Flutter Setup & Dart Intro

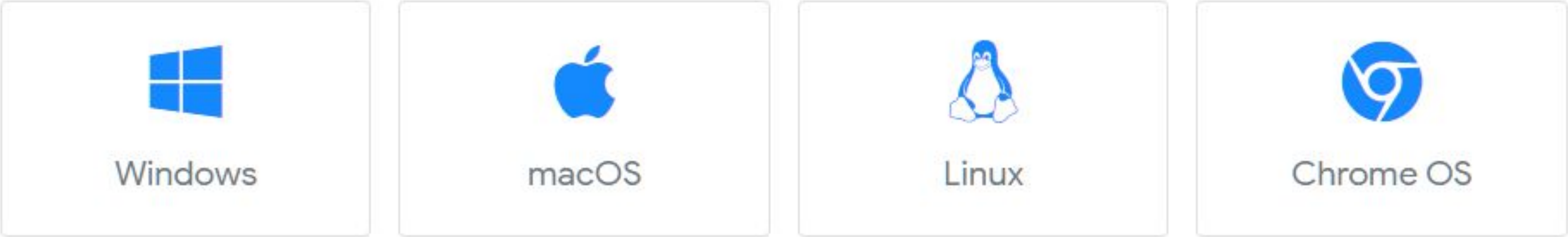
Mpho Mbonani • Software Developer

The Fellowship of The Framework

Getting Started
Get the Flutter SDK
Flutter and Dart extensions
Create a project

What is Dart?
Object-Orientation
Variables, Functions, Types
Final & Const
Resources

Getting Started



System Requirements

- Operating Systems: Windows 7 SP1 or later (64-bit), x86-64 based.
- Disk Space: 1.64 GB (does not include disk space for IDE/tools).
- Tools: Flutter depends on these tools being available in your environment.
  - Windows PowerShell 5.0 or newer
  - Git for Windows with Use Git from the Windows Command Prompt option.

Get the Flutter SDK

Installation Bundle

- Download the installation bundle to get the latest stable release of the Flutter SDK
- https://storage.googleapis.com/flutter_infra_release/releases/stable/windows/flutter_windows_2.2.3-stable.zip

- Extract the zip file and place the contained flutter in the desired installation location for the Flutter SDK (for example, C:\Users\<your-user-name>\Documents)

Flutter and Dart extensions

VS Code

- VS Code is a lightweight editor with Flutter app execution and debug support.
- https://code.visualstudio.com/

- Start VS Code.
- Invoke View > Command Palette….
- Type "install", and select Extensions: Install Extensions.
- Type "flutter" in the extensions search field, select Flutter in the list, and click Install. This also installs the required Dart plugin.

Create a project

What is Dart?

Breakdown

- Dart is a statically typed, object-oriented programming language developed by Google. It’s primarily used for creating frontend user interfaces for the web and mobile apps.
- Dart is a compiled language. That means, that your code is parsed by a compiler and transformed to native code.

Object-Orientation

Breakdown

- Everything in Dart is an object
- The idea behind object-orientation is that all data structures are seen as objects. A bit like in the "real world".
- Objects are created with the help of "Classes" because every object needs a blueprint based on which you can then create or instantiate it.

Four Pillars

- Abstraction
- Polymorphism
- Inheritance
- Encapsulation

Variables, Functions, Types

Variables

- var name = 'X Æ A-12';
- Variables store references. The variable called name contains a reference to a String object with a value of “X Æ A-12”

Functions

- In Dart, functions are objects and have a type, Function. This means that functions can be assigned to variables or passed as arguments to other functions

Types

- Defines type of data expected & useful for avoiding mistakes/bugs
- Numbers (int, double), Strings (String), Booleans (bool), Lists (List), Maps (Map) and more

final & const

Breakdown

- If you never intend to change a variable, use final or const. A final variable can be set only once; a const variable is a compile-time constant.

Final

```
final name = 'X Æ A-12';
```

A final top-level or class variable is initialized the first time it’s used.

You can’t change the value of a final variable

Const

```
const bar = 1000000;
```

Use const for variables that you want to be compile-time constants.

If the const variable is at the class level, mark it static const.

Resources

Getting Started

https://flutter.dev/docs/get-started/install

Dart Language Tour

https://dart.dev/guides/language/language-tour

Github Repo

https://github.com/mpho-mbonani

One Framework to rule them all, One Framework to find them, One Framework to bring them and in the darkness bind them.

Software Development Committee of Practise

Technology Enablement

IQbusiness