大丈夫。

これで、アプリが質問リストに質問を表示できるようになりました。また、質問と回答を照合して、ユーザーが正しい回答を得たかどうかを確認できます。

しかし、現時点では少し不格好です。

相互に関連付けられるべきものを追跡している2つのリストがあります。これはこの質問に対する答えであり、理想的には一緒に保存されていれば素晴らしいことです。

では、どうすればこれを行うことができますか？

さて、新しいクラスを作成できます。子プロパティ、パディングプロパティ、またはフレックスプロパティなどの異なるプロパティを持つウィジェットがあるように、質問プロパティと回答プロパティを持つクラスを作成して、質問と回答を関連付けることもできます。単一のオブジェクト。これを行うために、すべてのアクションが発生するlibフォルダーに移動し、右クリックして新規に移動します。

新しいDartファイルを作成することを選択します。このDartファイルの質問を呼び出すことにします。これはまさにそれが表すものだからです。

それから、「これをgitに追加しますか？」はいまたはいいえをクリックできます。

このプロジェクトではgitを実際に使用していないため、この場合はそれほど問題ではありません。

したがって、新しいクラスを作成するには、キーワードクラスを使用する必要があります。

その後、クラスに名前を付けます。

したがって、このケースでは、クラスの質問を呼び出します。そして、慣例により、あなたが作成するとき

新しいクラスは、すべてのクラスが大文字で始まるように、クイズページまたはウィジェットはすべて大文字で始まると言います。これらはそれぞれ異なるクラスに属しているためです。

そこで、ここで質問クラスを作成し、この質問クラスの特別な点を定義するために括弧のセットを開きます。質問クラスには2つのプロパティのみがあり、それらはすべての質問に必要なプロパティになります。

したがって、1つは文字列になり、questionTextと呼ばれ、もう1つはブール値になり、questionAnswerになります。

この質問の中で、新しい質問を作成するときにこれらの変数に値を与える必要があります。

そのためには、コンストラクターと呼ばれるものを作成する必要があります。そのため、この質問オブジェクトを作成するときに、質問テキストと質問回答の値を指定します。そして、クラス名と同じ質問を書くことでこれを行います。そして、括弧のセットを開き、中括弧を追加します。これは、関数のように見え始めています。

これは通常、入力が行われる場所です。この場合、これらは新しい質問を作成するときに提供する値です。

したがって、qと呼ばれる文字列を持ち、aと呼ばれるブールを持ちます。そして、あなたは私がここに行くところを見ることができます。

先ほど開いたこれらの括弧内で、この質問のquestionTextをqの値に等しく設定し、次に質問の回答をaの値に等しく設定します。

そして、コードのこの部分には特別な名前があります。

コンストラクターと呼ばれます。

質問オブジェクトのこの構造ができたので、main.dartに新しい質問オブジェクトを作成できます

そこで、ここで新しい質問オブジェクトを作成し、そのタイプを指定してそれを行います。

したがって、これは新しい質問タイプです。そして、現時点では、このクラスの質問を知りません。これは未定義のクラスの質問ですが、まさにここにありますか？

では、このファイルについてこのファイルにどのように伝えるのでしょうか？

インポートする必要があります。

それでは、入力を書きましょう。

ここで、question.dartファイルを作成します。

そのファイルがインポートされたので、エラーはなくなり、クラス内の質問が認識されます。

したがって、問題のデータ型を持つ新しい変数を作成しています。

そしてこれは呼ばれるつもりです、それをq1と呼びましょう。そして、新しい質問に等しくなるように設定し、ここで構築したコンストラクタを使用します。

qを提供し、aを提供します。

したがって、Enterキーを押すと、ここで最初の質問を入力するqプロパティをタップできることがわかります。そして、これは基準を満たす文字列を期待しているので、最初の質問の答えとなるaを入れます。これはブール値であるfalseです。

そこで、ここで作成した質問コンストラクターから新しい質問オブジェクトを作成しました。そして、私の新しい質問q1には、ここに入力したものに設定された質問テキストがあり、aにここに入力したものと同じに設定された質問の回答があります。

だから今私ができることは、私はprint q1と言うことができます。質問テキストまたは印刷q1。質疑応答。そして、これらの2つのデータがすべて、ここで作成したオブジェクトに関連付けられていることに注目してください。また、リスト内でそれぞれがこの順序で一致することを確認する必要はありません。テキストと回答のデータは、q1と呼ばれるこのオブジェクトにグループ化されています。質問と回答のリストを作成する代わりに、ここであなたに挑戦してください。

ここにある質問と回答を使用して、代わりに質問のリストを作成する方法を理解できますか？

そのため、ビデオを一時停止して、このチャレンジを完了することができるかどうかを確認してください。

大丈夫。

それでは、これらすべてのコード行をコメントアウトしましょう。そして、すぐ下に新しいリストを作成します。リストのデータ型は質問オブジェクトです。

質問のリストになります。そして、あなたはそれを好きなように呼び出すことができますが、私はそれを質問バンクと呼びます。そして、その角括弧表記を使用して新しいリストに等しくなるように設定します。リスト内で、いくつかの質問オブジェクトを作成する必要があります。そして、ここでq1を作成したのとまったく同じ方法で作成します。

質問コンストラクターを使用します。最初の質問であるqの値を指定し、次にaの値を指定します。この場合はfalseです。

それでは、次の質問を作成してみましょう。この質問にはこの文字列のキューがあります。

そして、aをtrueに設定する必要があります。それをここに入れましょう。

最後に、最後の質問に進みます。前と同じように、qまたはaを指定するだけです。そして、これも真実です。

大丈夫。

したがって、質問オブジェクトを含むリスト内に3つの質問オブジェクトがすべてあり、questionBankと呼ばれます。

したがって、これはもう必要ありません。

そして今では、質問バンクの質問ごとに、質問と回答が非常に簡潔にリンクされています。そのため、このコードをすべて削除するか、参照したい場合はコメントとして保持することができます。

しかし、私はそれを取り除くつもりです。

だから今、いくつかのエラーがあり、Dart分析がそれを教えてくれるので、エラーがあると言うことができます。

しかし、右側に一目でわかるように、そこに小さな赤いボタンがあると、エラーがあることがわかります。エラーを見つけるために、ここに1つがあり、ここに1つがあります。

下にスクロールすると、最初のエラーがここにあります。テキストウィジェット内のテキスト。以前は、questionsというリストがあり、1つのquestionNumberにあるアイテムを取り出したためです。

そのため、現時点では、それはそのリストの最初の項目です。

しかし、questionNumberはなくなったので、質問バンクからどのように取り出すことができますか？

まあそれはほとんど同じ方法です。 questionBankというリストをタップします。

このインデックスで質問を引き出します。

そのため、アプリを最初に起動したときに、インデックス0の質問バンクになります。

これで、この質問オブジェクトにアクセスできるようになりました。ただし、テキストウィジェットには文字列が必要なため、これは機能しません。

したがって、質問クラスではquestionTextと呼ばれるこの部分を表示する必要があります。したがって、このインデックス.questionTextに質問バンクを書くだけです。そして今、まったく同じことを行うことができますが、代わりに新しい質問バンクを使用します。

下にスクロールして、次のものを修正しましょう。

次の問題はここにあり、以前は回答リストがありましたが、現在はなくなっており、質問バンクに置き換えられています。

したがって、特定の質問番号の質問バンクをタップして正しい答えを取得し、.questionAnswerを作成します。

そして、それはもちろん、この質問オブジェクトを作成するときに、aの値に設定されているこのプロパティから来ています。これに対処したので、もう1つエラーがあります。それはここにあります。

前とまったく同じです。質問番号.questionAnswerのインデックスにある質問バンクは、現在の質問の正しい答えを取得します。したがって、すべてのエラーを修正し、保存して先に進み、アプリをホットリスタートすると、質問番号がゼロにリセットされ、最初の質問から開始する必要があります。

この質問でtrueをクリックすると、最初の質問の答えがfalseであるため、ユーザーが間違っていることがわかります。もちろん、質問がなくなるまでこれを進めることができ、その時点で再びクラッシュします。

しかし今では、質問と回答をまとめてオブジェクトと呼ばれるものにまとめました。そして、クラスを作成してオブジェクトを作成しました。したがって、クラスとオブジェクトの概念がよくわからず、ここで実際に何が起こっているのかわからない場合は、次のレッスンに進んで、クラスとオブジェクトとそれらがどのように機能するかについて深く掘り下げてください。ダーツ。

これに非常に精通しており、コンストラクターがどのように機能し、クラスがどのように作成され、オブジェクトがどのように構築されるかを理解できる場合は、次のレッスンをスキップしてアプリの構築を続けてください。