最後のレッスンでは、ジェスチャ検出器について学び、上部の再利用可能なカードに親として追加しました。男性と女性のアイコンが付いたもの。

そして、これらの性別の1つを選択し、色を変更してこれらのカードをアクティブ化できるupdateColorというメソッドを作成しました。

現在、コードは正常に機能しています。

しかし、それは少し冗長であり、それほど素晴らしいものではないものもいくつかあります。

そして最初に頭に浮かぶのは、性別を表す整数を渡すという事実です。

したがって、1つは男性に等しく、2つは女性に等しくなると言っています。

ですから、性別が1であるかどうかを確認するのは、男性カードが押されたということです。

しかし、性別が2の場合、それは女性カードが押されたことを意味します。

想像できるように、6か月後にこのコードに戻った場合、または他の誰かがそれを見ていた場合、あまり意味をなさないため、少し混乱する可能性があります。

ここにはたくさんのコメントがありますが、それは素晴らしいことではありません。

では、どうすればこれを解決できますか？このレッスンでは、列挙である非常に便利なプログラミングの概念を学習することにより、これらすべてに対処します。列挙は列挙を表します。

また、辞書で調べると、何かの数を確定するアクションを意味します。

そして、それは私たちがここで行っていることの一種です？

男性の性別が1番、女性の性別が2番であることを確認しています。

しかし、これは非常に痛みを伴い、非常に混乱します。

しかし、プログラミングでは、これが頻繁に発生する必要があります。

アプリでさまざまな種類の車をエンコードしようとしていて、carTypeと呼ばれるものがあり、1に等しい場合は変換可能であり、2に等しい場合はSUVであり、3であればハッチバックです。

現在、これらの数値は実際には本質的に意味を持ちません。

そして、実際に意味を割り当てて、それを後から思い出そうとしています。そして、私たちと一緒に働いている他の人々にもこの慣習に従うように強制しようとしていますが、それは難しいことです。

しかし、代わりに、すでに意味のある単語を実際に与えることができる場合を想像してください。

したがって、最初のモデルのcarTypeが.convertibleで、2番目のモデルが.SUVであるか、3番目のモデルが.hatchbackであると言った場合。

これは非常に理にかなっており、作業がはるかに簡単であり、これらの異なる種類の車をコーディングするときに話していることをコードで一目で見ることができます。

オンまたはオフまたは真または偽のいずれかであることがわかっているブール値が既にあり、電球のようなものであると想像する場合。非常にバイナリですよね？

オンとオフの2つの状態があります。

しかし、より多くの状態が必要な場合、電球を保持できます。

それから、Enumの使用を開始します。そして、あなたの名前は、あなたが複数のオプションを持っているブレンダーのようなものだと思いますか？

設定が高い場合、設定が低い場合、中程度またはジュースまたはブレンドまたはブレンドを変更する場合があります。そして、それは単なるオンまたはオフではありません。

多くの異なるオプションがあり、数字に依存するだけでなく、エンコードできるようにしたい多くの異なるオプションがある場合、それはまさに列挙型を使用するものです。

そして、実際には各オプションに意味のある名前を付けることができます。 Dartでは、列挙型は次のようになります。

列挙型を作成していることを示すために、最初にキーワード列挙型から始めます。そして、enumに名前を付けます。

したがって、carTypeでも性別でもかまいません。ただし、enum名の命名規則はクラス名と同じです。

キャメルケースがありますが、大文字で始まります。そして、異なるタイプのそれぞれを記述する中括弧のセットがあります。

タイプA、タイプB、タイプC、またはハッチバック、クーペ、コンバーチブルと言ってください。

そして、実際に列挙型を使用する必要がある場合は、単純にEnumNameを書き込むことができます。 typeAまたは。

タイプB。これをもう少し詳しく見るために、Dartpadに向かいましょう。

ここで、車のクラスがあり、以前は座席の数やドアの数などだけしか持っていなかったとしましょう。これは、整数を使用するのが理にかなっていますか？ 5つのドアは文字通り整数です。

ただし、carStyleをエンコードする場合は、整数を使用する必要があります。また、これがハッチバックであることを知っていることを覚えておく必要があります。

それが2つなら、それはSUVなどであることを意味し、それは非常に痛いです。ただし、車を初期化することを決定した場合、コンストラクタを作成し、車を作成するときにthis.carStyleを設定するとします。

その後、メイン機能に移動して、新しい車を作成できます。

それでは、データ型の車にしましょう。

それをmyCarと呼びましょう。これは、構築された車と等しくなります。

そして、carStyleを指定する必要があり、SUVと同等であるとしましょう。

私の車はSUVです。

さて、このコード行を読んだ場合、それを言うコメントをたくさん追加しない限り、人間にはまったく意味がありません。この車スタイル2は、私が新しいSUVを作成していることを意味します。

ここで、プログラムが何をするのかについてのあなたの思考の論理を説明するコメントを書くことは良い習慣です。

しかし、特定の行で実際に何が起こっているのかを理解できるようにするためにコメントが必要な場合、それは素晴らしいことではありません。

そして、これは列挙型が入って日を節約できる古典的なケースです。これらのコメントをすべて失い、2か月後にこのコードを見つけた場合、これはまったく意味をなさないからです。だから何

列挙型を作成した場合

ここで、他のクラスの外でこれを実行してみましょう。enumキーワードを使用して、新しいenumを作成していると言います。そして、enumには、大文字のラクダケースになる名前が付けられます。

そして、これをCarTypeと呼びます。

そして、中括弧のセットを開き、プログラムに必要な可能性のあるさまざまな車の種類をすべて追加します。

それで、ハッチバックを持っているかもしれません、我々はSUVと多分コンバーチブルを持っているかもしれません。そして、楽しみのためだけにクーペを追加しましょう。

したがって、4種類の車があり、それらは列挙型に囲まれているので、これらの車の種類を使用できます。

そのため、整数で表される車のスタイルを使用する代わりに、データ型を列挙型に変更して、carTypeにすることができます。そして今、carStyleを初期化するときに、クーペに1、ハッチバックに2を書き込む代わりに、実際に列挙型を名前付き値として提供できます。

したがって、carTypeを記述できます。ドロップダウンリストですべての値を取得していることがわかります。

だから、新しいコンバーチブルを作成しているとしましょう。

次に、carコンストラクターから作成されたmyCarがあり、そのcarStyleのプロパティは変換可能なcarTypeです。

今度はそれがどれほど良く見えるか見てみましょう。

たとえ100万年後にこのコードに戻ったとしても、コンピューターに既に奴隷にされているかもしれません。

いずれにせよ、私が長い時間を経てこれに戻ってきたとしても、このコード行は依然として非常に理にかなっています。

それでは、私たちのコードに戻りましょう。ここがあなたの挑戦です。ファイルの上部に性別列挙を作成します。

クラス内で列挙型を作成することはできません。そのため、他のクラスの外、おそらく定数の下のどこかにあることを確認してください。そして、更新カラー内およびジェスチャー検出器内でその性別列挙を使用して、この場合は押されたのは男性のカードであり、この場合は処理できるように押されたのは女性のカードだったと言うことができますこのupdateColorメソッド内では異なります。

そのため、ビデオを一時停止し、Dart列挙型を使用してコードをアップグレードしてください。

大丈夫。

まず最初に、enumキーワードを使用してenumを作成します。そして、単にGenderと呼びますが、変数または定数と同じではないことを示すために、最初に大文字で表記するようにします。

今、性別は2つの異なる値のみを持ち、男性と女性になります。

したがって、コードでこの列挙型gender.maleまたはgender.femaleを使用できます。たとえば、updateColorはint型の入力を取得する代わりに、性別の種類の入力として使用できるため、その名前を性別のままにするか、少しわかりにくい場合は、選択した性別に変更できます入力している名前であることを知るために。そして、選択した性別が1ではなくgender.maleに等しい場合、これをすべて実行する必要がありますが、選択した性別が代わりにgender.femaleに等しい場合、これをすべて実行することを確認できます。

そして、ここで色を更新するコードで、1つを渡す代わりに、gender.maleを渡し、ここでgender.femaleを渡すことができます。

それでは、コードがどれだけ良く読み取れるか見てみましょう。このカードでタップが検出されると、色を更新して更新することで状態を更新し、女性カードが押されたことを伝えます。そして、updateColorメソッドに進み、性別が選択された男性であるか、選択された性別が選択された女性であるかを確認し、このコードの残りすべてを実行します。

そのため、1つのプロパティに複数のオプションがある場合、enumは非常に便利です。

そのため、車のタイプや高低中設定、または私たちの場合、性別は男性と女性などです。コードをクリーンアップして表現力を高めることは本当に便利です。しかし、私たちの問題は、コードが非常に冗長であるため、実際に見るのが苦しいということです。

そのため、次のレッスンでは、3項演算子であるDartでアクセスできる別のことを学習します。そして、これらすべてをクリーンアップして、1行または2行のコードに減らします。

そのため、すべての魔法の準備ができたら、次のレッスンに進んでください。