**Deploy Python Applications to Different PaaS Cloud Platforms**

1. **Python Anywhere** - Online Integrated Development Environment and Web Hosting Service Based on the Python Programming Language. It provides in-browser access to server-based Python and Bash command-line interfaces, along with a code editor with syntax highlighting.
   * Official Site (Login / Sign Up) - <https://pythonanywhere.com/>.
   * You Can Create Only 1 Website with Basic Free Plan.
   * Supporting Python Web Frameworks are Django, Flask, Web2py, Bottle.
   * Deploy Flask Web Application to PythonAnywhere PaaS Cloud Platform – [Click Here!](https://medium.com/@tossia13/lets-take-it-online-deploying-your-flask-application-to-pythonanywhere-3ab87c1c851c)
2. **Heroku Cloud** - Heroku is a Platform as a service (PaaS) that enables developers to build, run, and operate applications entirely in the cloud.
   * Heroku Official Site (Login / Sign Up) - <https://heroku.com/home/>.
   * You can Create Up to 5 Websites with basic Free Plan.
   * Supporting Programming Languages are Node.JS, Ruby, Java, PHP, Python, Go, Scala, Clojure.
3. **Microsoft Azure Cloud Platform -** It is an open and flexible cloud computing platform which helps in development, data storage, service hosting, and service management. The Azure tool hosts web applications over the internet with the help of Microsoft data centers.

* Azure Portal (Login / Sign Up) - <https://portal.azure.com/>
* Deploy Python Django / Flask Web Application to Azure App Service (PaaS).
* Build & Deploy Python Azure Functions (Serverless) with / without Docker Containers.

1. **Amazon Web Service (AWS) -** AWS is a platform that offers flexible, reliable, scalable, easy-to-use and cost-effective cloud computing solutions. AWS is a comprehensive, easy to use computing platform offered Amazon. The platform is developed with a combination of infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS) offerings.

* AWS Official Site - <https://aws.amazon.com/>
* AWS Portal (Login / Sign Up) - <https://aws.amazon.com/console/>
* Deploy Python Django / Flask Web Application to AWS Beanstalk (PaaS). Its Linux by Default.
* Build & Deploy Python AWS Lambda using Zappa / Chalice (Serverless).

1. **Open Shift** – Its Red Hat's public cloud application deployment and hosting platform. Get on-demand access to Open Shift to build, deploy and manage scalable containerized applications, operated and supported by Red Hat.

* S
* Reference: <https://www.openshift.com/blog/getting-started-python>

**Prerequisites**

* Download Git - <https://git-scm.com/downloads>
* Download Heroku CLI Tool - <https://devcenter.heroku.com/categories/command-line>
* Download & Install Python Latest Version - <https://www.python.org/downloads/>
* Download & Install Docker - <https://docs.docker.com/desktop/>
* Once you Installed Python, Install below Python Packages in Command Line / Terminal.

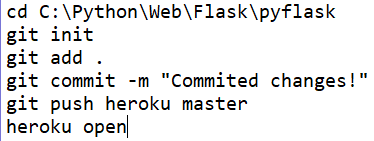


**1. Python Flask**

* **Popular**, Extensible Web Micro Framework for building web applications with Python.
* Flask Official Site - <https://flask.palletsprojects.com/en/1.1.x/>
* **Build & Deploy Python Flask Web Application to Heroku PaaS Cloud Platform.**



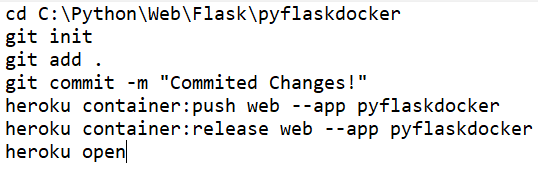
* Let’s Keep above 4 files under this Location **C:\Python\Web\Flask\pyflask**
* Go to Heroku Dashboard <https://id.heroku.com/> , Create New Application **pyflask**.
* Now Save Below Commands in Batch File **pyflask\_run.bat** under **C:\Python\Web\Flask**



* You will get URL <https://pyflask.herokuapp.com/> once Flask Web App is Successfully Deployed.
* **Build & Deploy Python Flask Web Application using Docker to Heroku PaaS Cloud Platform.**
* Clone the Git Project



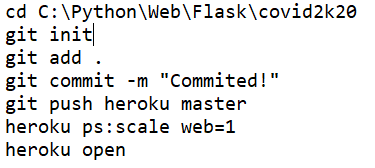
* Go to Heroku Dashboard <https://id.heroku.com/> , Create New Application **pyflaskdocker**.
* Now Save Below Commands in Batch File **pyflaskdocker\_run.bat** under **C:\Python\Web\Flask**



* GitHub Code Repository: <https://github.com/akashjeez/pyflaskdocker/>
* **Build & Deploy Python Dash Web Analytics Application to Heroku PaaS Cloud Platform**
* Clone the Git Project

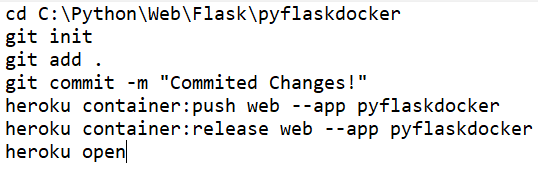


* Go to Heroku Dashboard <https://id.heroku.com/> , Create New Application **covid2k20**.
* Now Save Below Commands in Batch File **covid2k20\_run.bat** under **C:\Python\Web\Flask**



* GitHub Code Repository: <https://github.com/akashjeez/corona-virus-2020/>
* **Build & Deploy Python Dash Web Analytics Application Using Docker to Heroku PaaS Cloud Platform**
* **Build & Deploy Python Streamlit Application to Heroku PaaS Cloud Platform**
* **Build & Deploy Python Streamlit Application Using Docker to Heroku PaaS Cloud Platform**
* **Build & Deploy Python Streamlit Application Using Docker to Heroku PaaS Cloud Platform**
* Clone the Git Project



* Go to Heroku Dashboard <https://id.heroku.com/> , Create New Application **pyflaskdocker**.
* Now Save Below Commands in Batch File **pyflaskdocker\_run.bat** under **C:\Python\Web\Flask**
  + - 
* GitHub Code Repository: <https://github.com/akashjeez/pyflaskdocker/>
* **Build & Deploy Python Flask Web Application to Open Shift PaaS Cloud Platform**
* Clone the Git Project



* Login to Open Shift account using your Google / GitHub - <https://manage.openshift.com/>
* Then Open Web Console. Change **Administrator** to **Developer**, Click **Add**, Click **from Catalog**,

Select ALL in **Type**, then Expand **Languages** and Select **Python**, then Select **Python**

* Click **Create Application**, **Builder Image Version** Must be >3.6, then **Git Report URL** is <https://github.com/akashjeez/PyFlaskOpenShift.git>, then Application Name is **PyFlaskOpenShift** and Name is **py-flask-openshift** and Click **Create**.
* Click **Topology** from Left Navigator, click **py-flask-openshift**, Once Build is Completed, you can hit the Route’s [URL](http://py-flask-openshift-python-flask-web.apps.us-east-1.starter.openshift-online.com).
* GitHub Code Repository: <https://github.com/akashjeez/pyflaskopenshift/>