょう。また、ハイライトを紹介するバージョンを作成して、将来のメンバーにインスピレーションを与えるために活用してください。



評価用ルーブリック

ダウンロード >

チーム名: _____

	もう一歩(1点)	できました(2点)	よくできました(3点)	非常によくできました(4点)	得点
プレゼンテーションの内容	アプリケーションの目的や対象ユーザーなど、基本的な情報を共有している。	アプリケーションの目的とデザインの方向性、それがどうユーザーのニーズと結びついているかについて、明確に説明している。	解決しようとしている課題、市場のニーズ、対象ユーザー、ユーザーのデザインニーズについて、明確かつ有効的に提示している。	どのようにアプリケーションがユーザーのニーズを満たし、生活を変えるのかについて、エビデンスを示しながら説得力を持ったプレゼンテーションをしている。	
プレゼンテーション技術	重要項目を説明している。	自信と熱意を持ってプレゼンテーションを行っている。	聞き手を引きつける様々な方法を使って魅力的なプレゼンテーションを行っている。	独創的で、印象的なストーリーで構成されている。視覚要素をうまく使い、聞き手を引きつけながら発表できている。チームメンバー間の連携もスムーズである。	
ユーザーインターフェイス	UIデザインによりアプリケーションの機能が高められており、すべての画面でテーマが一貫している。	UIデザインには、iOSでよく使われるインターフェイス要素、アイコン、テキストスタイルなどが採用されており、使いやすく機能性に優れている。プロトタイプには、アプリケーションの動作がわかるインタラクティブな要素が含まれている。	美しく、シンプルでストレスなく使えるUIで、色、レイアウト、読みやすさが細部まで考慮されている。ユーザーが困らないよう、ナビゲーションが工夫されている。	デザイン性が高く、もっと使ってみたくなる仕掛けがある。アニメーションや色、レイアウトを工夫し、シームレスで楽しい体験が提供できている。	
ユーザー体験	アプリケーションの目的や、アプリケーションを操作して目的を達成する方法が明確に表現されている。	一貫性があり、標準的なナビゲーションが実装されている。直感的に画面遷移ができる。	ユーザーのニーズに柔軟に対応できる構成になっている。アクセシビリティやプライバシー、セキュリティを考慮した機能が含まれている。	革新的で驚きのある体験が提供されている。ほかのチームや既存のアプリケーションには見られないスタイルや個性が表現されている。	
コーディングの概念	アプリケーションの機能と実装するコードに関連性が見られる。	データ型、条件ロジック、タッチイベントなどの基本的なコーディングの概念と、アプリケーションとの関係について説明している。	必要なコーディング作業の具体的な説明があり、そのコードによってどのような機能が実装されるのかが、わかりやすく提示されている。	アプリケーションのアーキテクチャ、データ構造、アルゴリズム、および機能についての説明があり、なぜそのような選択をしたのかについての過程が説明されている。	
技術的観点(任意)	Swiftコードが特定の条件で動く。Xcodeを使ったプロトタイプを審査する場合および、評価者がSwiftとiOSの開発についての知識を持ち合わせている場合に使用	あらゆるケースでエラーなく実行できる。基本的なSwiftコードであり、ある程度抽象化されている。	Swiftの命名規則が使用されており、整理されて読みやすく、高いレベルで抽象化されている。iOSのガイドラインに準拠している。	コードはコメントを使って十分に説明されている。Swiftの機能が効果的に使われている。MVC(Model View Controller)などの標準的なデザインパターンが採用されている。	
コメント:					0 合計点



Swift Coding Club

Develop in Swift

修了認定証

さん

あなたは

を修了したことを、ここに認定します。

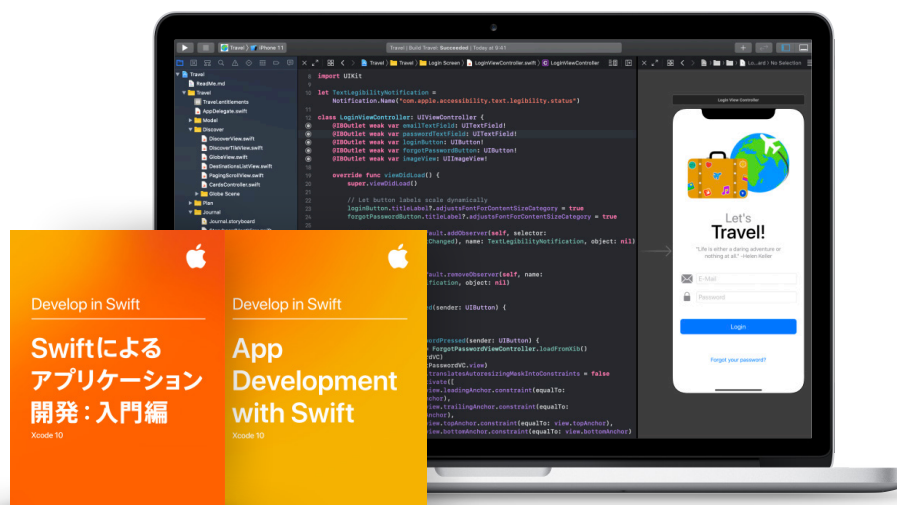
サイン

日付

さらに進める

Swift Coding Clubは、プログラマーとしての第一歩に過ぎません。Develop in Swiftでは、Macでの本格的なアプリケーション開発を楽しく取り組めるようにサポートする教材を提供しています。「App Development with Swift」(英語)を完了した生徒には、「App Development with Swift」の認定資格(英語)も用意されています。

また、クラブ活動だけでなく、コーディングを授業に取り入れるための教師用ガイドも用意されており、高校から高等教育の生徒を対象とした授業案が揃っているので、体系的、段階的にコーディングを授業に取り入れることができます。



[Develop in Swiftの教材についてさらに詳しく >](#)

