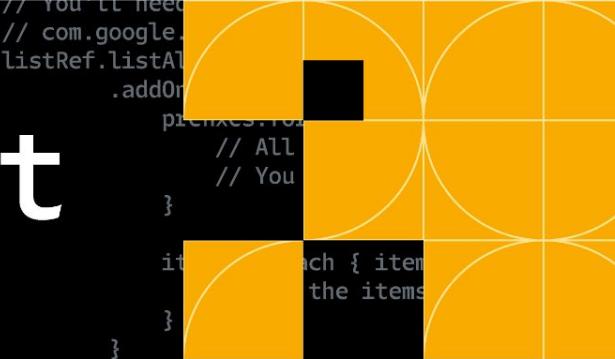


devfest



Evolving Dart: Macros

 Google Developer Groups
Bogor





Erico Darmawan Handoyo

- Parahyangan Catholic University
- Bandung Institute of Technology
- Lecturer @ Maranatha Christian University
- The Most Awesome Spectacular Legendary Programming Book Author
- Flutter tutorial creator @ YouTube
- Flutter Trainer



Erico Darmawan Handoyo
Private English Optine School (PES)



- Traveling
- Eating
- Watching movies
- Anything costs money
- Coding



youtube.com/@ericodarmawan



s.id/fb-ericodh



s.id/komunitas-flutter



s.id/ig-ericodh



s.id/flixid



s.id/in-ericodh

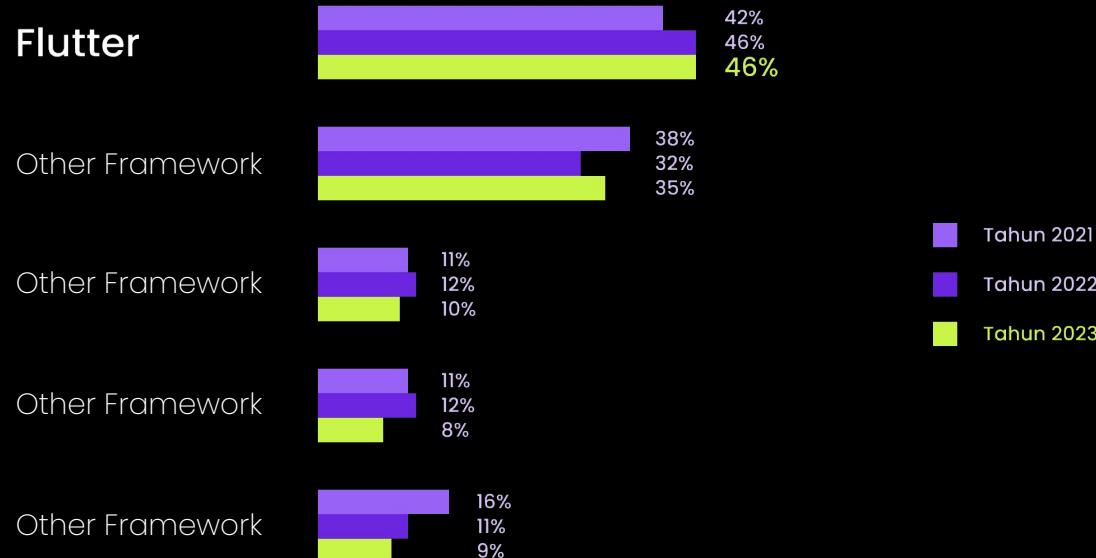


ericodarmawan.com



Erico Darmawan Handoyo
Private English Online School

Which cross-platform mobile frameworks do you use?



Why do Mobile Developers Love Flutter?

- Flutter is **platform-agnostic**
- Flutter apps offer an **excellent user interface and experience**
 - Flutter's widget system simplifies creating great-looking interfaces and works well across platforms.
 - It doesn't need platform-specific UI components.
- Numerous useful open-source packages.



Why do Mobile Developers Love Flutter?

- Flutter is **easy to learn**
 - Learning Flutter & Dart is easy for Java, Python, or C# devs due to its familiar syntax.
 - Excellent documentation.

```
public class Person {  
    public String name;  
    public int age;  
  
    public Person(String name, int age) {  
        this.name = name;  
        this.age = age;  
    }  
}
```



Erico Darmawan Handoyo

Private Training & Online School

```
Person person = new Person("Alice", 30);
```

```
class Person:  
    def __init__(self, name, age):  
        self.name = name  
        self.age = age  
  
person = Person("Alice", 30)
```

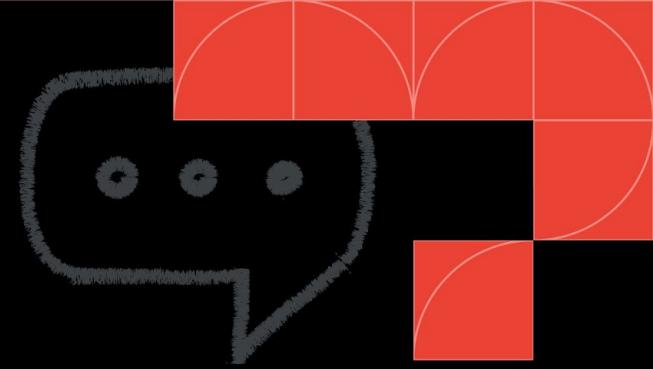
```
class Person {  
    final String name;  
    final int age;  
  
    Person({required this.name, required this.age});  
}  
  
Person person = Person(name: 'Alice', age: 30);
```

```
text:  
  'Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.red[200],  
  ),  
,),
```

devfest

```
s.star,  
r: Colors.red[500],
```

```
Text('23'),
```



Evolving Dart



Erico Darmawan Handoyo
Private Training & Online School 

Evolving Dart

- **Null Safety:** prevents null errors by ensuring variables can't be null unless specified.
- **Enhanced enum:** allow enum to have additional features such as fields, methods, and constructors.
- **Dart 3 new class modifiers:** mixin, final, seal, base, & interface class.



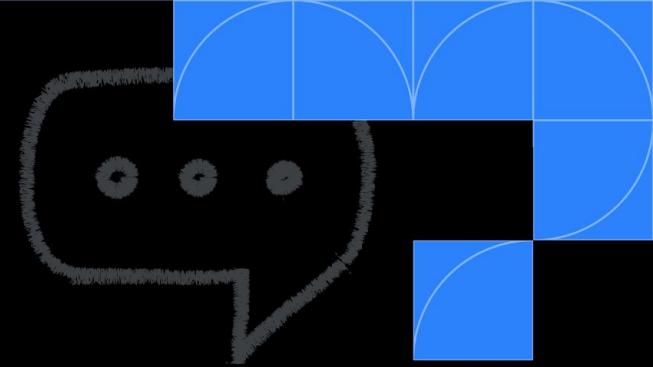
```
enum Role {  
    admin('Administrator', 'Has full access'),  
    user('User', 'Has limited access'),  
    guest('Guest', 'Has minimal access');  
  
    final String name;  
    final String description;  
  
    const Role(this.name, this.description);  
  
    void describe() {  
        print('$name: $description');  
    }  
}
```



```
text:  
  'Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.blue[200],  
  ),  
,
```

devfest

```
s.star,  
r: Colors.blue[500],  
  
Text('23'),
```



Macros



Erico Darmawan Handoyo
Private Training & Online School 



Build Runner Package

Provides a concrete way of generating files using Dart code.



```
import 'package:json_annotation/json_annotation.dart';

part 'person.g.dart';

@JsonSerializable()
class Person {
    final String name;
    final int age;

    Person({required this.name, required this.age});
}
```

```
> dart run build_runner build
```



```
// // GENERATED CODE - DO NOT MODIFY BY HAND

part of 'person.dart';

Person _$PersonFromJson(Map<String, dynamic> json) => Person(
    name: json['name'] as String,
    age: (json['age'] as num).toInt(),
);

Map<String, dynamic> _$PersonToJson(Person instance) => <String, dynamic>{
    'name': instance.name,
    'age': instance.age,
};
```



Erico Darmawan Handoyo

Private Training & Online School



What is Dart Macro?

Dart macro is a user-definable piece of code that takes in other code as parameters and operates on it in real-time to create, modify, or add declarations.

```
import 'package:json/json.dart';

@Json Codable()
class Person {
    final String name;
    final int age;

    Person({required this.name, required this.age});
}
```

```
augment library 'file:///Users/ericodh/Dart%20Project/macros_demo/bin/macros_demo.dart';

import 'dart:core' as prefix0;

augment class Person {
    external Person.fromJson(prefix0.Map<prefix0.String, prefix0.Object?> json);
    external prefix0.Map<prefix0.String, prefix0.Object?> toJson();
    augment Person.fromJson(prefix0.Map<prefix0.String, prefix0.Object?> json, )
        : this.name = json[r'name'] as prefix0.String,
        this.age = json[r'age'] as prefix0.int;
    augment prefix0.Map<prefix0.String, prefix0.Object?> toJson() {
        final json = <prefix0.String, prefix0.Object?>{};
        json[r'name'] = this.name;
        json[r'age'] = this.age;
        return json;
    }
}
```



How to Use Macro?

1. Switch to the [Flutter master channel](#).
2. Run `dart --version` and make sure you have Dart version **3.5.0-152** or later.
3. Edit the [SDK constraint](#) in your pubspec to require the Dart version:
sdk: ^3.5.0-152.
4. [Add the package json to dependencies](#): `dart pub add json`.

How to Use Macro?

5. Enable the experiment in your package's `analysis_options.yaml` file. file at the root of your project:



6. Run your project with the experimental flag:

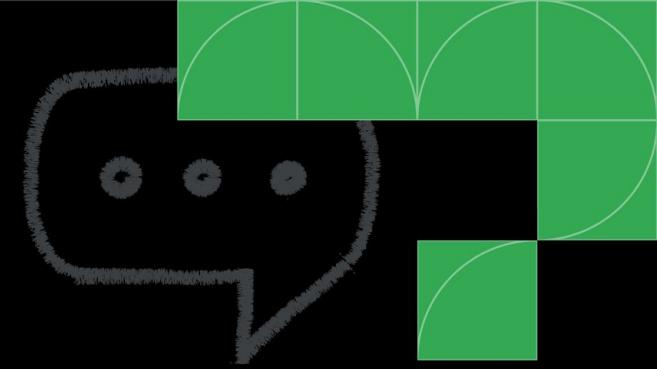
```
dart run --enable-experiment=macros your_file.dart
```

```
text  
  'Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
,
```

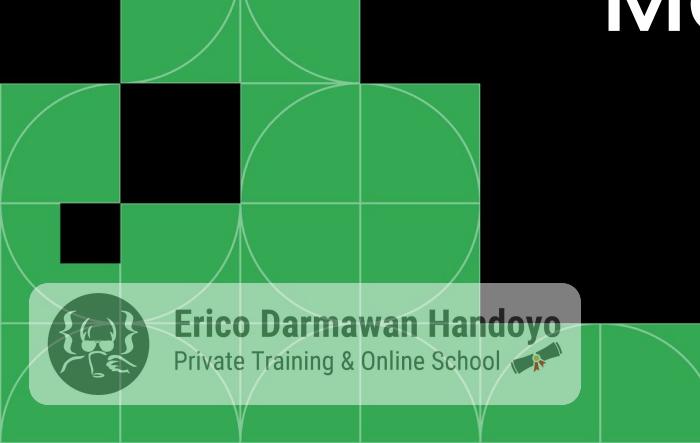
devfest

```
s.star,  
r: Colors.green[500],
```

```
Text('23'),
```



How to Create Macro (Demo)



Erico Darmawan Handoyo
Private Training & Online School





13.39

Good afternoon, Jennie!
Let's book your favorite movie!
IDR 110.000

Search movie

Now Playing

Saving Bikini Bottom
Beetlejuice Beetlejuice
The Garfield Movie

Promotions

Get 40% discount for foods & drinks every Wednesday

Buy 1 Get 1 Free Ticket every Wednesday

upcoming

Erico Darmawan Handoyo
Private Training & Online School

codecon

13.55

codecon

Register Contact

August 8 - 9, 2024

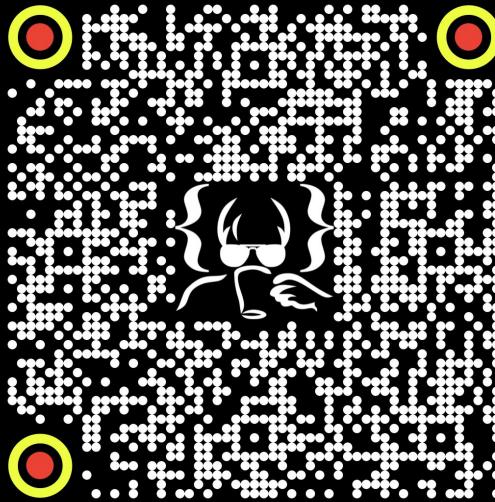
The Ultimate Event for Developers

Intercontinental Bali Resort, Bali, Indonesia
IDR 750,000

Register Now

Seminars, Talks, & Workshops

Thank You



SPECIAL
OFFER
50% OFF

IDR 399.000
IDR 199.000
9 - 13 November 2024

s.id/flixid
s.id/codecon

Kode Promo: **DEVFESTBOGOR24**