

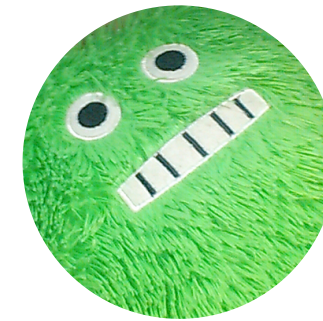
MelosでFlutterのmonorepo環境を構築する

自己紹介

Toi Yoshimaru (@0maru_dev)

twitter_login のメンテナ

https://github.com/0maru/twitter_login



Melos とは

Melos is a CLI tool used to help manage Dart projects with multiple packages (also known as mono-repos). It is currently still in active development however is in use on projects such as FlutterFire.

-> 複数のFlutter プロジェクト/Dart パッケージを管理するCLIツール

Melos が使用されているプロジェクト

[FirebaseExtended/flutterfire](#)

[fluttercommunity/plus_plugins](#)

[rrousselGit/river_pod](#)

Melos でできること

複数のプロジェクトを管理することができる

そもそもFlutterで複数のプロジェクトとは??

Monorepo とは？

Flutter

```
- project
  - packages/
    - app/                # main.dart があるエントリーポイント
    - oooo_login/         # 自社のOAuth用のライブラリ
    - oooo_utils/         # よく使う処理やWidget をまとめたライブラリ
```

複数言語

```
- project
  - typescript
    - client
    - utils
    - BFF
  - go
    - server
    - batch
  - python
    - tools
```

Melos の使い方

1. Melos のインストール
2. プロジェクトに対してMelos を有効にする
3. Melos の構成ファイルを作成する

Melos のインストール

```
dart pub global activate melos
```

設定ファイルの作成

```
touch melos.yaml
```

melos.yaml

```
name: flutter_monorepo_project
packages:
  - "*"
  - "packages/**"

scripts:
  get:all:
    run: |
      melos exec -c 1 -- \
        "flutter pub get"
```

<https://melos.invertase.dev/configuration/overview>

Melos の実行方法

スクリプトを指定して実行する

```
melos run get:all
```

スクリプト一覧をみてから実行する

```
melos run
```

```
Enter your choice? 1
```

Melos のオプション

–scope（実行対象を指定する）

```
melos run --scope="app" -- "flutter test -t integration_test"
```

–ignore（特定のディレクトトを実行対象外にする）

```
melos run --ignore="*example*" -- "flutter pub get"
```

–flutter（Flutter に依存しているプロジェクトのみで実行する）

```
melos run --flutter -- "flutter run build"
```

ライブコーディング

まとめ

会社紹介

新卒・中途のエンジニア募集中です！！