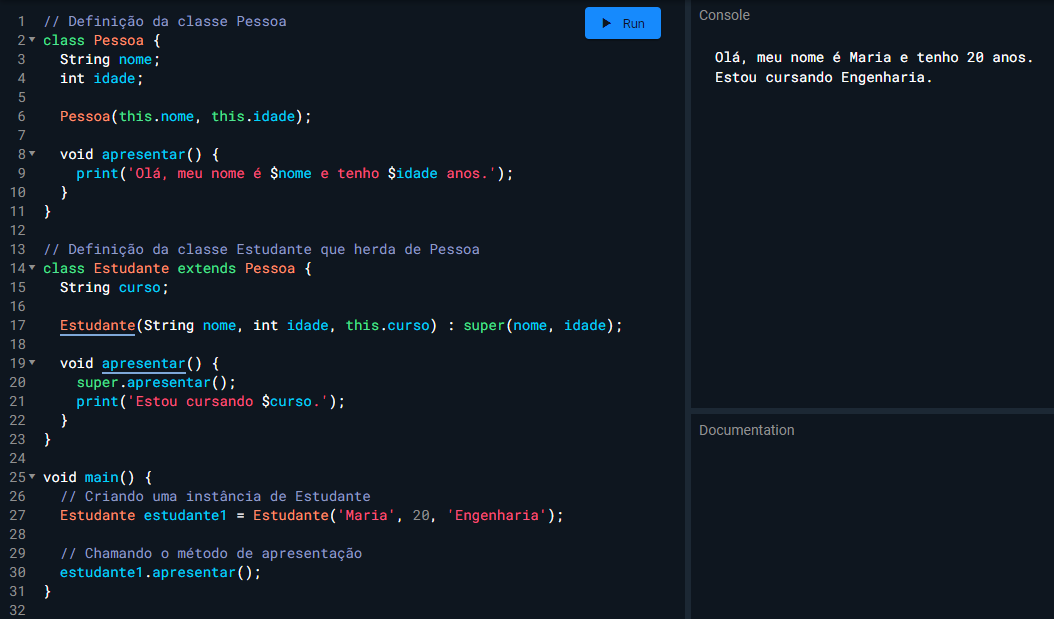
Aula 10

Construindo um projeto

Na última aula do nosso curso introdutório de Dart, vamos aplicar todos os conceitos aprendidos no desenvolvimento de um pequeno projeto. Além disso, faremos uma revisão geral dos tópicos abordados para consolidar o conhecimento.

Vamos criar um programa simples que simula o cadastro de estudantes. Cada estudante terá um nome, idade e curso. Utilizaremos os conceitos de orientação a objetos aprendidos nas aulas anteriores.

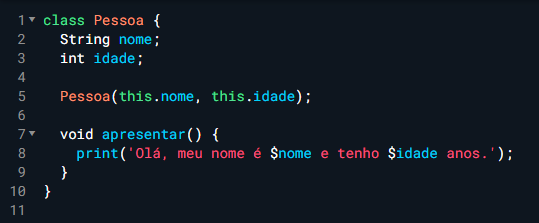


Este exemplo mostra a criação de duas classes, `Pessoa` e `Estudante`, e a utilização de herança para reaproveitar características da classe base. O método `apresentar` na classe `Estudante` chama primeiro o método da classe base (`super.apresentar()`) antes de adicionar informações específicas do estudante.

Explicando o código

O código fornecido define duas classes em Dart: `Pessoa` e `Estudante`. Vamos analisar cada parte do código em detalhes:

1. Definição da classe Pessoa:

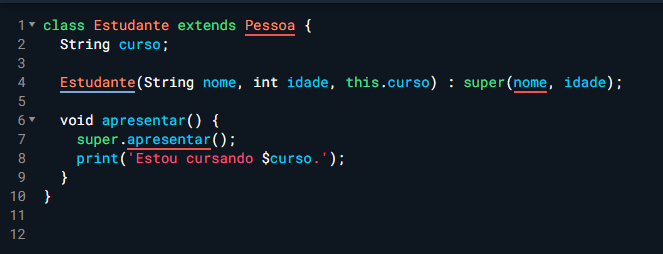


- A classe `Pessoa` possui dois atributos: `nome` do tipo `String` e `idade` do tipo `int`.

- O construtor `Pessoa` é declarado utilizando a sintaxe simplificada, onde os parâmetros `nome` e `idade` são inicializados automaticamente com os valores passados durante a criação de um objeto `Pessoa`.

- O método `apresentar()` é definido dentro da classe `Pessoa` para exibir uma mensagem formatada com o nome e a idade da pessoa.

2. Definição da classe Estudante que herda de Pessoa:

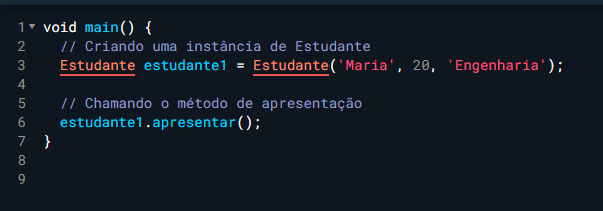
- A classe `Estudante` é uma subclasse da classe `Pessoa`, indicada pela palavra-chave `extends`. Isso significa que um estudante herda todos os atributos e métodos da classe `Pessoa`.

- Além dos atributos `nome` e `idade`, a classe `Estudante` possui um atributo adicional chamado `curso`.

- O construtor `Estudante` é definido de forma similar ao da classe `Pessoa`, com a adição do parâmetro `curso`. A palavra-chave `super` é utilizada para chamar o construtor da classe pai (`Pessoa`) e inicializar os atributos herdados.

- O método `apresentar()` na classe `Estudante` sobrescreve o método homônimo da classe `Pessoa`, utilizando a palavra-chave `super` para chamar o método da classe pai e em seguida exibindo uma mensagem adicional informando o curso do estudante.

3. Função `main()`:



- A função `main()` é o ponto de entrada do programa.

- Dentro da função `main()`, é criada uma instância da classe `Estudante` chamada `estudante1`, passando os argumentos 'Maria', 20 e 'Engenharia' para o construtor da classe `Estudante`.

- Em seguida, o método `apresentar()` é chamado para exibir as informações da estudante `estudante1`, incluindo seu nome, idade e curso.

Em resumo, o código demonstra o conceito de herança em Dart, onde a classe `Estudante` herda os atributos e métodos da classe `Pessoa`, e também mostra como é possível sobrescrever métodos para personalizar o comportamento em subclasse.

Atividade: Projeto

Crie duas classes em Dart: uma classe Veiculo e uma classe Carro.

A classe Veiculo deve ter uma variável privada para a marca e outra para o preço, um construtor para inicializar essas variáveis e um método para exibir informações sobre o veículo. A classe Carro deve herdar da classe Veiculo, adicionar uma variável privada para o ano do carro, um construtor para inicializar todas as variáveis e um método para exibir informações sobre o carro.

Depois, crie instâncias de ambas as classes e demonstre como acessar os atributos usando getters e como chamar métodos para exibir informações sobre os veículos."

Tente gerar este código e somente após isso veja como ficou o projeto na visão do professor.

Resposta:

