よし、新しいアプリを作成するときです。

そして、このモジュールでは、ユーザーが質問に答えなければならないクイズアプリを構築し、どれだけ正解したかを伝えます。

そして、それを構築しながら、クラスとオブジェクト、およびDartを使用してそれらを作成する方法についてすべて学習します。

モジュールの終わりまでに、次のようなアプリが作成されます。

ユーザーのスコアを追跡し、真と偽の質問に答えることができます。アプリはシンプルに見えますが、内部では、さらに中間的なプログラミングの概念に取り組んでいきます。また、アプリをより良く構成する方法を見つけます。

いつものように、まず最初に、開始ファイルとして取得したGitHubリポジトリを使用してセットアップしましょう。

これは、Quizzlerアプリの開始プロジェクトです。

そしていつものように、URLをコピーしてAndroid Studioに移動し、バージョン管理からプロジェクトをチェックアウトし、Gitを選択して、そのURLに貼り付けます。以前に取得し、プロジェクトのクローンを作成しました。そして、いつものようにnoを選択します。

代わりに、Android Studioプロジェクト内にある、保存した場所を開くことにします。

そして、Flutterパッケージをすべて取り込み、Flutter開発用にプロジェクトをセットアップするために、Get依存関係をクリックする必要があります。

いつものように、すべての開始コードを含むmain.dartがあります。

この場合、実際にはpubspecに変更を加えておらず、実際にはそのままにしておきます。また、Android側とiOS側のアプリアイコン以外に、追加したリソースや画像はありません。

ただし、開始ファイルが用意されている主な理由は、実際に進んで非常にシンプルなユーザーインターフェイスを作成したためです。

したがって、プロジェクトのクローンを作成したら、先に進み、実行して、何をしたかを確認してください。アプリが実行されると、これが表示されるはずです。

暗い背景があり、白いテキストがあります。そして、まさにそれが言っているように、これは私たちの質問が行く場所であり、これらはユーザーが答えをクリックして選択できる真と偽のボタンです。

そして、これが私たちがやったことのすべてです。そして、単なるテキストウィジェットと2つのフラットボタンウィジェットです。そしてもちろん、列内の拡張されたウィジェット内に埋め込まれているため、適切なスペースを使用できます。

余分な練習をするためだけに、このユーザーインターフェイスを最初から完全に作成する場合は、新しいプロジェクトを最初から作成してください。

必要な画像はなく、pubspecに変更はありません。しかし、このモジュールで多くのことを学習することを考えると、物事の作成と実行に多くの時間を費やす必要がないように、これらの開始コンポーネントを提供することは良い考えだと思いましたすでに方法を知っています。

したがって、コードを確認してウィジェットの設定方法を確認したら、次のレッスンに進む準備がほぼ整い、スコアキーパーの構築を開始します。

それで、そのすべてについて、次のレッスンでお会いしましょう。