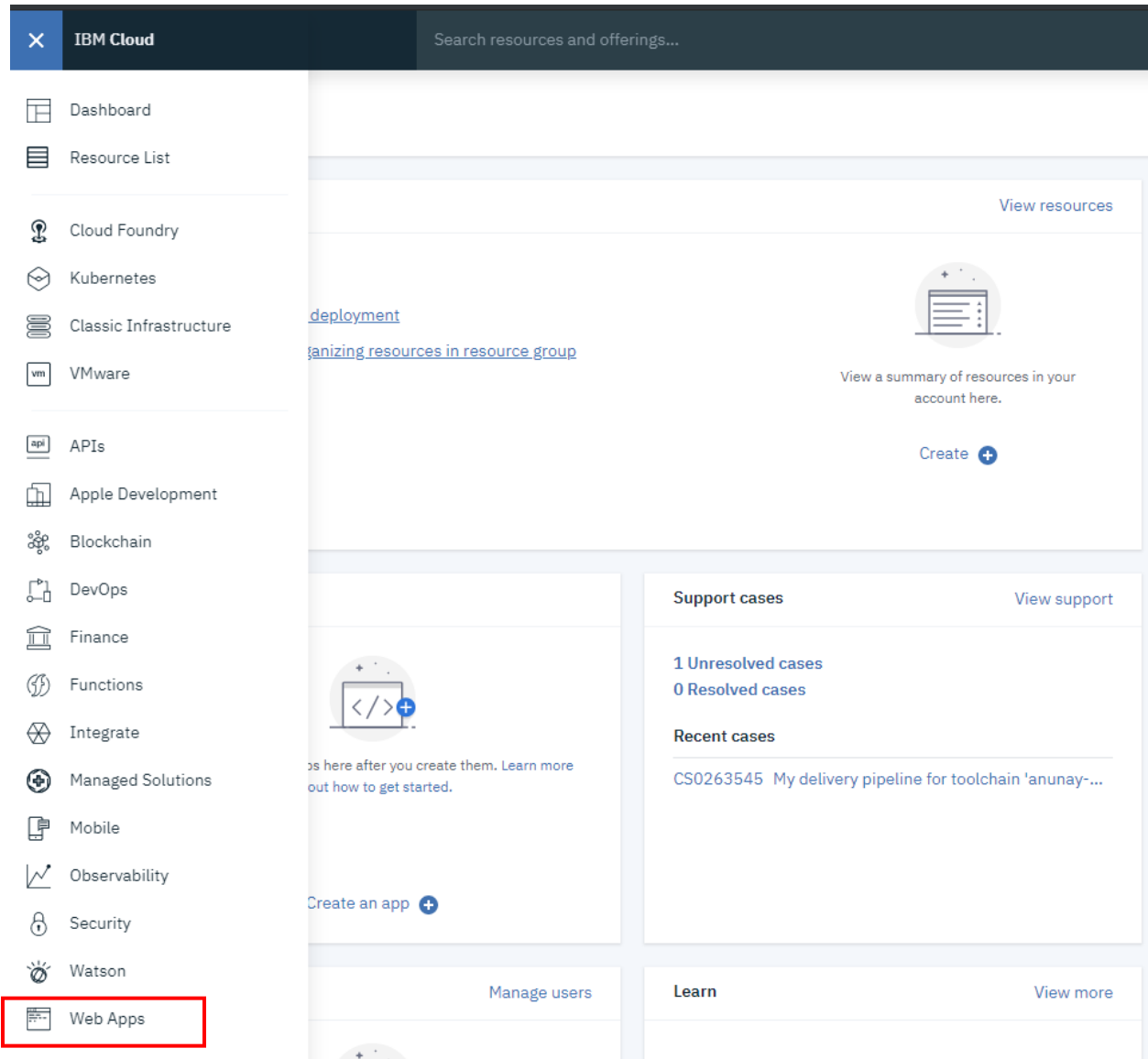


Create IBM Cloud Web App using Watson NLU

Step 1

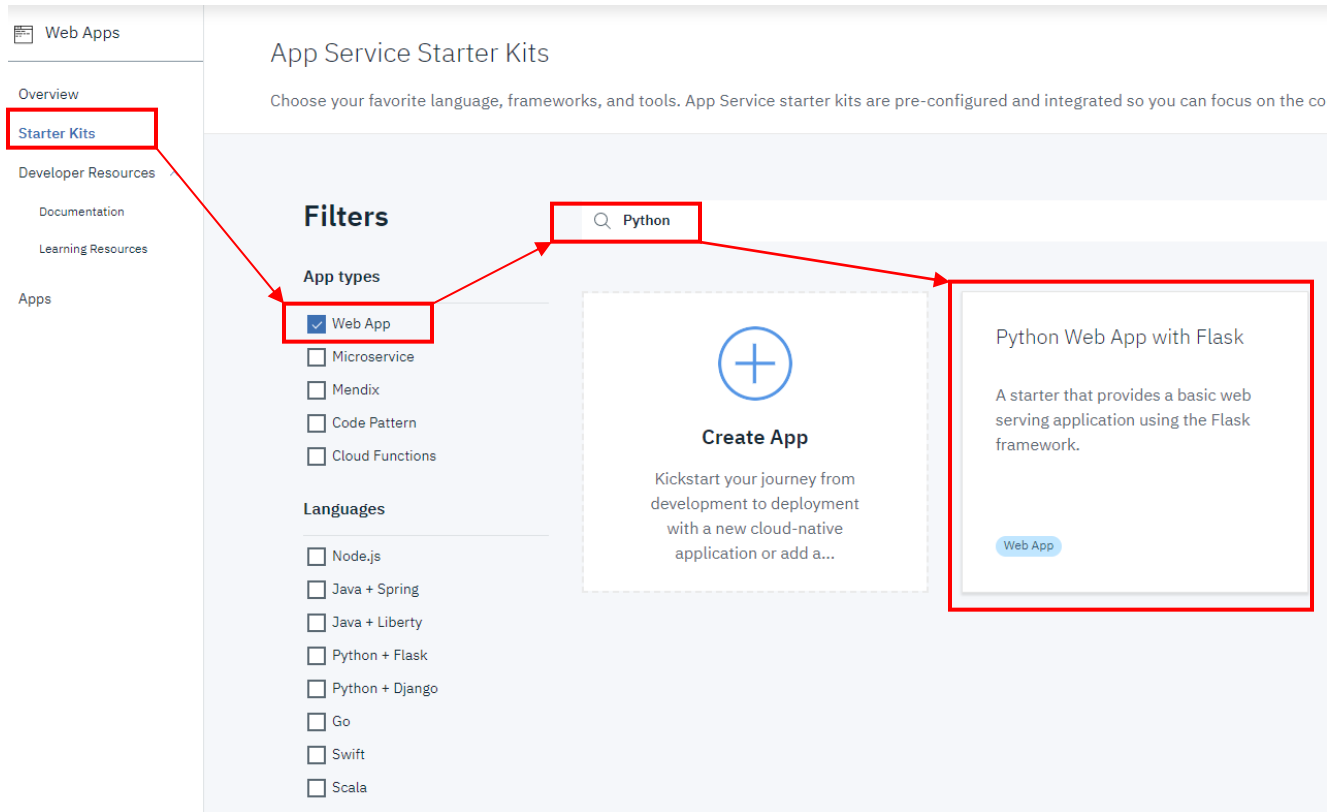
Login to <https://cloud.ibm.com/> and from side bar menu choose Web Apps



Create IBM Cloud Web App using Watson NLU

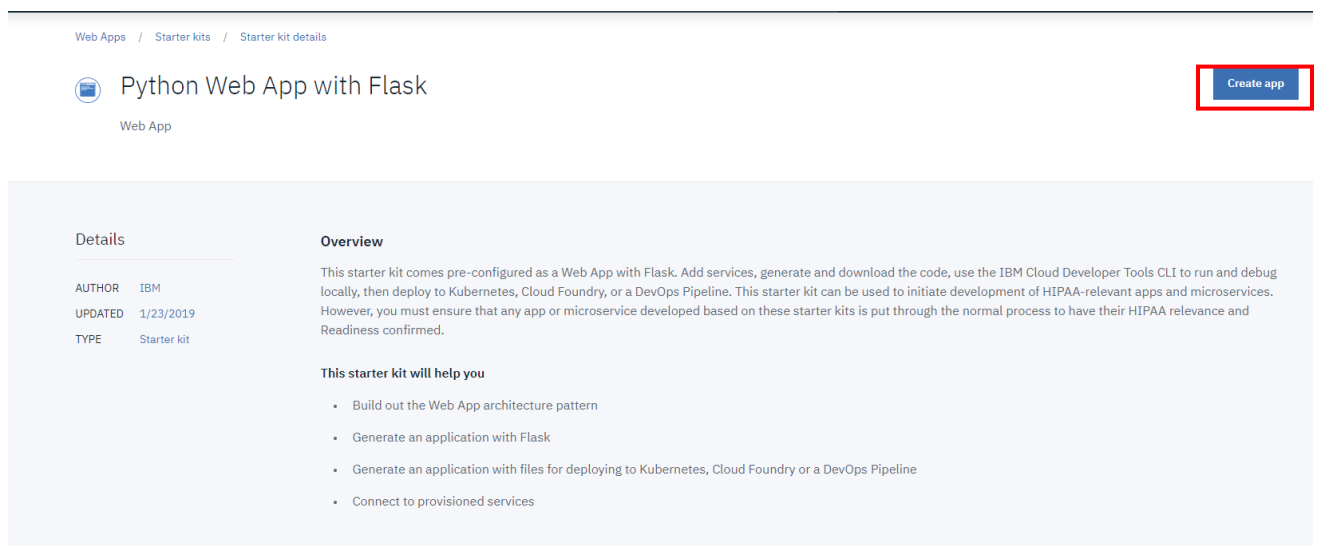
Step 2

Click on 'Starter Kits' and filter by 'Web App' with 'Python' and select 'Python Web App with Flask'



Step 3

Click on the 'Create App' button on the top right of the starter kit details page



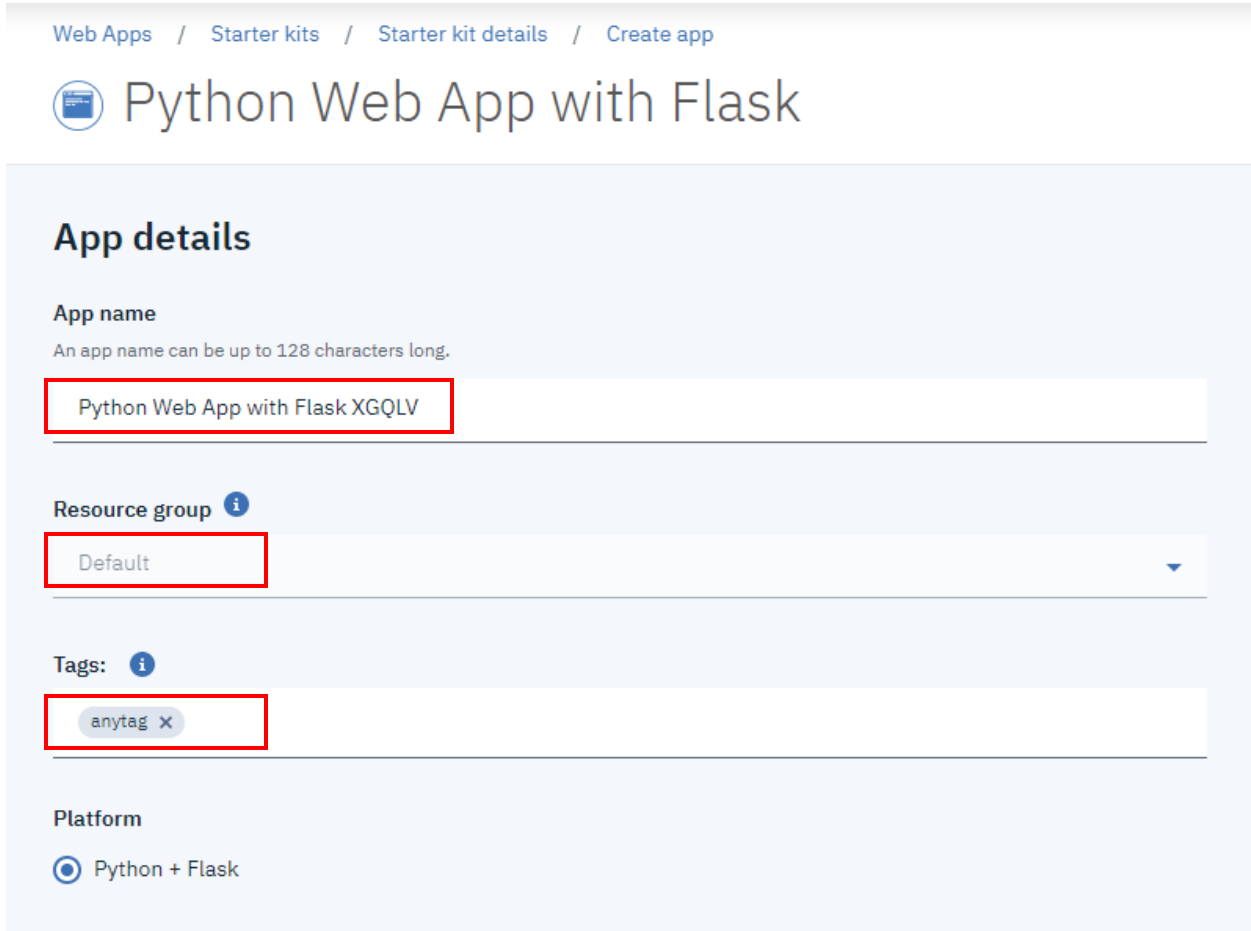
Step 4

Fill in the required details as below and click on 'Create' on top right of the page.

App name – You can keep the default name or chose a new one

Resource group – Leave default

Tags – Choose any tag



The screenshot shows the 'Create app' page for 'Python Web App with Flask'. The breadcrumb navigation at the top is 'Web Apps / Starter kits / Starter kit details / Create app'. The page title is 'Python Web App with Flask'. The 'App details' section contains the following fields:

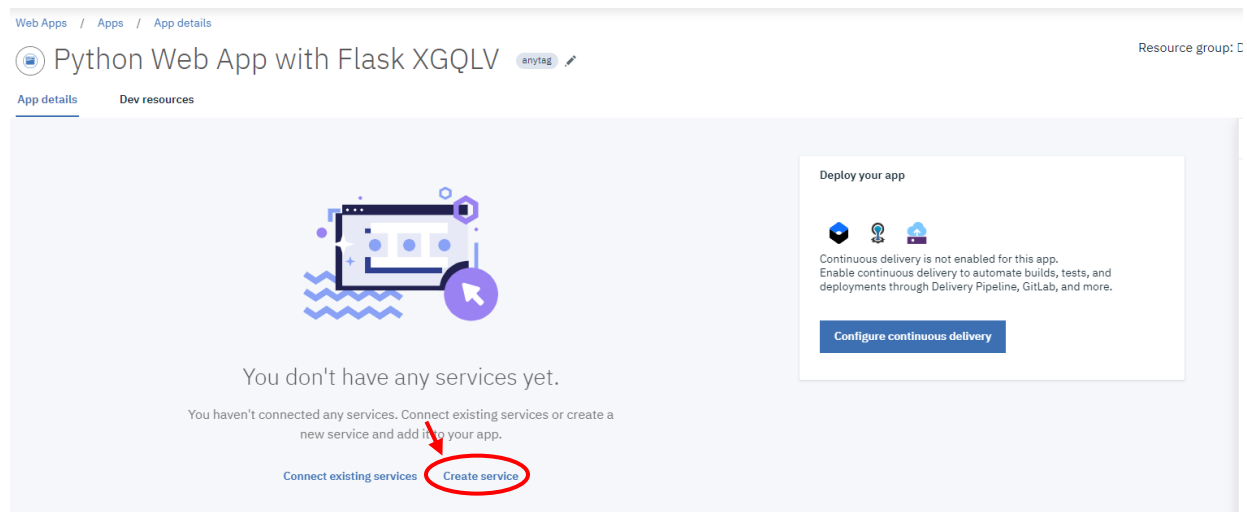
- App name:** A text input field containing 'Python Web App with Flask XGQLV'. Below the field is a note: 'An app name can be up to 128 characters long.'
- Resource group:** A dropdown menu showing 'Default'.
- Tags:** A field containing a tag 'anytag' with a close button 'x'.
- Platform:** A radio button selection with 'Python + Flask' selected.

Your next view will be the App details page.

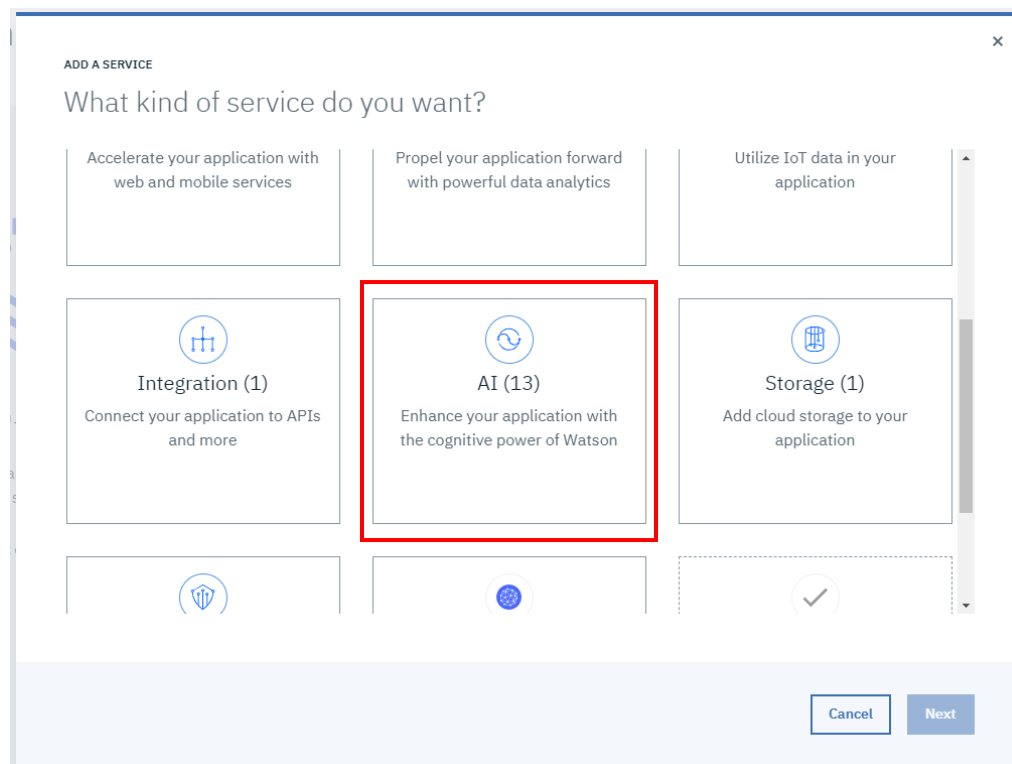
Create IBM Cloud Web App using Watson NLU

Step 5

On the App details page, click on 'Create Service' button to create the Watson Natural Language Understanding Service for your application.

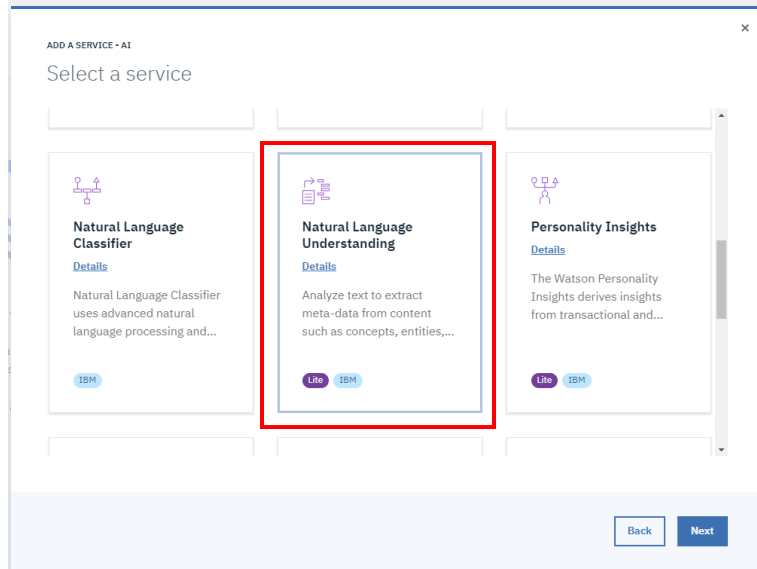


Scroll down and select AI (13) box on the displayed pop-up and click Next.

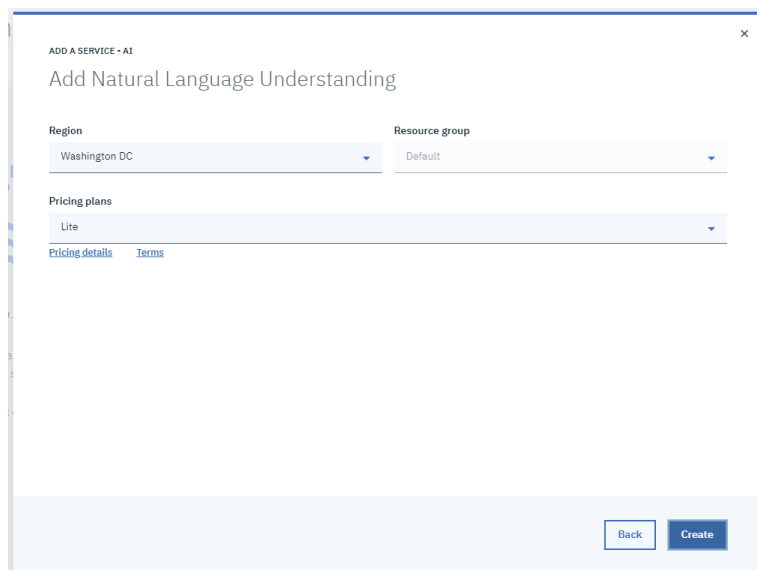


Step 5 ... continued

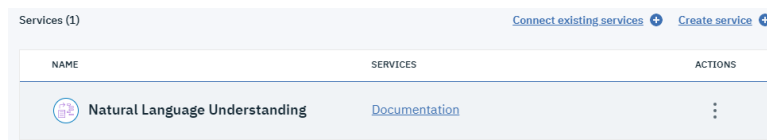
Scroll down and select 'Natural Language Understanding' and click Next.



Leave all the default values and click on 'Create' button



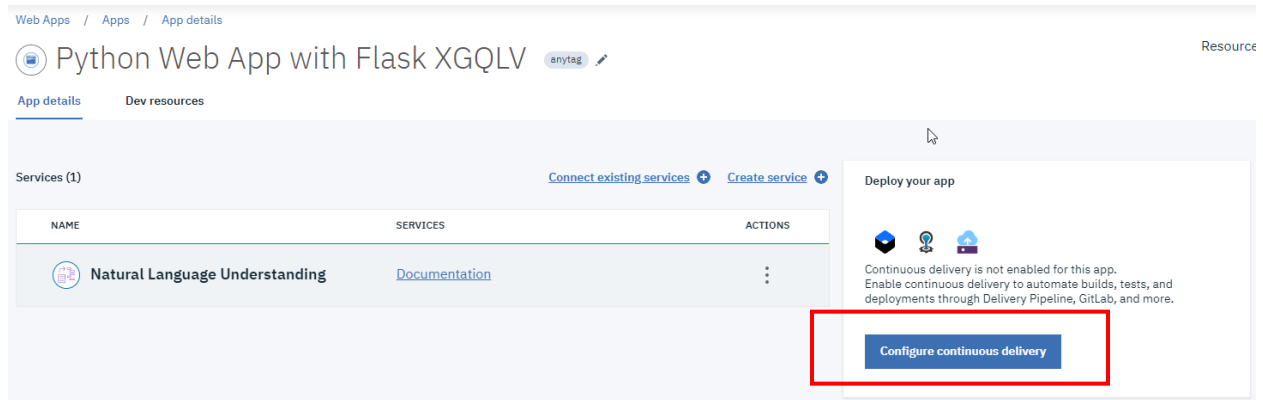
You should now see the service created on the App details page



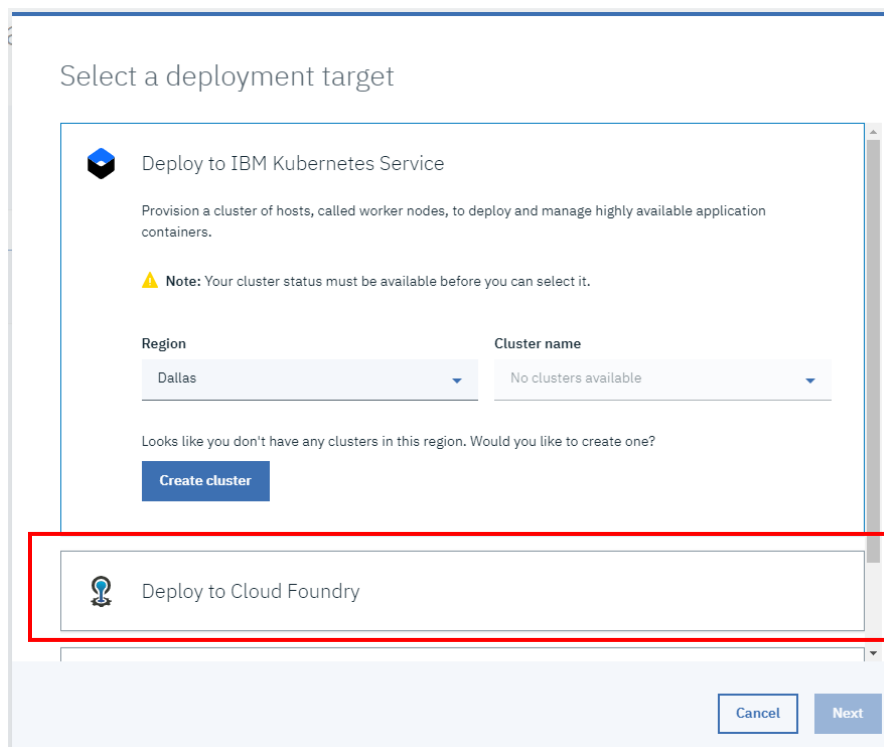
Create IBM Cloud Web App using Watson NLU

Step 6

Now, click on the 'Configure continuous delivery' button.



Select 'Deploy to Cloud Foundry'



DO NOT CLICK 'Next' button yet.

Step 6 ... continued

Once you select 'Deploy to Cloud Foundry', leave all the default values. **Only change the 'Host' field** and change it to any value. The host field value results in a publicly accessible domain name so you can even set it to your own unique name as shown below. Ex – As per below image, this app will be accessible on -

<https://anunay.eu-gb.mybluemix.net>

Select a deployment target

Deploy to Cloud Foundry

Deploy your app without managing underlying infrastructure.

Number of instances

1

Memory allocation per instance

128 MB 256 2000 MB

Select region to deploy in

London

Select an organization

anunay2k@gmail.com

Select a space

dev

Host

anunay

Domain

eu-gb.mybluemix.net

Cancel Next

After clicking 'Next' button, on next page, set a name for your toolchain or leave the default one and click on 'Create' button. **Do not change the default region.** Toolchain takes a few minutes to create.

Configure toolchain

Toolchain name

PythonWebAppwithFlaskXGQLV

Select the region and resource group that you want your toolchain to be created in.

Region

Dallas

Resource group

Default

Back Create

Create IBM Cloud Web App using Watson NLU

Step 7

Navigate to the newly create Toolchain. On the App details page, click on the 'View Toolchain' button.

Web Apps / Apps / App details

Python Web App with Flask XGQLV anytag

App details Dev resources

Warning
IBM Cloud Container Registry has recently changed domains for Docker images. If you have deployed with Kubernetes and your cluster was created on or before February 25th 2019, you must update your container registry credentials and Kubernetes secrets to continue with deployment to Kubernetes environments. See this [blog post](#) for more detail.

Services (1) [Connect existing services](#) [Create service](#)

NAME	SERVICES	ACTIONS
Natural Language Understanding	Documentation	

Continuous delivery [Remove from toolchain](#)

<https://git.ng.bluemix.net/anunay2k/PythonWebAppwithFL...>

Configured
Continuous delivery is enabled for this app.
View the toolchain to see information about this git repository and delivery pipeline which will deploy the code to IBM Cloud.

Tools:

[View toolchain](#)

Toolchain is created with a web-based Eclipse Editor, Git Repo and a delivery pipeline.

Resource list / App details /

PythonWebAppwithFlaskXGQLV

Resource Group: Default Location: Dallas [Add tags](#)

IBM DevOps Insights Lite plan is now free and unlimited with Continuous Delivery
Add DevOps Insights to this toolchain to gate releases to production and review from Continuous Delivery Pipelines and Repos. [Learn more](#)

THINK

THINK	CODE	DELIVER
Issues PythonWebAppwithFL... ✓ Configured	Git PythonWebAppwithFL... ✓ Configured	Delivery Pipeline Python Web App with ... ✓ Configured

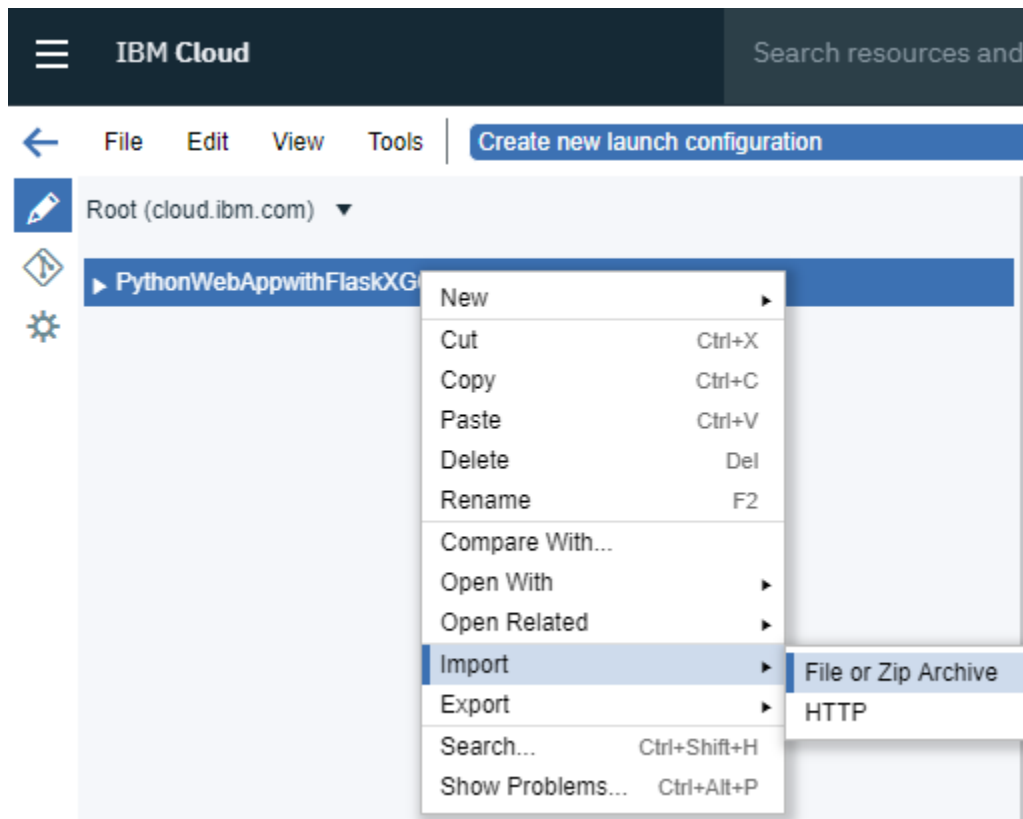
Eclipse Orion Web IDE
✓ Configured

We are now done with the initial Web App setup.

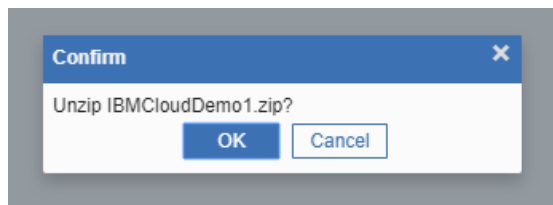
Step 8

Let's code our Web App now. While in the Toolchain, click on the 'Eclipse Orion Web IDE'. This will launch the code editor where we will import the pre- modified code. Download the zip file with modified code here - <https://github.com/kumaranunay123/ibmclouddemo1/raw/master/IBMCloudDemoV1.zip>

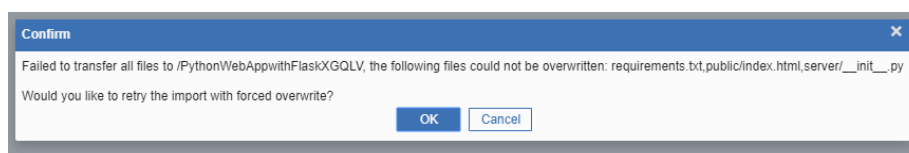
In Eclipse IDE, right click on the project and Select Import



Click OK to unzip the file.



Click OK to force overwrite the existing code.

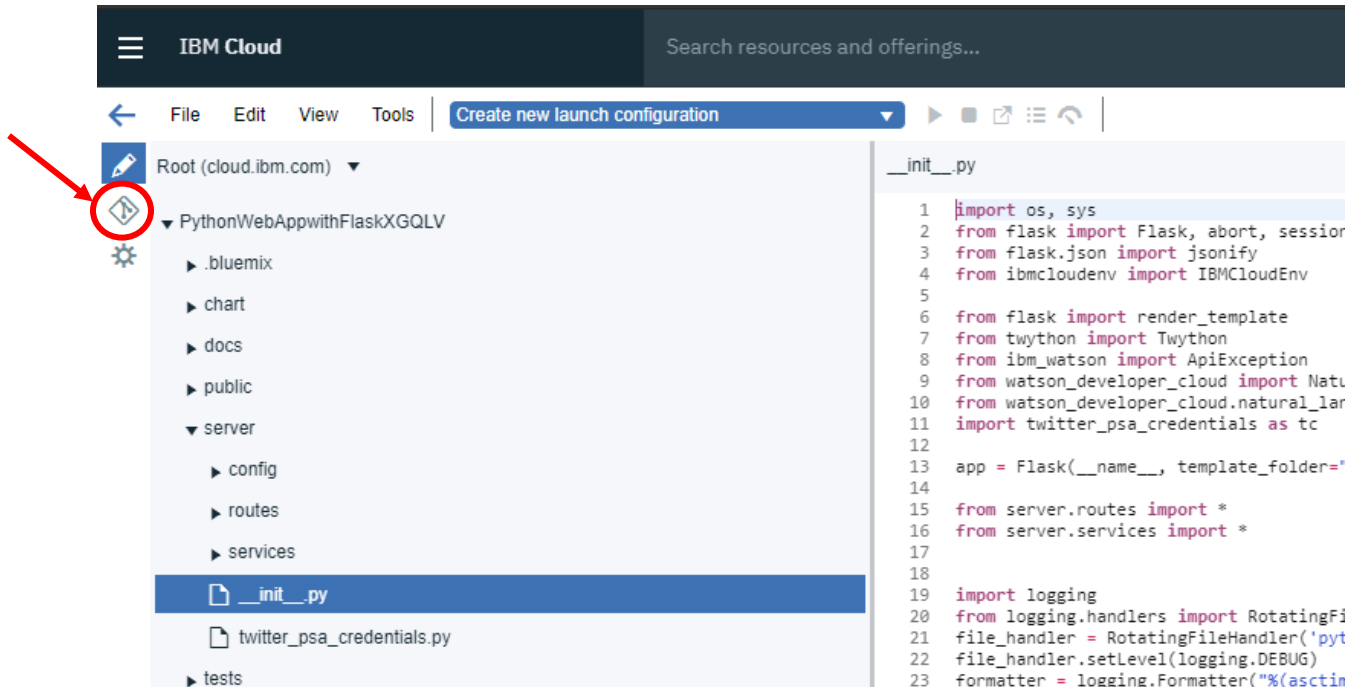


Create IBM Cloud Web App using Watson NLU

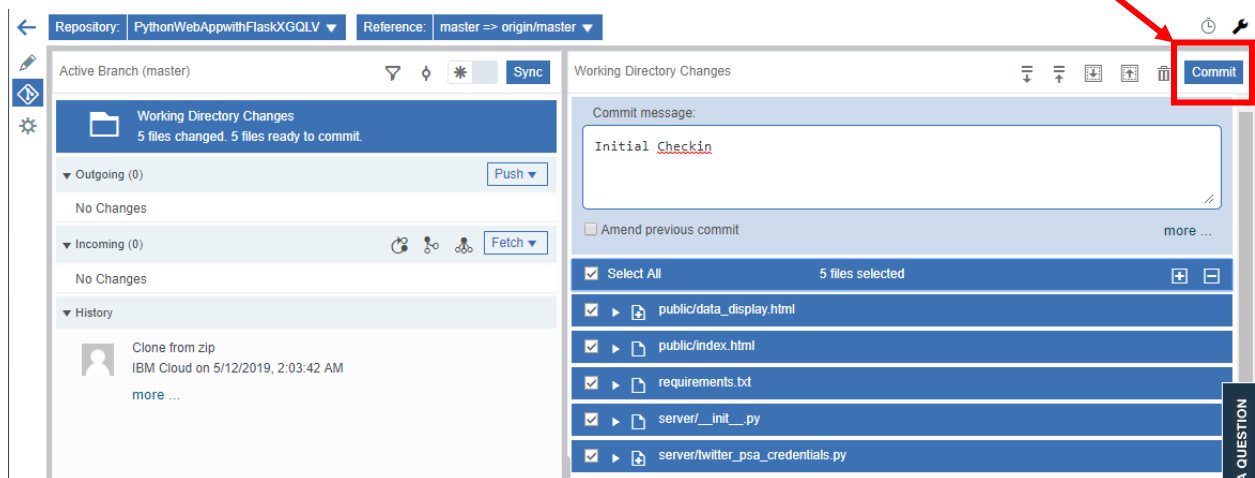
Step 9

The code is now imported. Now we need to commit the new code in Git that we just imported.

Click on the Git button on side menu bar to load the Git view.

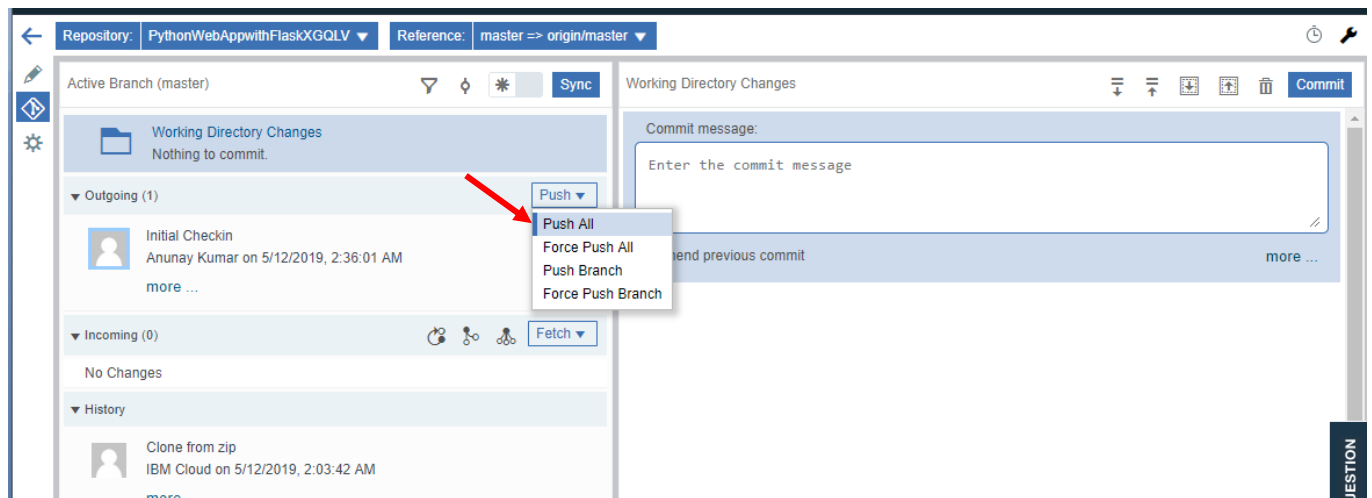


Type in a comment for the commit and click on the 'Commit' button.

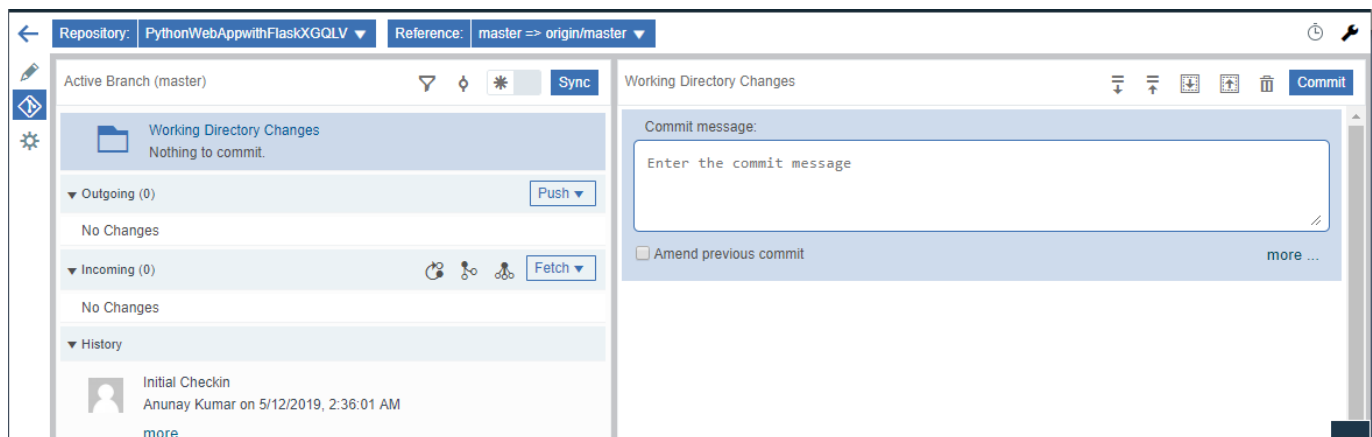


Step 9 ... continued

Now push the changes to the repository. From the 'Outgoing' block choose the 'Push All' option from the dropdown.



Once the changes are pushed successfully, there should be no outgoing changes in queue.



At this point the delivery pipeline gets triggered and redeploys your app with the modified code.

Step 10

Navigate to the Delivery pipeline from the Toolchain.

Go to Toolchains - <https://cloud.ibm.com/devops/toolchains>

Select the correct data center that has your tool chain which is 'Dallas' in most cases.

Click on the Toolchain for your app.

DevOps

Getting Started

Toolchains

Toolchains

RESOURCE GROUP: Default | CLOUD FOUNDRY ORG: -- | LOCATION: Dallas | Filter by resource name... | Create a Toolchain

Toolchains 1/200 Used

Name	Tool Integrations
PythonWebAppwithFlaskXGQLV	

Click on the Delivery Pipeline

Toolchains / PythonWebAppwithFlaskXGQLV

Resource Group: Default | Location: Dallas | Add tags

DELIVER

Issues
PythonWebAppwithFL...
✓ Configured

Git
PythonWebAppwithFL...
✓ Configured

Delivery Pipeline
Python Web App with ...
✓ Configured

Eclipse Orion Web IDE
✓ Configured

Step 10 ... continued

The below image shows the current status of the delivery pipeline where the deployment of the application is in progress.

Toolchains / PythonWebAppwithFlaskXGQLV / Python Web App with Flask XGQLV

Python Web App with Flask XGQLV | Delivery Pipeline

The screenshot displays the IBM Cloud Delivery Pipeline interface for the 'Python Web App with Flask XGQLV' toolchain. It shows three stages: Build Stage, Deploy Stage, and Health Stage. The Build Stage is 'STAGE PASSED' with a job 'Build' that passed 'now'. The Deploy Stage is 'STAGE RUNNING...' with a job 'Deploy Running'. The Health Stage is 'STAGE PASSED' with a job 'Test' that passed '34m ago'. Each stage card includes a 'LAST INPUT' section, a 'JOBS' section with a 'View logs and history' link, and a 'LAST EXECUTION RESULT' section.

Once the pipeline completes, all the stages must be green as below.

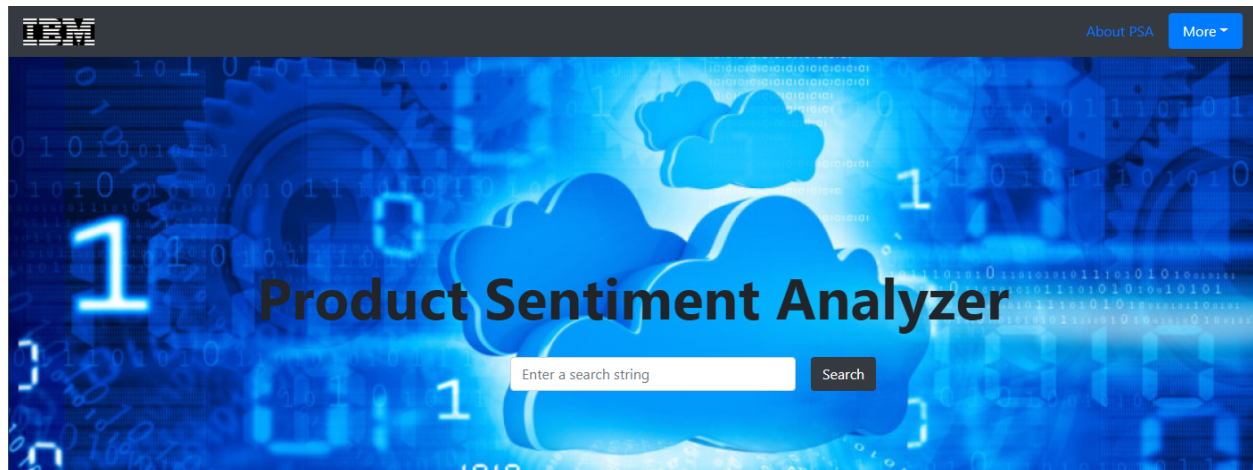
This screenshot shows the IBM Cloud Delivery Pipeline after completion. All three stages—Build Stage, Deploy Stage, and Health Stage—are now 'STAGE PASSED'. The Build Stage job 'Build' passed '11m ago'. The Deploy Stage job 'Deploy' passed '9m ago'. The Health Stage job 'Test' passed '9m ago'. The 'LAST EXECUTION RESULT' for the Deploy Stage now shows 'anunay' with a 'View console' link, indicating a successful deployment.

Step 11

Access your application.

You can access your application on the domain name which we setup on Page 7

Example - <https://anunay.eu-gb.mybluemix.net>



Key in a hash tag like #MothersDay and click on 'Search' button. You should see a sentiment icon depending on the sentiment score returned by Watson NLU for a particular tweet.

Product Sentiment Analyzer			
User	Date	Tweet	Sentiment
louise_gw	Sun May 12 09:57:45 +0000 2019	RT @Imbuto: "A mother, in almost all cultures, is seen as the wellspring of wisdom that never runs dry & a pillar of love that never collap...	😊
MShahhzaib	Sun May 12 09:57:45 +0000 2019	Blessing and love for all mothers in the world.❤️ #MothersDay	😊
iliviamutic	Sun May 12 09:57:43 +0000 2019	Happy Mother's Day #Mothersday #Moms #momlife https://t.co/4CRXuEYaUt	😊
AzhariIrfan_	Sun May 12 09:57:43 +0000 2019	RT @USAmbIndia: Happy #MothersDay from all of us at the U.S. Embassy New Delhi! How would you describe your mother in one word? Here is w...	😊
NoirinCoffey	Sun May 12 09:57:43 +0000 2019	RT @LakotaMan1: My great grandmother Rose hid my mother in the cellar when the Jesuits came to take the little Indian children away. This...	😊
fanty__	Sun May 12 09:57:42 +0000 2019	RT @_myself_kiran: Mom you left me when i needed u the most. i lie on the ground and cry each n everyday. miss u everytime. You always defend...	😊
occupiedbypak	Sun May 12 09:57:42 +0000 2019	RT @BBhuttoZardari: "Your Heaven lies under the feet of your mother." Happy #MothersDay to all the mama's out there. https://t.co/LORyQqs8ZG	😊
No_tya_tyi	Sun May 12 09:57:42 +0000 2019	RT @YolaMkhize: Kwaze kwamnandi ukuthanda lo muntu. To the first of many 🥰❤️ #MothersDay https://t.co/L65AZ2qGyg	😊
MelBa79563320	Sun May 12 09:57:41 +0000 2019	Happy Mother's Day to all mothers in the world. #MothersDay https://t.co/RmLUvykxye	😊

Credits

This application has been originally developed by

- **Ramchandra Bobhate** **IBM Intern | IBM Cloud and Cognitive Software**
- **Prachi Ayare** **IBM Intern | IBM Cloud and Cognitive Software**

Supervised by

- **Pallavi Singh** **Senior Software Engineer | IBM Cloud and Cognitive Software**

Application modified and documented for IBM Cloud by

- **Anunay Kumar** **DevOps Engineer | IBM Cloud and Cognitive Software**