How To Develop Mobile Apps Using Python?

The importance of smartphones in today's life cannot be overstated. You use mobile applications for a wide variety of tasks, from daily social media engagement to accessing banking details. When you talk about mobile applications, you often hear about iOS or Android development. But what about the other mobile operating systems available on the market? Wouldn't it be great if there was a way to bring your app to all of these platforms? Well, there is, lets get our hands dirty and develop mobile apps using python.

Photo by William Hook on Unsplash

Python

Python is a general-purpose programming language and used for different domains. It has been popular in data science and machine learning as it offers out of the box libraries.

The Python language reached its popularity on desktop platforms, but with time attained popularity in app development as well. It is flexible and uses a lot of pre-written components. In fact, it is possible to use some of Python modules as libraries for other languages.

There are a lot of third-party packages or modules that you can import and use, including all the tools needed to do mobile development.

Develop Mobile Apps Using Python

Some of the use cases which are best suited for development using Python are:

- Command-line utility apps
- Complex apps like analytical tools that can operate on huge datasets, Banking mobile apps, Trading apps, and more.
- Game development
- System administrative apps

Frameworks

To develop intuitive mobile apps, you need a great framework. There are two popular frameworks that you go very well with python development - Kivy and BeeWare.

Kivy

It works very well with Android, IOS, Linux, OS X and Windows too making it a popular choice for your frontend development. It uses custom UI instructions that allows you to use same component on different platforms.

BeeWare

It can be used on Android and IOS for app development. It uses underline native components of the operating system that you have chosen to build your app on. Its versatile native elements gives a look and feel of a native app.

These frontend modules interact with backend written in Python to give you an interactive and sturdy mobile app.

Get Started - Hello World

You can follow these simple steps to get started with your app development.

Set Environment

As a general thumb rule, it is always a best practice to set up your environment. This helps you to maintain different versions of your libraries and isolates your code base.

Install Libraries

Once you set up your environment, you can install your libraries.

```
pip install kivy # to install Kivy

pip install briefcase # to install BeeWare
```
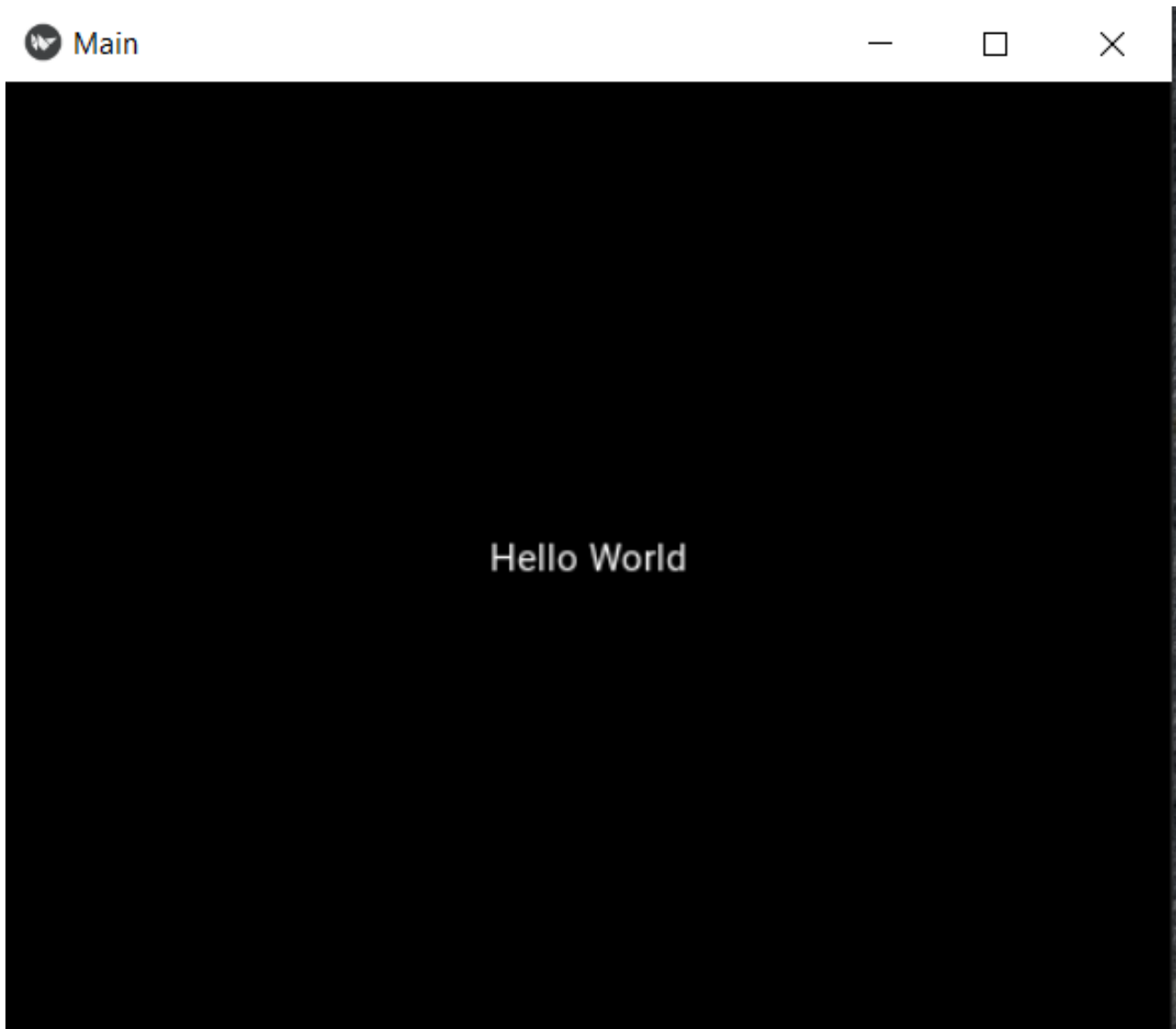
Backend

Open your IDE and let's code.

```
from kivy.app import App

from kivy.uix.label import Label

    class Main(App):

        def build(self):

        return Label(text='Hello World')

    Main().run()
```

Execution

When you execute your above code sample, you will see the output generated as below:

Your first app developed using python

Popular apps developed with Python

Some of the popular apps that developed using python are:

Pinterest

It is a widely used app on internet and millions of users use this app to share visual content. It is developed using Python's Django framework with works very well with massive datasets.

Instagram

It is a famous platform to share photos. With so much data to handle, the app needed a sturdy backend that was developed on Python.

Spotify

It is a live-streaming music library where you can get almost every music record you would love to listen. It has many services that are developed in Python connecting to its messaging tool to give you a seamless experience.

Takeaway

Python offers a wonderful solution that generates native apps for all mobile operating systems while still using one single code base. The language is very powerful, flexible and fast. It is free, secure and cross-platform. It has a lot of advantages when you compare it with other languages like Java or C++. The primary being the integration of many third-party libraries makes it a robust platform to develop yur mobile apps.

With Python at your disposal, get the most out of it to develop your vibrant, intuitive mobile application and revolutionize the app world.