大丈夫。

ユーザーのデバイスの場所を取得したので、次のステップはその場所を実際の気象データに実際に変えることです。そのためには、Open Weather Mapなどの実際の気象データを収集するオンラインサービスを使用する必要があります。

ユーザーの現在の位置を取得して、その場所の天気を伝えることができたらいいと思いませんか？これを行うには、APIと呼ばれるものを使用する必要があります。また、Open Weather Mapによると、20を超える天気APIがあります。

では、これらのAPIとは正確には何ですか？ APIは、アプリケーションプログラミングインターフェイスの略です。

また、ウィキペディアを見ると、APIがコマンド、関数、プロトコル、オブジェクトのセットであり、プログラマーがソフトウェアの作成や外部システムとのやり取りに使用できるという、さらに長い定義があります。

最後のポイントは、一般的な操作を実行するための標準コマンドを開発者に提供し、コードを最初から記述する必要がないことです。

これは非常に広い定義であり、このコースで行ったことすべてを振り返ってみると、その定義の多くが実際にこの定義に適合すると想像できます。

そのため、APIの1つのタイプは、Flutterなどのソフトウェアの作成に使用されるAPIです。

Flutterのドキュメントを見ると、次のことがわかります。

「これはFlutter APIリファレンスドキュメントです。」

Flutter SDKまたはFlutterフラグにバンドルされているライブラリを使用することで、ソフトウェアを作成するために使用するコマンド関数、プロトコル、オブジェクトの束を作成したAPIを使用していますか？

その意味でAPIを使用しています。

しかし、他の種類のAPIもあります。外部システムとのやり取りを可能にするAPIがあります。この例は、マッチメイキングアプリであるTinderアプリです。

インターフェースを見ると、共有された友人と共有された興味に関するセクションがあることがわかります。

さて、これらはどこから来たのですか？

このデータは、Tinderが特定のユーザーに関する情報を取得するようにFacebookに要求すると、「この人はアンジェラユ、友達ですか？」

そして、彼らがそれらの応答を取り戻すと、それは自身のユーザーに対してチェックし、それらの友人のすべてを見ることができます、それらの友人のどれがTinderにもありますか？外部システムとやり取りするとき、それは小さな町の市民であるようなものですよね？

この方法でやり取りする特定の建物または特定の構造物があります。

たとえば、ダンキンドーナツに行って、いくつかのドーナツをリクエストし、お金を渡して、その見返りとしてドーナツを返品できます。

しかし、他の種類のAPIもあり、そこに行って何かを要求することはできません。

たとえば、銀行でですか？

ただ歩いて100ポンドが欲しいと言うことはできません。

誰もが銀行にその要求を行うことができるわけではありません。

したがって、銀行と連携する方法は、お金を得るためのリクエストを行う必要があるだけでなく、これがアカウントのロックを解除するためのキーであるというキーを渡す必要があるということです。

そして、それがチェックアウトした場合、銀行は実際に金庫を開きます。

そのため、外部サービスとやり取りするとき、通常は外部サービスから少しのデータを取得しようとしています。また、特定のAPIを使用すると、サーバーの負荷が大きくなりすぎて実際に多くのリソースが消費されないようにする必要があります。外部サーバーを他の人の電気代で実行されているコンピューターと同じように想像すると、そのコンピューターに対して大量の要求を行うと、実際には所有者にかなりの費用がかかることを想像できます。

だから、単にノートブックを持っているだけの銀行を持ち、みんなが入って来て、「私の名前はアンジェラ・ユーだった。 11月25日に100ポンドを払いました。」しかし、実際にはそこに行き、あなたはもっと多くを取ります。

これは、APIがキーを要求し始める場所です。そのキーは、銀行窓口に「助けてくれませんか」と尋ねて、「100ポンド引き出したい」と言うのと似ているからです。そして、「パスワードは何ですか？」

ここに私のパスワードがあります、dここに私のAPIキーがあります。

そして、それが彼らが銀行に持っている口座と一致するなら、彼らは言うでしょう、「さてさあ、どうぞ。

これはあなたが要求した£100ポンドです。」

したがって、この場合、その銀行窓口、あなたと金庫の間のインターフェース、つまりAPIです。そして、要求できること、できること、パスワードを求めて確認することなど、従わなければならない特定のルールを強制します。

それは基本的に、アプリケーションプログラミングインターフェイスです。 APIを使用すると、単にデータを取得するだけではありません。

データベースにデータを入れることもできます。

銀行の状況では、£100を入金することをお勧めします。

また、このAPIがこれを処理し、データを取得、検証、検証し、適切な場所に移動したことを確認してから、独自のデータベースに配置します。

別のWebサーバーのデータベースと直接対話することはできません。

だから、例えば、Open Weather Mapに「あなたは何を知っているの？天気データを作成しました。

だからここに行きます。

それを受け取って、実際に今ロンドンにある、それが35度であるということを世界中に伝えてください。」

本質的に、アプリが他の人のプログラムや他の人のサーバーや他の人のコードとやり取りしているとき、このAPIは、どのようにやり取りするかを指示する中間のインターフェースであり、それがエンゲージメントのルールです

本質的に。リクエストできるものがいくつかありますが、それが有効なリクエストであれば、有効なレスポンスを返すことができます。

今、銀行の窓口に行って彼女に尋ねた場合、「周期表の31番目の要素は何ですか？」

なぜなら、それは宿題の中にあり、それを素早く簡単に行う方法が必要だからです。

彼女は知りません。

そして、エラーを返すか、webspeakで404ステータスコードを返します。これは、探しているリソースが見つからなかったことを意味します。

基本的に無効なリクエストを行いました。

彼らが処理できるのは特定のことだけであり、彼らが応答するのは特定の要求だけです。

したがって、APIでもまったく同じです。

APIの元の定義に戻って、外部システムからデータを取得しようとすると、使用するために事前に定義された一連のコマンドと関数を使用してデータを取得する必要があります。

そして、必要なデータを実際に取得できるのは、システムと対話するこれらの承認された方法を介してのみです。そして、それらが作成したこれらの関数またはメソッドについて学習する方法は、APIのドキュメントを使用することです。

したがって、たとえばこの場合、オープンな天気図APIがあります。また、ここには、取得できる各データの詳細なドキュメントがあります。

したがって、地球上の任意の場所の現在の天気データにアクセスする場合、通常のURLに少し似たものを使用して呼び出すことができますか？

したがって、データの場所はapi.openweathermap.org/dataであり、その後、それらは対話しているAPIのバージョンに対応した何らかのバージョン管理コードを持っています。

次に、やり取りする内容を指定し、クエリパラメータを指定します。

したがって、この場合、「ロンドンの天気のデータが必要です」と言っているので、ブラウザの上部にあるURLボックスに入力するだけでこのリクエストを行うことができ、一連のデータが返されます。ですから、ロンドンでは現時点で霧雨があり、気温は280Kです。

そのため、次のレッスンでは、Open Weather MapサーバーからAPIを操作してデータを取得する方法を検討し、これを実現するためにFlutterのネットワークライブラリを使用します。

それで、それ以上については、次のレッスンで見ます。