



Your ultimate guide to code generation productivity for Dart and Flutter

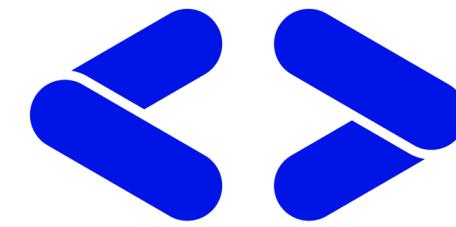
Anna Leushchenko

Fluttercon europe

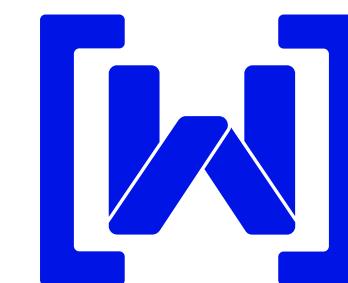
Agenda

- Key `build_runner` principles
- Useful code-generating packages
- Efficient codebase maintenance

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER



Experts



Women
Techmakers



Flutteristas



Business banking
& beyond



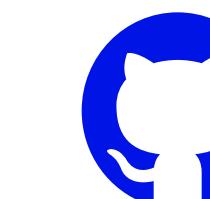
Anna
Leushchenko



AnnaLeushchenko



AnnaLeushchenko



foxanna



foxanna



foxanna

Fluttercon europe

Key build_runner principles

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

build_runner

pubspec.yaml

```
dev_dependencies:  
  build_runner: x.y.z
```

- › `dart run build_runner build -d`
- › `dart run build_runner watch -d`

Useful code-generating packages

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

JSON deserialization with json_serializable

```
@JsonSerializable()  
class Article {  
  const Article({required this.id});  
  
  final String id;  
  
  factory Article.fromJson(Map<String, dynamic> json) =>  
    _$ArticleFromJson(json);  
  
  Map<String, dynamic> toJson() =>  
    _$ArticleToJson(this);  
}
```

Dart classes enhancement with freezed

```
@freezed
class Article with _$Article {
  const factory Article({required String id}) = _Article;

  factory Article.fromJson(Map<String, dynamic> json) =>
    _$ArticleFromJson(json);
}
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

RESTful API consumption with retrofit

```
@RestApi()  
abstract class SpaceFlightNewsApi {  
    factory SpaceFlightNewsApi(Dio dio) = _SpaceFlightNewsApi;  
  
    @GET('/articles')  
    Future<List<Article>> getArticles();  
}
```

Dependency injection with injectable

```
@lazySingleton
class SpaceFlightNewsRepository {
  const SpaceFlightNewsRepository(this._api);

  final SpaceFlightNewsApi _api;
}

@Injectable
class GetSpaceFlightNewsUseCase {
  const GetSpaceFlightNewsUseCase(this._repository);

  final SpaceFlightNewsRepository _repository;
}
```

Enhanced navigation with auto_route

```
@RoutePage<String>()
class ArticleDetailsPage extends StatelessWidget {
  const ArticleDetailsPage({
    super.key,
    required this.id,
  });

  final String id;

  @override
  Widget build(BuildContext context) => Scaffold(...);
}
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Flutter assets management with flutter_gen

```
flutter:  
  uses-material-design: true  
assets:  
  - assets/images/
```

```
flutter_gen:  
  output: lib/presentation/assets  
assets:  
  outputs:  
    package_parameter_enabled: true
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Translations management with i18n

```
title: 'Space Flight News'  
retryButtonText: 'Retry'  
on(String date): 'on $date'  
launches(int n): '$n ${_plural(n, one:'launch', many:'launches')}'  
error: 'Something went wrong'
```

Documentation and tests with bdd_widget_test

Feature: Counter

Scenario: Initial counter value is 0

Given the Counter app is running

Then I see {'0'} text

Scenario: Tapping the Plus icon once increases the counter to 1

Given the Counter app is running

When I tap {Icons.add} icon

Then I see {'1'} text

Barrel files management with barrel_files

```
@includeInBarrelFile
```

```
class ExampleClass {}
```

```
@includeInBarrelFile
```

```
const exampleGlobalConst = 0;
```

```
@includeInBarrelFile
```

```
enum ExampleEnum {one, two, three}
```

```
@includeInBarrelFile
```

```
String exampleFunction() => 'example';
```

```
@includeInBarrelFile
```

```
typedef ExampleTypeDef = void Function(int i);
```

Efficient codebase maintenance

Flutterconeurope

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Optimize Code Generator Input

Optimize Code Generator Input

build.yaml

```
targets:  
  $default:  
    builders:  
      json_serializable:  
        generate_for:  
          # a single `example.dart` file  
          - lib/src/example.dart  
          # all `.dart` files under the `foo` directory  
          - lib/src/foo/*.dart  
          # all `.dart` files under the `bar` directory  
          # and its subdirectories  
          - lib/src/bar**/*.dart
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Keep Packages Small

Keep Packages Small

› `dart pub global activate melos`

`melos.yaml`

```
name: <project_name>
```

```
scripts:
```

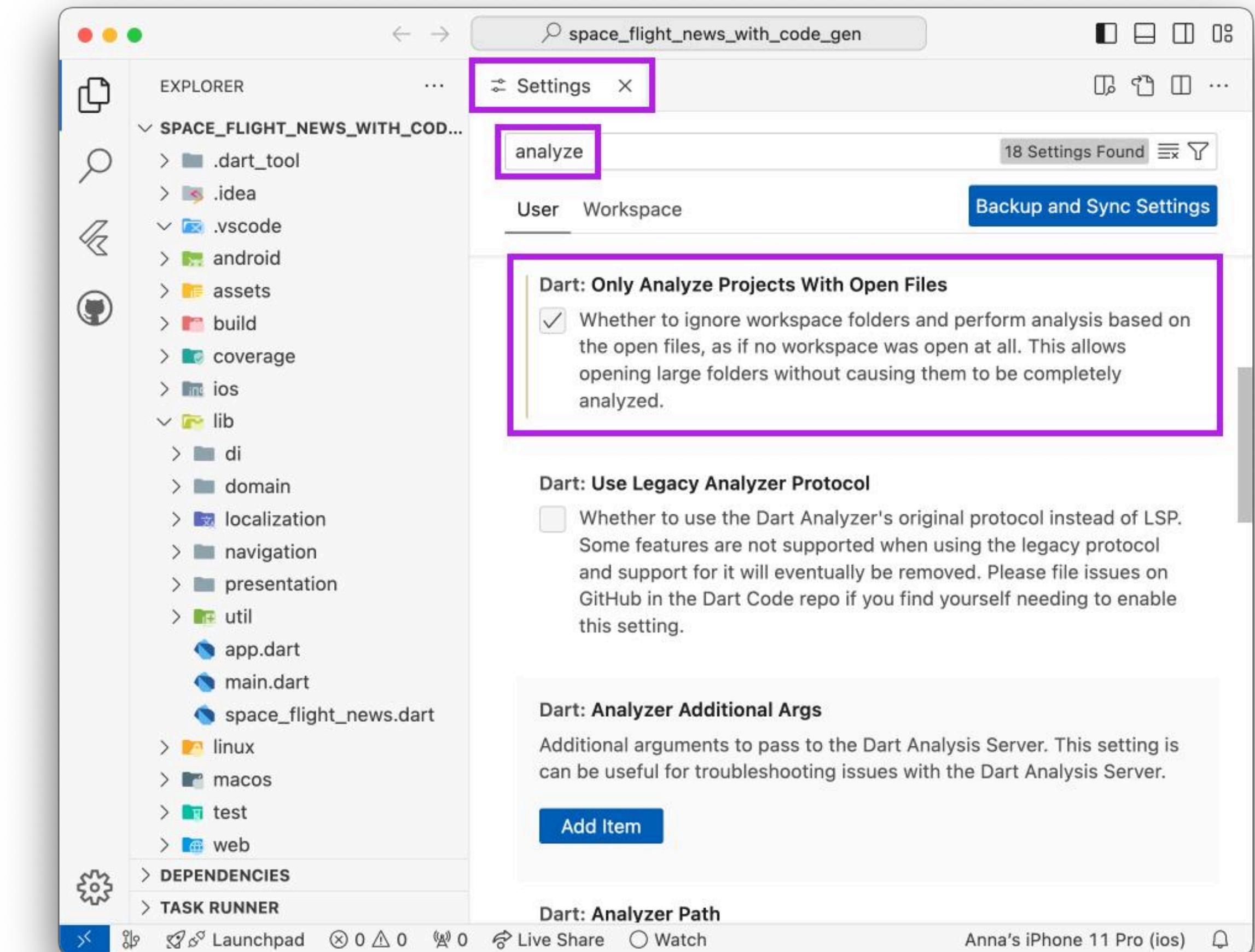
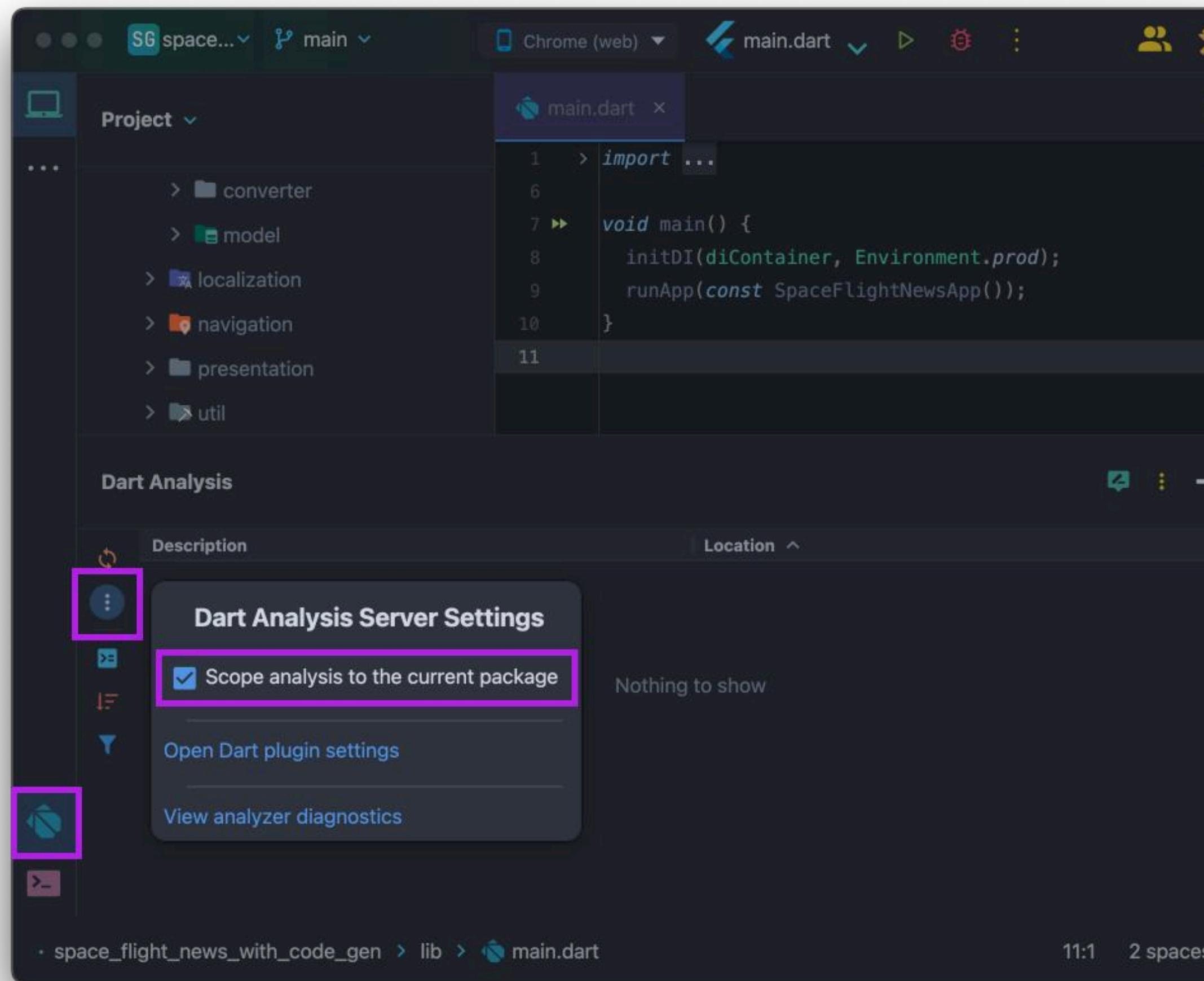
```
  generate:
```

```
    run: melos exec -c 1 --depends-on="build_runner" -- \
      "dart run build_runner build -d"
```

› `melos run generate`

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Keep Packages Small

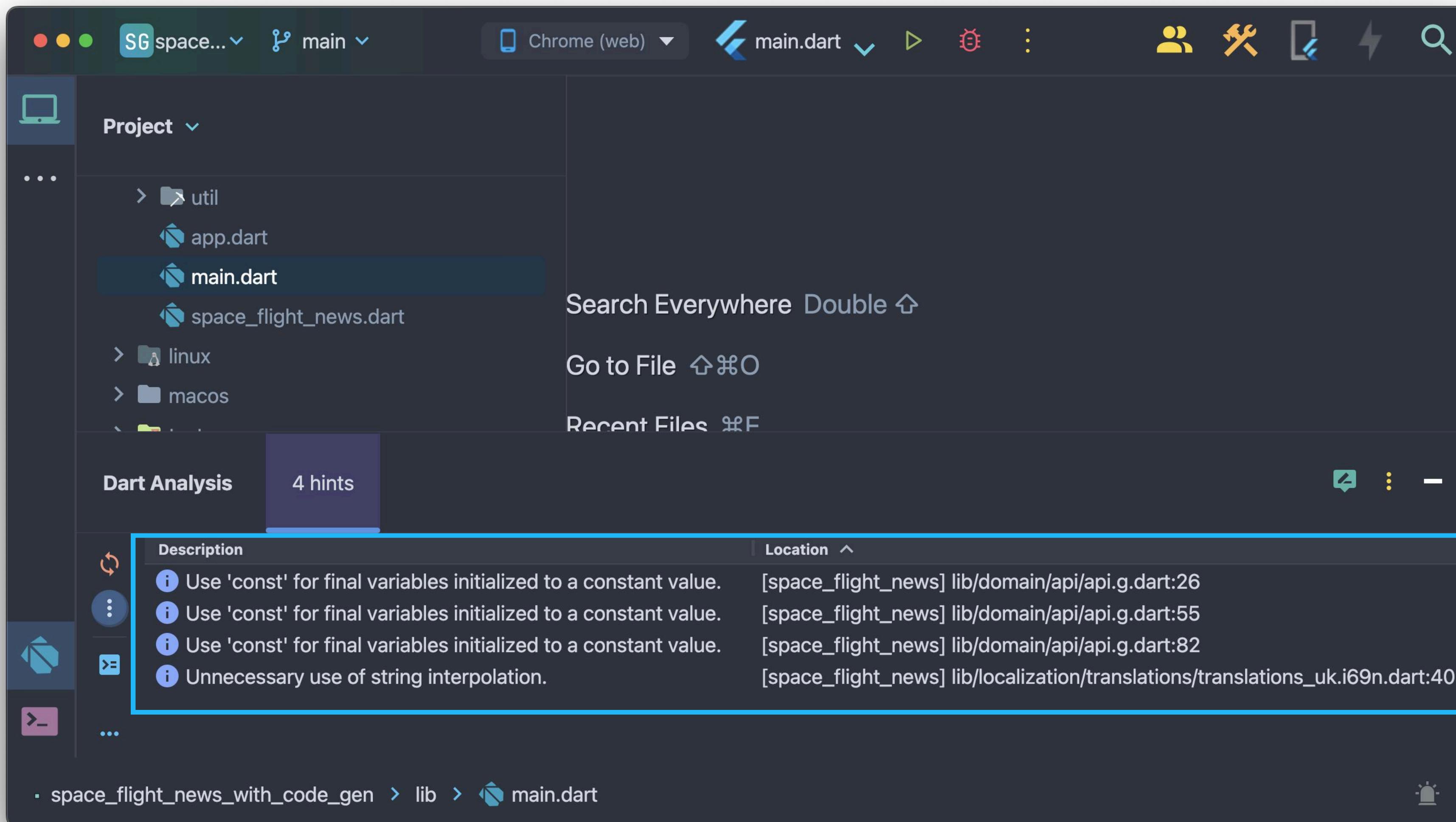


YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Configure Static Analysis

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Configure Static Analysis



Flutterconeurope

Configure Static Analysis

`analysis_options.yaml`

```
analyzer:  
  exclude:  
    - '**/*.config.dart'  
    - '**/*.freezed.dart'  
    - '**/*.g.dart'  
    - '**/*.gen.dart'  
    - '**/*.gr.dart'  
    - '**/*.i69n.dart'  
    - '**/feature**/*_test.dart'
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Add Generated Files to Git

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Add Generated Files to Git

What if I don't?

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Add Generated Files to Git

What if I do?

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Collapse Generated Files for Review

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

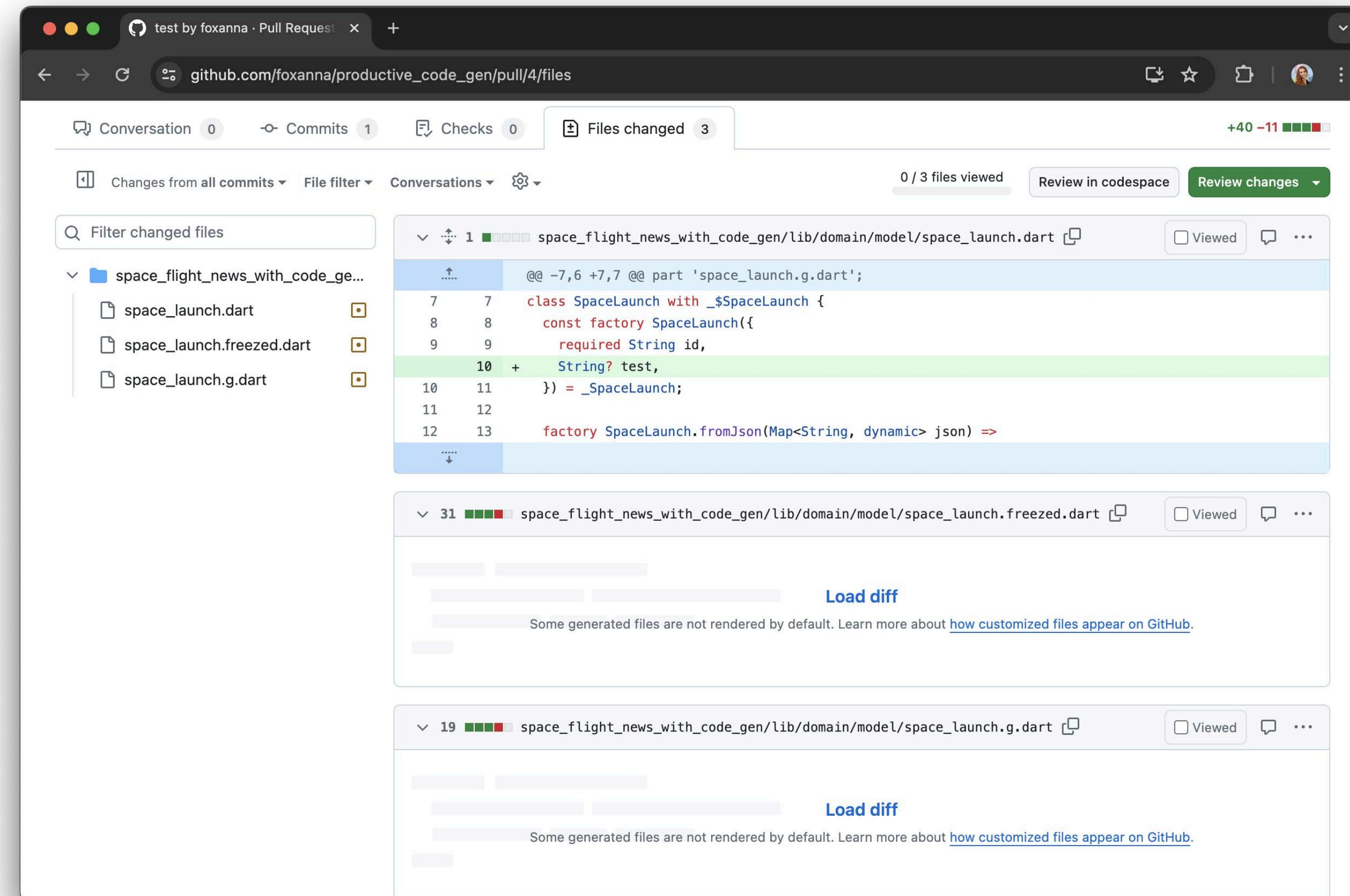
Collapse Generated Files for Review

.gitattributes

```
**/*.g.dart linguist-generated  
**/*.freezed.dart linguist-generated  
**/*.config.dart linguist-generated  
**/*.gr.dart linguist-generated  
**/*.gen.dart linguist-generated  
**/*.i69n.dart linguist-generated  
**/feature/*_test.dart linguist-generated
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Collapse Generated Files for Review



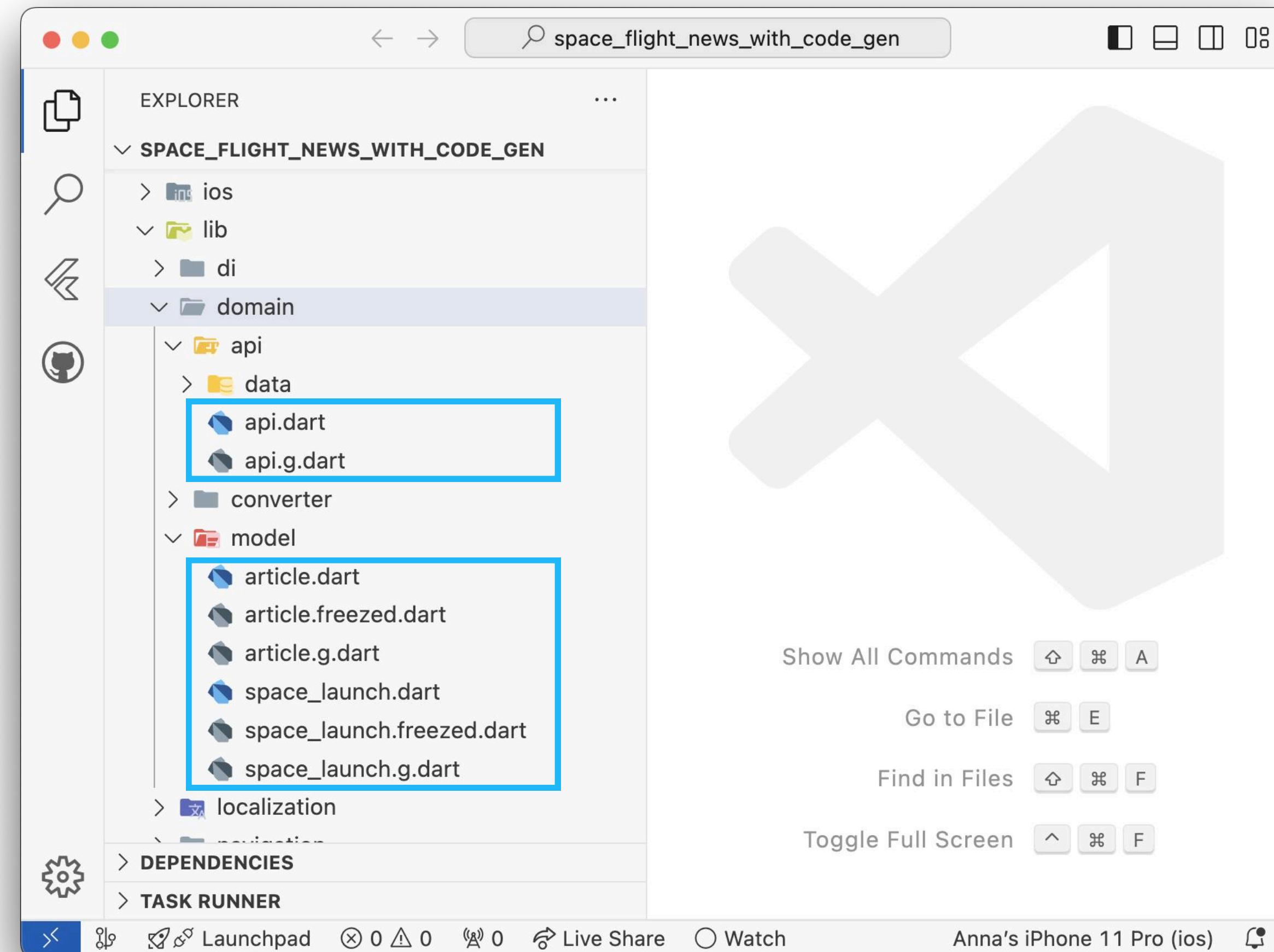
Flutterconeurope

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Collapse Generated Files in IDE

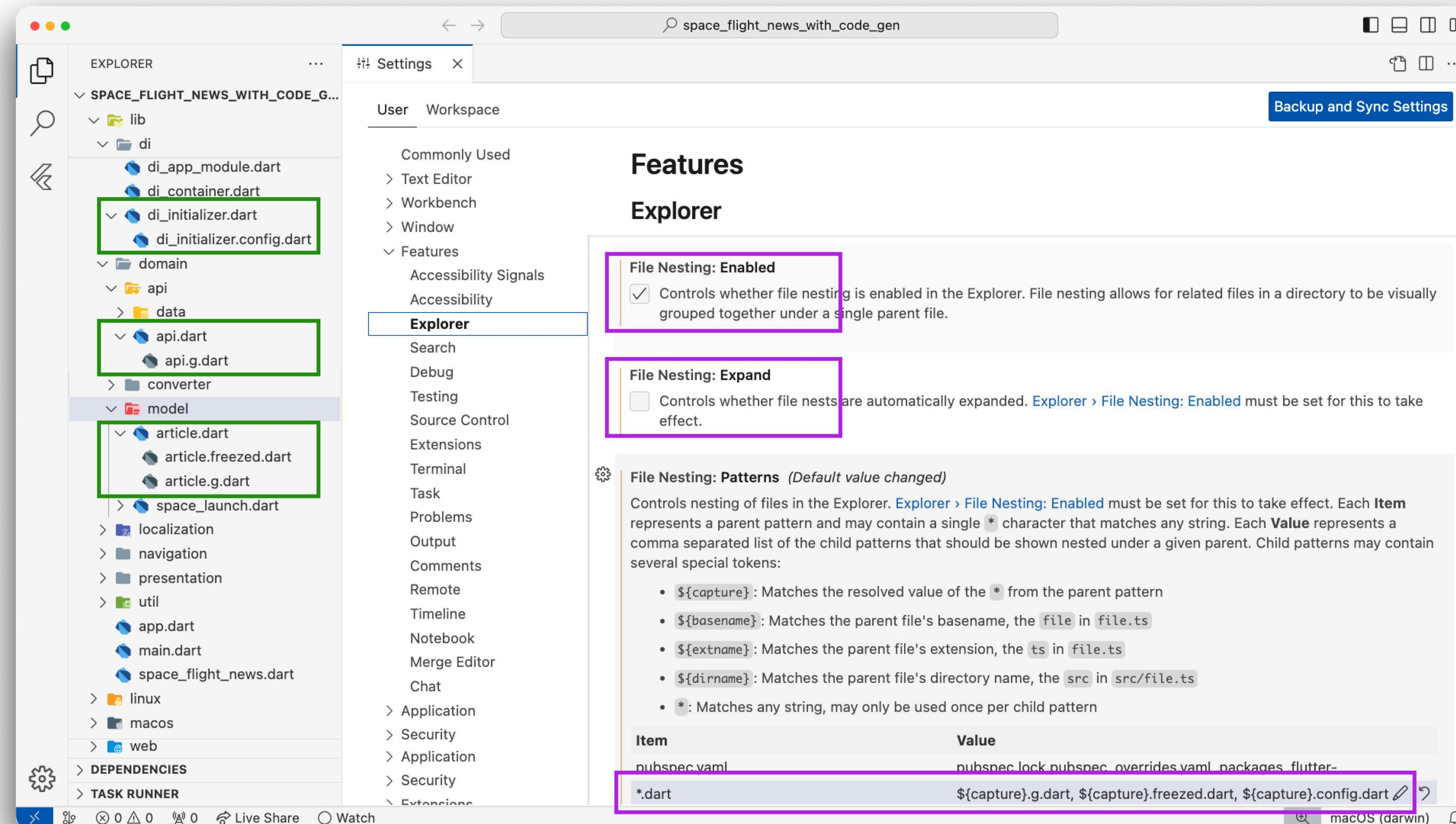
YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Collapse Generated Files in IDE



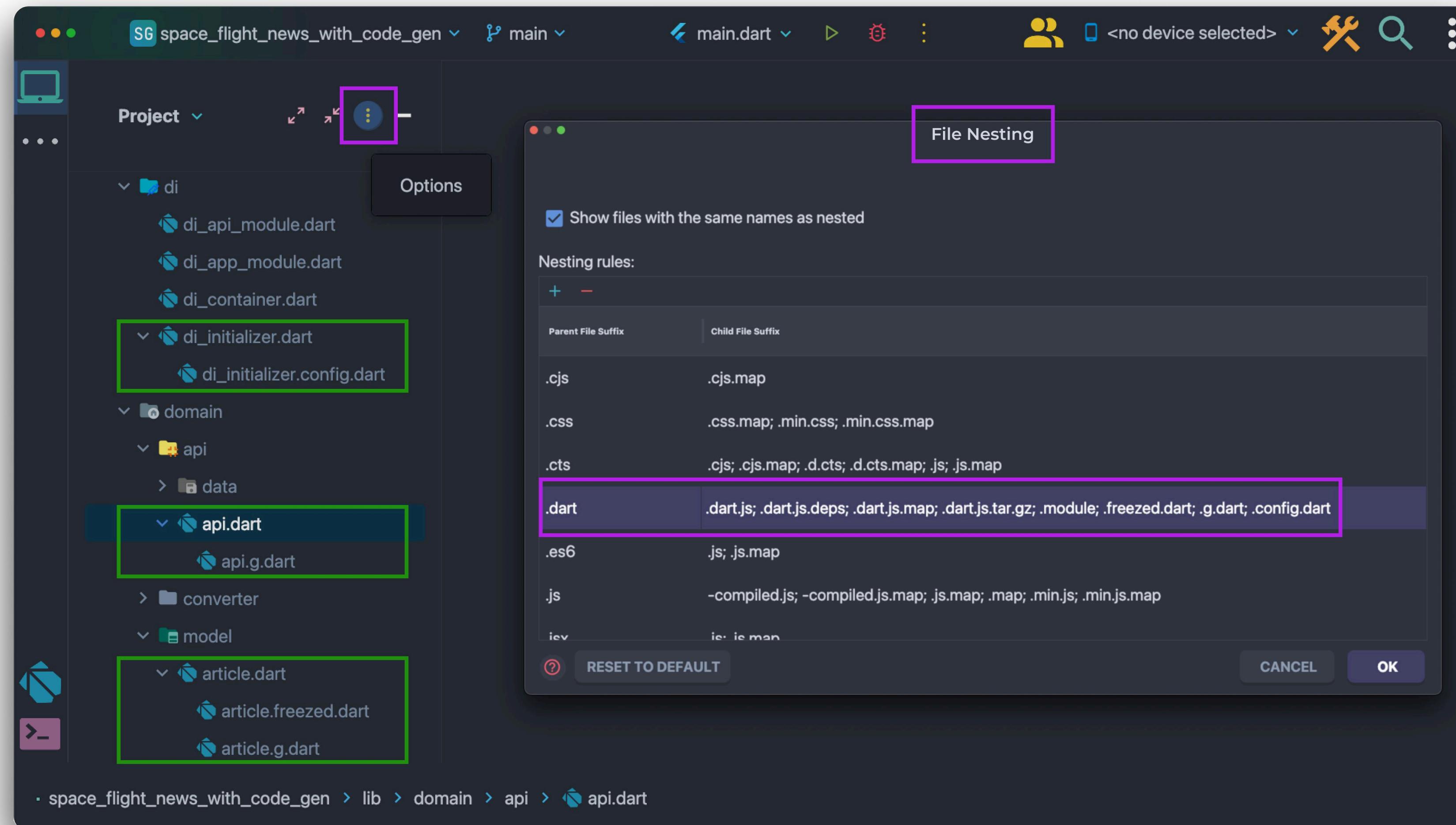
YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Collapse Generated Files in VSCode



YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Collapse Generated Files in Android Studio



Flutterconeurope

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Lock Dependencies Versions

Lock Dependencies Versions

pubspec.yaml

```
dev_dependencies:  
    auto_route_generator: 8.0.0  
    barrel_files: 0.1.1  
    bdd_widget_test: 1.7.4  
    build_runner: 2.4.11  
    flutter_gen_runner: 5.5.0  
    injectable_generator: 2.6.1  
    freezed: 2.5.3  
    json_serializable: 6.8.0  
    retrofit_generator: 8.1.0
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Control Code Generation Order

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Control Code Generation Order

```
build.yaml  
global_options:  
  freezed:  
    runs_before:  
      - json_serializable  
json_serializable:  
  runs_before:  
    - retrofit_generator
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Reuse Configured Annotations

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Reuse Configured Annotations

```
build.yaml
targets:
  $default:
    builders:
      json_serializable:
        options:
          include_if_null: false
          explicit_to_json: true
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Reuse Configured Annotations

```
@JsonSerializable(includeIfNull: false, explicitToJson: true)  
class ExampleModel {...}
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Reuse Configured Annotations

```
const serializableModel = JsonSerializable(  
    includeIfNull: false,  
    explicitToJson: true,  
);
```

```
@serializableModel  
class ExampleModel {...}
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Simplify Generation Launching

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Simplify Generation Launching

.zshrc

```
alias brb="dart run build_runner build -d"  
alias brw="dart run build_runner watch -d"
```

```
alias fpg="flutter pub get"
```

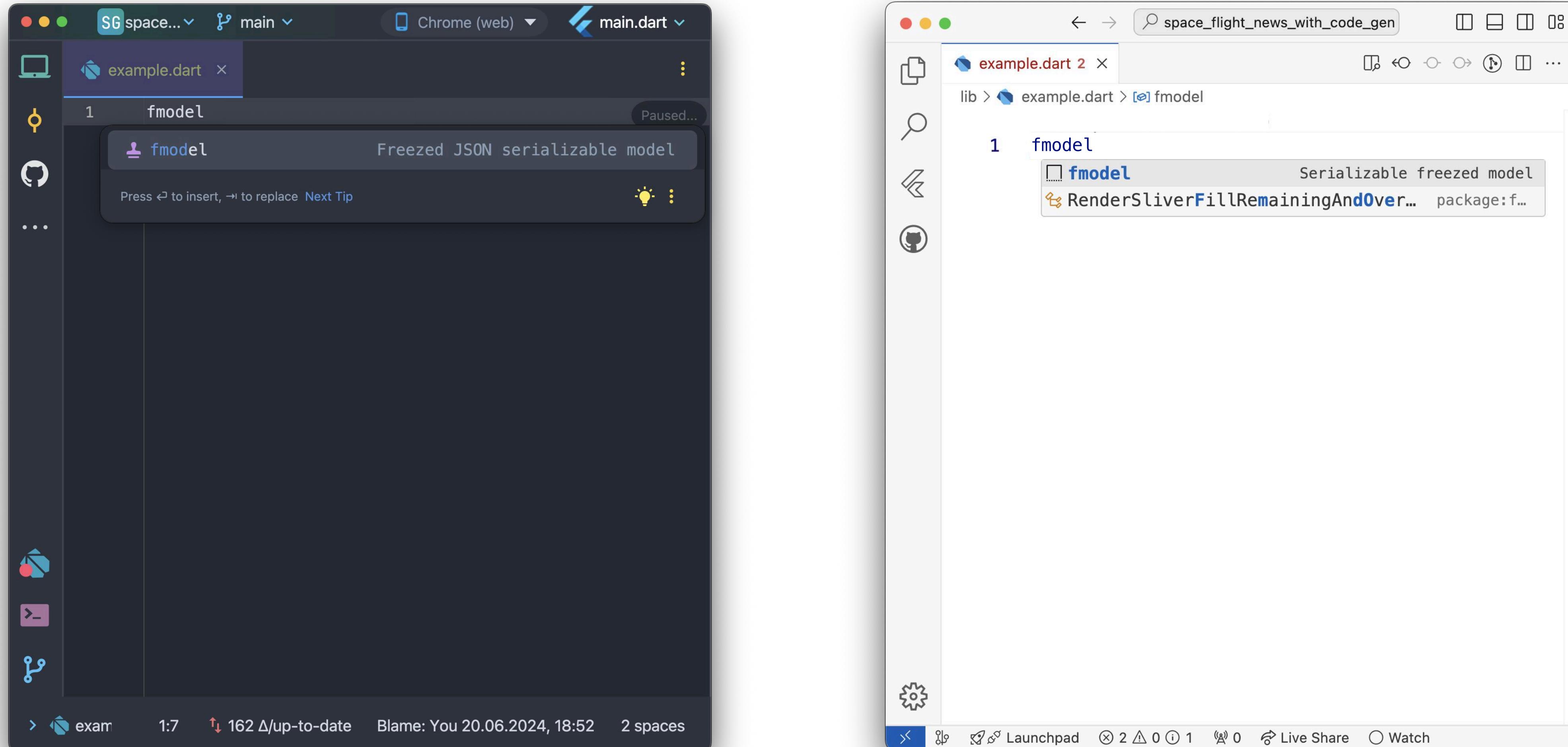
```
alias fpgrb="fpg && brb"  
alias fpgrw="fpg && brw"
```

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Create Code Snippets

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Create Code Snippets

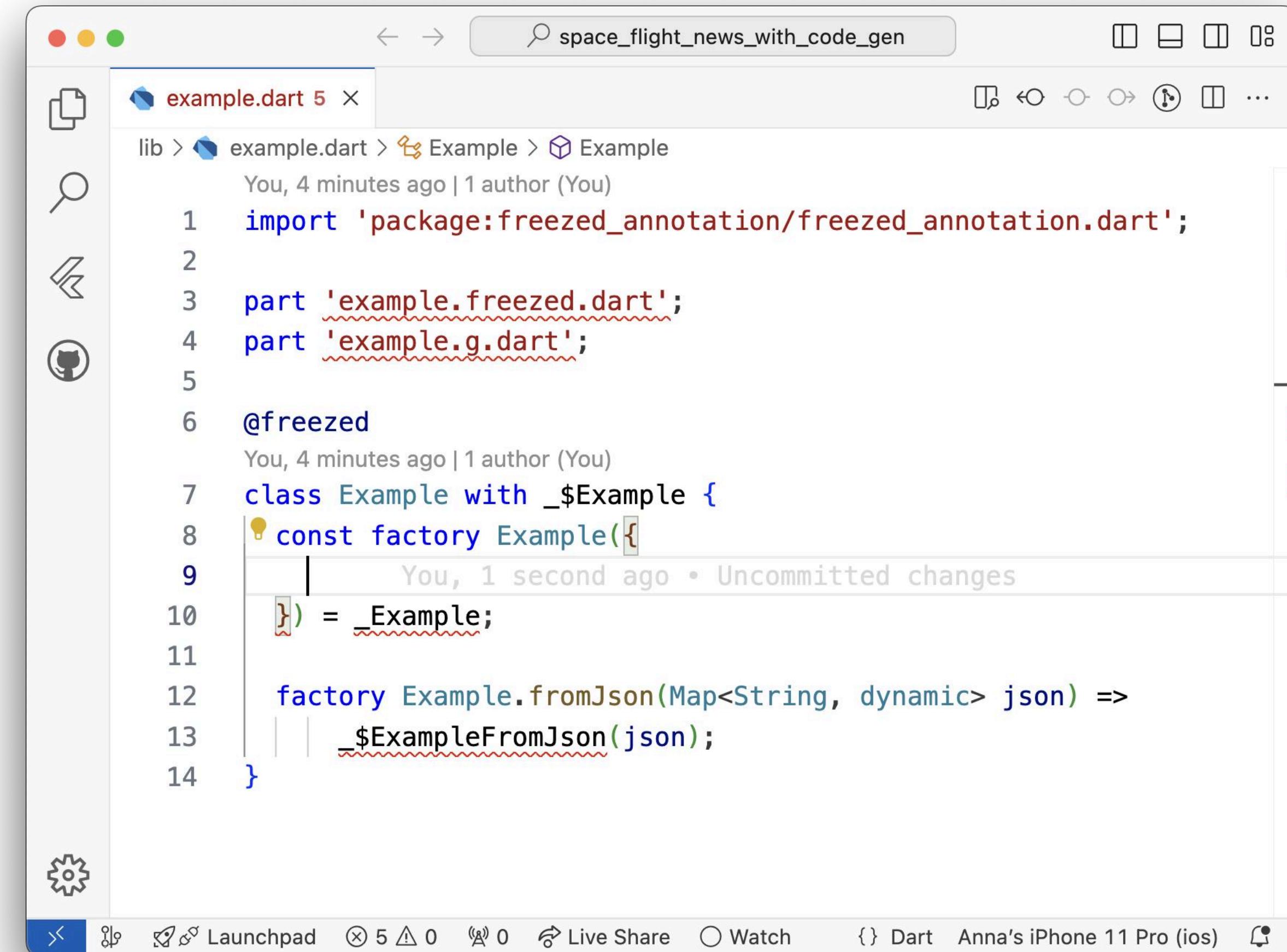


<https://www.youtube.com/@foxanna>

Flutterconeurope

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Create Code Snippets



A screenshot of a Dart code editor, likely VS Code, showing a file named `example.dart`. The code is a Dart class definition with annotations and factory methods. A cursor is positioned at the end of the first line of the constructor's body, where a tooltip displays the message "You, 1 second ago • Uncommitted changes". The editor interface includes a sidebar with icons for file operations, a search bar at the top, and various status indicators and toolbars at the bottom.

```
1 import 'package:freezed_annotation/freezed_annotation.dart';
2
3 part 'example.freezed.dart';
4 part 'example.g.dart';
5
6 @freezed
7 class Example with _$Example {
8   const factory Example({
9     You, 1 second ago • Uncommitted changes
10   }) = _Example;
11
12   factory Example.fromJson(Map<String, dynamic> json) =>
13   _$ExampleFromJson(json);
14 }
```

<https://www.youtube.com/@foxanna>

Flutterconeurope

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Update Code Coverage Report

Update Code Coverage Report

- › `flutter test --coverage`
- › `dart pub global activate remove_from_coverage`
- › `remove_from_coverage -f coverage/lcov.info \`
`-r '.g.dart$' -r '.config.dart$' -r ...`

YOUR ULTIMATE GUIDE TO CODE GENERATION PRODUCTIVITY FOR DART AND FLUTTER

Efficient Codebase Maintenance

- Optimize Code Generator Input
- Keep Packages Small
- Configure Static Analysis
- Add Generated Files to Git
- Collapse Generated Files for Review
- Collapse Generated Files in IDE
- Lock Dependencies Versions
- Control Code Generation Order
- Reuse Configured Annotations
- Simplify Generation Launching
- Create Code Snippets
- Update Code Coverage Report

**Code Generation
with Dart & Flutter:
The Ultimate Guide**



<https://bit.ly/3y9n8Gh>

Anna Leushchenko

Flutterconeurope