大丈夫。

したがって、このレッスンでは、Dartリストについて詳しく説明します。そして、私たちは作成できることをすでに見てきました

買い物リストを作成するのと同じように、リストにいくつかのアイテムを追加し、それらを順番に入れます。

したがって、たとえばこの場合、今後のパーティーのゲストリストを作成していた場合、リストに名前を追加すると、順番にリストに追加されます。

また、コンピューターを扱うときは、常にゼロからカウントを開始します。

つまり、最初のアイテムは実際にはリストのゼロ番目のアイテムになり、その後、順序は論理的に続きます。

したがって、リストからアイテムを選択するときは、インデックスでそれらを参照します。

したがって、この場合、アンジェラはインデックス0にあり、ケイティはインデックス2にあります。そして、リストが0から始まるという事実のために、少し直感に反します。

したがって、リストから3番目を選択した場合、実際には1 2 3ではなく、実際には1つです。これがDartPadを使用してどのように機能するかを確認できます。

DartPadでは、実際に作成したコードを共有できます。

したがって、この演習を少しでも簡単にするために、以前に作成した少しのコードを共有したいので、すべての名前をコピーする必要はありません。そして、DartPadを共有する方法は、コードを記述して、GitHubの要点を作成することです。 GitHubリポジトリとは異なり、GitHubの要点は1つのコードファイルにすぎず、コードのスニペットを非常に簡単に共有でき、作成と使用は完全に無料です。

そのため、これをアカウントの下に作成しました。そして今、この要点を見ると、このアドレスの下にあることがわかります。

したがって、ユーザー名とスラッシュの後にあるすべてをコピーしたら、dartpad.dartlang.org Webサイトの最後に貼り付けることができます。

そこで、スラッシュを追加して、ここに貼り付けます。

Enterキーを押すと、DartPad内のDartファイルとして読み込まれます。

そのため、このURLを表示します。このURLにアクセスすると、まさにこのURLが表示されます。

したがって、同じファイルで一緒に作業できます。

ここに来ると、myListという文字列のリストがすでに作成されていることに気付くでしょう。

そして、リスト内で、いくつかの開始データを使用して作成しました。

そのため、ここでスライドで前に見たものと正確に一致する4つの名前があります。

リストから3番目の項目を選択する場合、myListと呼ばれるこの変数を参照しているリストを言うだけで、角括弧のセットを追加し、内部に整数を指定できます。番号。これは、リストから選択するアイテムのインデックスを参照できます。

したがって、ここに数字3を入力すると、これは3番目の位置にあるリスト内の項目と等しくなります。

ゼロ、1、2、3。

したがって、これはジャックに等しくなります。

そして、コンソールに印刷するだけでこれを確認できます。

だから、私はそれの周りに印刷ステートメントをラップし、実行をヒットするつもりです。

ここで、ジャックが印刷されることを確認できます。

そのため、リストから3番目のアイテムを取り出しました。

また、データの一部を提供することにより、インデックスの取得を試みることもできます。

たとえば、リストでケイティを探していた場合、インデックス番号2にあることがわかります。

そして、単にmyList.indexOfと言うだけでこれを行うことができます。探している文字列を提供する必要があります。この場合はKatieです。

そして、大文字の使用に関してはまったく同じ方法でつづらなければなりません。そして、代わりにこれを印刷するとき、それをコメントアウトして、作成したばかりのこれを印刷ステートメントでラップしましょう。

実行をクリックすると、コンソールに2が表示されます。これは2に等しいためです。前に見たように、単に.addを使用してかっこ内でリストに追加することもできます。追加したいものは何でも提供します。

また、リストに追加する場合、新しいアイテムは常に、リストの先頭またはランダムではなく、リストの最後に配置されることに注意してください。したがって、この場合、先に進んでこれをコメントアウトし、myList.addを作成し、追加するのはゲストリストに別の人がいる場合です。

そして今、myListを印刷すると、今では大丈夫なように見えます。

コードを実行してみましょう。リストがアンジェラ、ジェームズ、ケイティ、ジャック、ベンで構成されていることがわかります。また、コンソールは、角括弧内にデータを表示することで、これがリストであることを非常に役立ちます。

したがって、addを使用すると、常にデータが最後に追加されることがわかります。

正しい？

ただし、中間にデータを追加したい場合は、これを行う方法もあります。実際にデータの一部をリストに挿入し、その後に来たすべてを次のインデックスにプッシュできます。

したがって、この場合、Benがここに入り、位置2に挿入され、位置2にあった前のアイテムを位置3に置き換えました。そして、これを行う方法は、myList.addの代わりに記述することです。代わりにmyList.insertと言います。そして、インデックスを提供する必要があります。リストのインデックスのどこに挿入するのでしょうか。また、挿入したいものは何ですか？

したがって、Benは位置2に昇格します。もちろん、挿入する文字列はBenという名前です。

実行すると、以前にBenが最後に追加されたことがわかります。Benは2の位置に表示されます。リストは0、1、2から始まることに注意してください。

リストには他にもたくさんのことができるので、入力中であってもそれらのいくつかを発見できます。

したがって、myListを作成するとき。 DartPadがリストを使用してできることをすべて提案していることがわかります。

たとえば、myList.firstと記述するだけで、リストの最初のアイテムを取得できます。

また、.lastと記述することで、リストの最後の項目を取得できます。

また、リストの並べ替えやリスト内のアイテムの削除など、他にもできることがたくさんあります。

Dartリストでできることのすべてについて詳しく知りたい場合は、このURLへのリンクを追加します。このURLは、Dart言語ガイドのページで、リストとそのすべてについて説明しています。 Dartリストをより詳細に使用できます。

そのため、このWebサイトを参照し、一覧でできることを一目で確認することをお勧めします。

しかし、実際のユースケースに出くわした後、リストの特定の機能を使用する必要がある場合は、単純にそれを検索すると、次のようなドキュメントに遭遇するため、それらを記憶する必要はありません。これまたはStack Overflowの投稿で、必要に応じて使用できます。リストにすべてあります。次のレッスンでは、アプリのコーディングに戻り、いくつかの質問の表示を開始します。