droidcon academy

Mastering Code Generation in Dart to Write Repetitive Code Automatically

Aachman Garg

@imaachman

in/imaachman

Overview

- What is code generation in Dart?
- Advantages of code generation
- Annotate classes for code generation
- Inspect classes using a visitor
- Write a custom serialisation generator
- Learn to configure the builder
- Use the generator in a Flutter project
- Code challenge

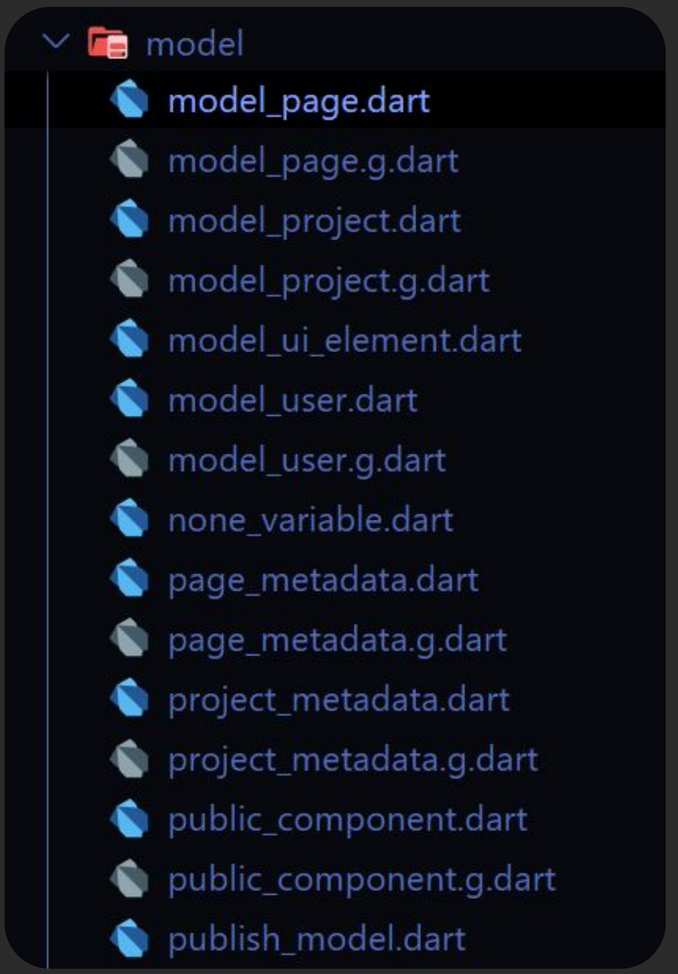
droidcon academy

What is code generation in Dart?

droidcon academy

Repetitive Code Example

```
Person copyWith({
  String? name,
  int? age,
  bool? codes,
}) =>
    Person(
      name: name ?? this.name,
      age: age ?? this.age,
      codes: codes ?? this.codes,
    );
```



droidcon academy

Use Code Generation

Step 1: Annotate the class

```dart
@clone
class Person {
  final String name;
  final int age;
  final bool codes;
}
```

Step 2: Run the ***build*** command

```
dart run build_runner build
```

Advantages of Code Generation

droidcon academy

Advantages of Code Generation

1. Saves time and effort
2. Prevents human errors
3. Clean, consistent and readable code

droidcon academy

Project Demonstration

droidcon academy

Annotate Class for Code Generation

Write Custom Generators

droidcon academy

Dependencies

- **build**: Exposes classes, such as ***`Builder`***, that enable templating the code and generate new files from existing ones.

- **source_gen**: Wrapper around the ***build*** package that provides high level API to write custom code generators.

- **analyzer**: Enables class inspection, i.e., extracting metadata from the classes. No need for explicit import as the ***build*** package uses ***analyzer*** under the hood and exposes it.

- **build_runner**: Enables the ***build*** command. It traverses through the entire codebase in search of annotations and generates the code accordingly.

Class Inspection

Configure the Builder

droidcon academy

Example Usage

droidcon academy

Code Challenge

droidcon academy

Code Challenge Solution

droidcon academy

Summary

- The concept of code generation
- Advantages of code generation
- Annotate classes for code generation
- Class inspection
- How to write custom generators
- Builder configuration
- Usage in a Flutter project

droidcon academy

droidcon academy

Thank You!

Aachman Garg

@imaachman

in/imaachman