

# Flutter 소개 및 개발환경 셋업

PoApper 2022-1 Flutter Seminar 1  
성치호

2022-03-23

# 개요

- Flutter 세미나 안내
- Flutter 소개
- Flutter 개발 셋업
- 과제

# Flutter 세미나 안내

## 일정

- Flutter 소개 및 개발환경 셋업
- 문법
- Stateless Widget
- Stateful Widget
- Provider
- Future & API 연동
- Animation
- ...?

# Flutter 세미나 안내

## 형식

- 매주 목요일에 녹화본, PDF 공유
  - Youtube, GitHub
- 세미나 시간동안은 빠르게 설명
- 세미나 직후 Zoom을 통한 질문 시간 별도 운영
  - 세미나 중간 질문 가능, 종료 후 답변할 수도 있음
  - 세미나 이외 시간에는 Slack or 카카오톡으로 질문가능

# Flutter 세미나 안내

## 평가

- 주간 과제
  - 세미나 종료 직전에 안내
  - 제출기한 : 다음주 화요일 10시
  - 미제출 및 지각시 이수 불가능
- 최종 과제
  - 학기 종료 후 최종 과제 안내
  - 미제출 및 지각시 이수 불가능
- 출석은 미반영

# Flutter 소개

## Flutter란?

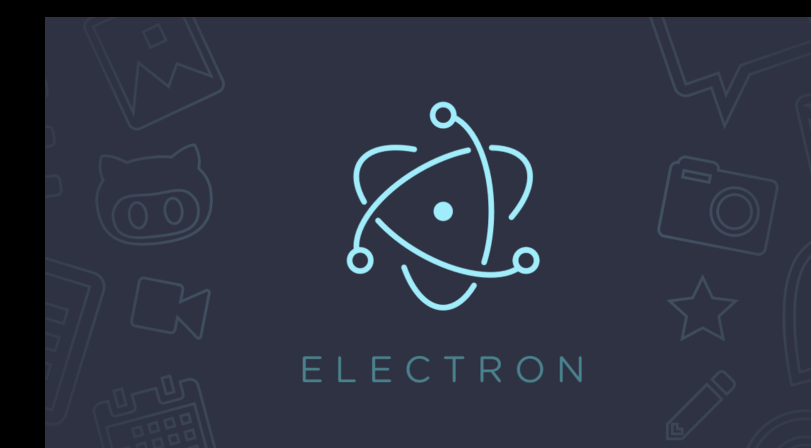
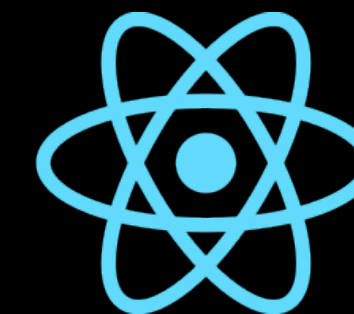


- Flutter
  - 구글이 출시한 오픈 소스 크로스 플랫폼 GUI 애플리케이션 프레임워크
  - 지원 플랫폼
    - Stable : Android, iOS, Web
    - Beta : Linux, Window, macOS
  - 다트 기반

# Flutter 소개

## 크로스 플랫폼?

- Cross-platform
  - 여러 종류의 컴퓨터 플랫폼에서 동작 가능
- Cross-platform 개발 방식
  - 프로그래시브 웹 앱
  - 하이브리드
  - 네이티브 컴파일 솔루션
- Cross-platform 프레임워크 예시
  - Qt, React Native, Xamarin, Electron, Web?



# Flutter 소개

## 특징

- Dart 기반
  - C++, 객체지향
  - 동적, 정적 타입 모두 이용 가능
  - Null safety
- Stateless, Stateful Widget
- 매우 많은 지원 플랫폼
- 네이티브로 컴파일됨 -> 네이티브급 성능
- Hot reload, Hot restart
- Pub.dev <- 다양한 라이브러리
- 코드로 디자인



# Flutter 소개

## 추천 사이트

- 라이브러리 검색
  - <https://pub.dev>
- 공식문서
  - <https://docs.flutter.dev>   간단
  - <https://api.flutter.dev/index.html>   자세
- 커뮤니티
  - <https://github.com/flutter/flutter/issues>
  - <https://stackoverflow.com>

# Flutter 개발 셋업

## 페이지 안내

- Window 사용자 : 12page부터
- macOS 사용자 : 16page부터

# Flutter 개발 셋업(Window)

## IDE, Editor 설치 및 세팅

- Android Studio 설치
  - <https://developer.android.com/studio>
  - Android Studio 설치
  - 실행 후 초기 Editor 세팅
  - Plugin Tab에서 Flutter 검색 후, 설치 후 재시작

# Flutter 개발 셋업(Window)

## SDK 설치 및 세팅(1)

- Flutter SDK 다운로드
  - <https://docs.flutter.dev/get-started/install/windows>
  - Flutter SDK zip 파일 다운로드
  - 원하는 장소에 압출 풀기
- Flutter SDK 환경변수 설정
  - 윈도우 검색창 > 고급 시스템 설정 > 환경 변수 > Path > 편집 누르기
  - 압축푼 곳\flutter\bin 등록하기
  - Cmd 열어서 flutter doctor 쳐서 버전 뜨는지 확인하기

# Flutter 개발 셋업(Window)

## SDK 설치 및 세팅(2)

- Flutter SDK 다운로드
  - <https://docs.flutter.dev/get-started/install/windows>
  - Flutter SDK zip 파일 다운로드
- Android SDK 세팅
  - Android Studio > Preferences > Apperarance & Behavior > System Settings > Android SDK 로 이동
  - SDK tools에서 Android SDK Command-line Tools (latest) 선택 후 Apply, OK

```
$ flutter doctor --android-licenses
$ flutter config --no-enable-web # web build 비활성화
$ flutter doctor # Android toolchain 항목이 통과인지 확인
```

# Flutter 개발 셋업(Window)

## Emulator 세팅 및 테스트

- 사전 준비
  - Android Studio에서 New Flutter Project 누르기
  - SDK path /Users/<user\_name>/development/flutter 로 설정
  - 프로젝트 생성 완료하기
- Android Emulator
  - Tools > Device Manager 에서 기본으로 설치된 sdk gphone64 arm64 선택 후 실행
    - 설치된 avd가 없다면 create device.
  - Run 'main.dart' 버튼 누르기
  - 하단의 run 탭에서 진행사항 파악
  - 앱 실행 화면이올바르게 나타나는지 확인
- Flutter Setup 완료 확인

```
$ flutter doctor # 모든 항목이 통과인지 확인
```

# Flutter 개발 셋업(macOS)

## IDE, Editor 설치 및 세팅

- Android Studio 설치
  - <https://developer.android.com/studio>
  - PC의 cpu(silicon or intel)에 해당하는 Android Studio 설치
  - 실행 후 초기 Editor 세팅
  - Plugin Tab에서 Flutter 검색 후, 설치 후 재시작
- Xcode 설치
  - App Store를 통해 Xcode 설치

# Flutter 개발 셋업(macOS)

## SDK 설치 및 세팅 (1)

- Flutter SDK 다운로드
  - <https://docs.flutter.dev/get-started/install/macos>
  - Flutter SDK zip 파일 다운로드

```
$ cd ~  
$ mkdir development # Flutter SDK를 설치할 폴더  
$ cd development  
$ unzip ~/Downloads/<flutter_sdk_name>  
$ echo 'export PATH="$PATH:$HOME/development/flutter/bin"' >> ~/.zshrc  
$ flutter doctor # Flutter version 확인
```



# Flutter 개발 셋업(macOS)

## SDK 설치 및 세팅 (2)

- Android SDK 세팅
  - Android Studio > Preferences > Apperarance & Behavior > System Settings > Android SDK 로 이동
  - SDK tools에서 Android SDK Command-line Tools (latest) 선택 후 Apply, OK

```
$ echo 'export ANDROID_HOME="$HOME/Library/Android/sdk"' >> ~/.zshrc
$ flutter doctor --android-license
$ flutter config --no-enable-web # web build 비활성화
$ flutter doctor # Android toolchain 항목이 통과인지 확인
```

# Flutter 개발 셋업(macOS)

## SDK 설치 및 세팅 (2)

- Android SDK 세팅
  - Android Studio > Preferences > Apperarance & Behavior > System Settings > Android SDK 로 이동
  - SDK tools에서 Android SDK Command-line Tools (latest) 선택 후 Apply, OK

```
$ echo 'export ANDROID_HOME="$HOME/Library/Android/sdk"' >> ~/.zshrc
$ flutter doctor --android-license
$ flutter config --no-enable-web # web build 비활성화
$ flutter doctor # Android toolchain 항목이 통과인지 확인
```

# Flutter 개발 셋업(macOS)

## Emulator 세팅 및 테스트(1)

- 사전 준비
  - Android Studio에서 New Flutter Project 누르기
  - SDK path /Users/<user\_name>/development/flutter 로 설정
  - 프로젝트 생성 완료하기
- Android Emulator
  - Tools > Device Manager 에서 기본으로 설치된 sdk gphone64 arm64 선택 후 실행
  - Run 'main.dart'
  - 하단의 run 탭을 통해 진행사항 파악
  - 앱 실행 화면이 올바르게 나타나는지 확인

# Flutter 개발 셋업(macOS)

## Emulator 세팅 및 테스트(2)

- iOS Emulator

```
$ sudo gem install cocoapods  
$ open -a Simulator
```

- 상단의 Flutter Device Selection 칸에서 iOS 기기 선택후 Run 'main.dart'
- 하단의 run 탭을 통해 진행사항 파악
- 앱 실행 화면이 올바르게 나타나는지 확인

- Flutter Setup 완료 확인

```
$ flutter doctor # 모든 항목이 통과인지 확인
```

# 과제

## 세미나 과제 공통 설명

- Public GitHub Repo 생성
  - 링크 제출.
- Repo 구조 형식
  - <Homework Repo>
    - Chapter1
      - ...
    - Chapter2

# 과제

## 1주차 과제

- 개발환경 세팅 완료하기.
- 숙제 제출용 Github Repo 생성 후 링크 보내기.
- 개발 환경 세팅 완료했다는 인증샷 Github에 푸시해놓기.
- android emulator에서 앱 실행되는 모습(mac의 경우 ios까지), 안드로이드 스튜디오, flutter doctor 실행 결과 모두 보이게 캡처

