大丈夫。

開始プロジェクトのクローンを作成したので、画面を見てみましょう。そして、すべての画面を表示できるようにするために、まずmain.dart内にいくつかのルートを設定します。

マテリアルアプリにroutesプロパティを追加しますが、これはチャレンジとして残しておきます。

Flutterアプリで名前ルートを使用して次のことができるようにする方法を、GoogleまたはGoogleで試してみてください。

4つの名前付きルートのマップを作成できますか？ 1つのルートは、ようこそ画面、ログイン画面、登録画面、チャット画面の順に移動する必要があります。次に、初期画面を設定してようこそ画面に移動し、アプリを実行します。ようこそ画面が表示されていることを確認し、その後、ログイン画面に移動してアプリを実行するように初期ルートを設定します。今回は、代わりにログイン画面が表示されるはずです。

そのため、準備ができたらビデオを一時停止して、このチャレンジを完了してみてください。

大丈夫。

あなたが覚えているなら、名前付きルートを構築する方法に関する素晴らしい料理のレシピがあります。

そして、あなたはそれに従うか、これを行う方法を覚えているかもしれません。

したがって、routesプロパティで最初に最初に、期待されるものを確認しましょう。

そのため、キーとして文字列を持ち、オブジェクトとしてルートビルダーを持つマップオブジェクトが必要です。

そのため、現在のコンテキストと、そのルートに移動した結果としてどの画面またはどのステートフルウィジェットを構築するかを指定する必要があります。

したがって、ここで示したように、独自のルートのマップを作成します。

それでは、routesプロパティに戻り、中括弧のセットを開いて新しいマップを作成します。

そして、ルートの最初の項目には、ようこそ画面のキーがあります。そして、コンテキスト、現在のコンテキストを入力として受け取り、希望の画面であるウェルカム画面を返す匿名関数を作成することにより、ルートを構築できます。

それで、これが邪魔にならない最初のルートです。

先に進んで、ログイン画面となる2番目のものを作成しましょう。そして、ここでもまったく同じことを行います。

ただし、ようこそ画面に移動する代わりに、ログイン画面に移動して、他の画面でこれを繰り返しましょう。

登録とチャットがあります。

大丈夫。

これで、すべてのルートがキーと値のペアとして定義され、初期ルートを指定できるようになります。

したがって、アプリの開始時に選択するのは、ようこそ画面になります。

最初のルートを使用する場合は、homeプロパティを使用できないことを覚えておいてください。

彼らはまったく同じことをします。

したがって、両方が存在する場合、それらは互いに競合します。

これで、アプリを実行してその外観を確認する準備が整いました。すべて順調に進み、ログインまたは登録するためのボタンがあるウェルカム画面が表示されるはずです。

また、他の画面をチェックアウトする場合は、ルートの名前を入れ替えるだけです。

そのため、ログイン画面を確認したい場合は、もちろんそれをここに置いて、ホットリスタートを押して、画面を表示またはログインできるようにする必要があります。

それはすべて非常によくて良いことですが、ここに問題があります。他の文字列を参照するために文字列を使用しているだけなので、ここにタイプミスがある場合、IDE Android Studioは実際には役に立ちません。

それで、私が意味することをお見せしましょう。

ログイン画面を入力する代わりに、十分なコーヒーがなく、画面に長い間入力するか、他の種類のタイプミスを入力した場合、ルートマップ内にまったく同じようなルートがないため、実際にはまったく何も言及していませんこの。

ログイン画面と呼ばれるものがありますが、コンピューターはそれが私たちが言及していることを知るのに十分ではありません。

そのため、アプリを実行しようとするとクラッシュし、本当にあいまいに見えるもの、グローバルキーなどについて何かが発生します。

基本的に、最初の画面を表示することはできません。

これがプログラマーが用心深い理由であり、Android StudioのIDEが私たちを助ける方法がないため、重要なときに文字列の使用を避けようとします。

したがって、ここのような文字列を使用する必要がある場合、文字列として存在する場所が1つだけであることを確認しようとします。

また、他のすべての場所では、コードはチェック可能であるのに対して文字列はチェックできないため、何らかのコードを使用してエラーを最小限に抑えます。文字列は何でもかまいません。

これを変更しましょう。これらのルートの名前を、それが指す実際の画面に関連付けないのはなぜですか？

たとえば、ようこそ画面に移動し、ここの上部にあるようこそ画面のステートフルウィジェットに移動できます。このウィジェットは、ここをナビゲートするときに作成したいものです。新しいプロパティを作成し、文字列が正しい場合はどうでしょうか。

そして、それを単にidと呼びましょうか？この特定の画面のID。

そして、文字列を指定して、welcome\_screenと呼ぶことができます。

ここで、ここまたはここでテキストを提供する代わりに、welcomeScreen.idを参照するだけで、ここでも同様に行うことができますか？

したがって、ログイン画面に移動する代わりに、ようこそ画面に再度移動する場合は、そのIDを指定できます。そのため、新しいウェルカム画面が作成され、そのIDをタップして、どのルートに一致するキーがあるかを確認します。

それでは、これを実行しましょう。そして、私たちのルートが機能していることがわかります。

最初のウェルカム画面が画面に表示されます。しかし、これについて考えると、実際にはコードの点でかなり無駄です。

こことここの両方でそのIDを取得できるようにするために、まったく新しいウェルカム画面を作成しています。

そして、それは私たちのリソースではあまり効率的ではありません。

それでは、他に何ができますか？

この値を持つidという名前の文字列を作成する代わりに、静的な単語をその前に追加するとどうなりますか？

したがって、これは修飾子であり、Java、Dart、Swiftなどのさまざまな言語で使用されています。そして、これを使用してこの変数を変更し、クラスに関連付けられるようにすることができます。

したがって、私ができることは、ウェルカムスクリーンオブジェクトを作成する代わりに、そのIDがクラスに存在することです。そのため、括弧を削除することができます。つまり、新しいオブジェクトを作成しなくなったため、より効率的になります。

ようこそ画面が表示されているかどうかを非常に難しいと言っても、Android Studioがそのことを教えてくれるので、ここにエラーがあることを教えてくれます、これが何であるかわかりません。

あなたはおそらく十分なコーヒーを飲んでいないか、ご存知のとおり、朝食を食べていません。

これをレビューしてください。それで、私はそれを見て、そうだね、それは歓迎すべきスクリーンであり、今では常に世界でうまくいくはずだ。しかし、私が使用しているこの静的キーワードとは正確には何ですか？また、このような変数に追加したときの意味は何ですか？次のレッスンでは、静的なメソッドと変数について詳しく説明します。

したがって、すでに静的に非常に精通しているプログラミング言語を使用している場合は、次のレッスンをスキップしてください。

しかし、これがDartで何であり、どのように機能するかを知りたい場合は、次のレッスンで詳しく説明します。