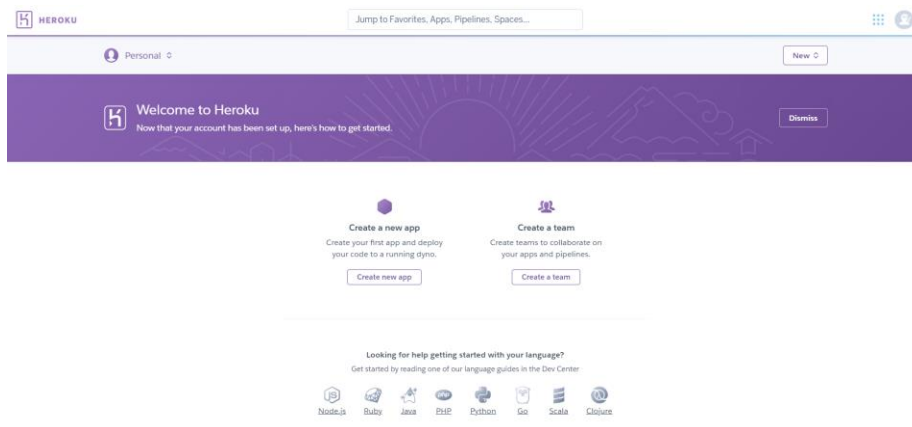


Heroku Deployment

Step 1: Create an account in Heroku.

Once you create and done your verification through mail , then main page looks like this



Step 2: Install the Heroku CLI

Download and install

- ⓘ The Heroku CLI requires **Git**, the popular version control system. If you don't already have Git installed, complete the following before installing the CLI:
- [Git installation](#)
 - [First-time Git setup](#)

macOS

```
$ brew tap heroku/brew && brew install heroku
```

Windows

Download the appropriate installer for your Windows installation:

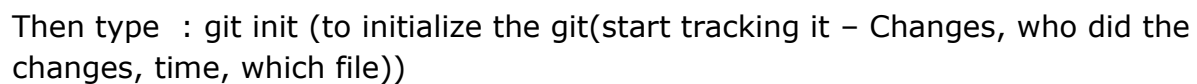
[64-bit installer](#)

[32-bit installer](#)

Step3 : Install the Git



Navigate to your project folder --- >right click ---- > choose Git Bash Here



Git folder is created in our project folder

.git	12/15/2020 11:07 AM	File folder	
__pycache__	10/10/2020 1:57 PM	File folder	
com_in_inuron_ai_utils	3/20/2020 8:43 PM	File folder	
static	12/18/2019 4:27 PM	File folder	
templates	3/21/2020 12:19 AM	File folder	
app	10/10/2020 7:18 PM	JetBrains PyCharm ...	1 KB
cnn	3/20/2020 6:52 PM	JetBrains PyCharm ...	3 KB
HowToRun	10/11/2020 12:01 AM	Text Document	1 KB
model.h5	3/20/2020 6:44 PM	H5 File	46,171 KB
predict	3/20/2020 8:59 PM	JetBrains PyCharm ...	1 KB
requirements	10/2/2020 10:03 PM	Text Document	1 KB

Step 6: git add .

Now add all the files which I have in my project folder.

```
MINGW64/c:/Users/Asus-2020/Downloads/DogCatClassifier
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git init
Reinitialized existing Git repository in C:/Users/Asus-2020/Downloads/DogCatClassifier/.git/

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git add .
warning: LF will be replaced by CRLF in HowToRun.txt.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in app.py.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in cnn.py.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in predict.py.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in requirements.txt.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in static/css/style.css.
The file will have its original line endings in your working directory
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ |
```

Step 7: git status

```
MINGW64/c:/Users/Asus-2020/Downloads/DogCatClassifier
$ git add .
warning: LF will be replaced by CRLF in HowToRun.txt.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in app.py.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in cnn.py.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in predict.py.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in requirements.txt.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in static/css/style.css.
The file will have its original line endings in your working directory
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   HowToRun.txt

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ |
```

Step 8: git commit -m "initial commit"

If you use the git bash for the very first time then it will ask your username and mail once you done then try to commit again and you will the message like below.

```
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git config --global user.name "Amarnath"

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git config --global user.email amarnathdhinakar27@gmail.com

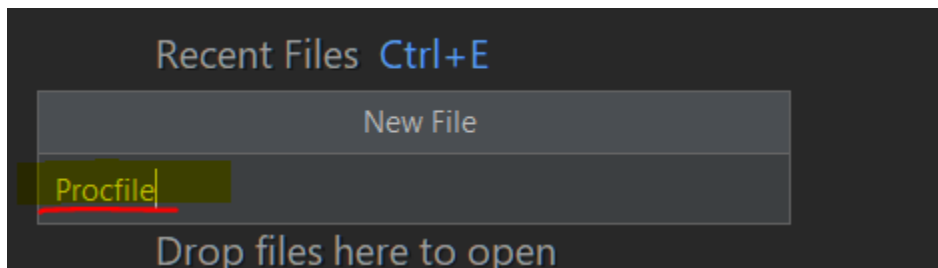
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git commit -m "initial commit"
[master 68df4da] initial commit
1 file changed, 6 insertions(+)
```

If I run the git status command again and it says that nothing to commit.

```
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git status
On branch master
nothing to commit, working tree clean
```

Step 9:

We need to add one file inside in our project folder and name it as Procfile without any extension.



And add the below line into the procfile

```
web: gunicorn app:app
```

Here web is basically a web server and gunicorn is a kind of server and which will used on a cloud side. app:app (app is the file name : app is the flask object name)

Then go to git bash, check the git status and do the git add and commit for the changes that we had done in the project folder.

```
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git add .

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   .idea/.gitignore
    new file:   Procfile

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git commit -m "initial commit"
[master bfecd7] initial commit
2 files changed, 3 insertions(+)
create mode 100644 .idea/.gitignore
create mode 100644 Procfile
```

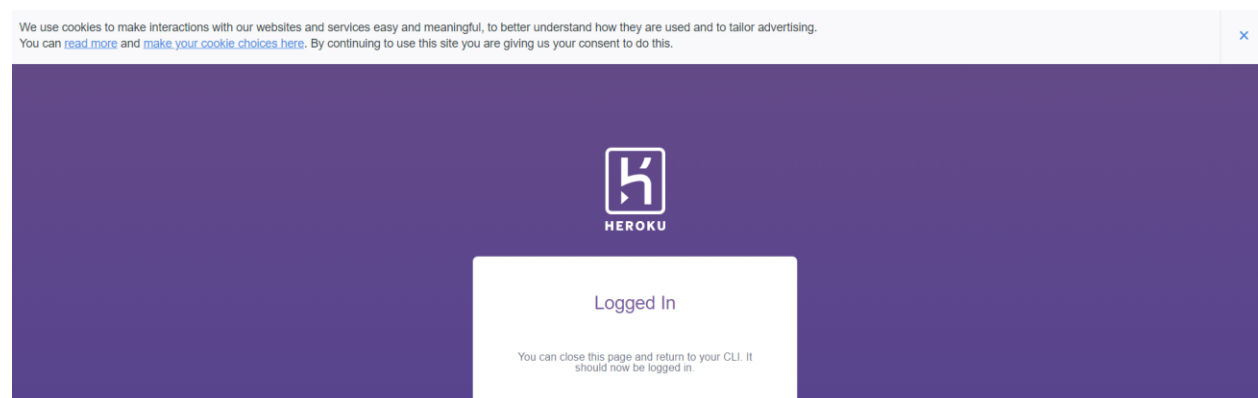
Step 10:

Open the Git bash and type Heroku login

```
MINGW64:/c/Users/Asus-2020/Downloads/DogCatClassifier

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ heroku login
```

Then it ask us to hit any key to open the browser to login to Heroku account and once logged in the it will show the message like below



```

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ heroku login
  » Warning: Our terms of service have changed:
  » https://dashboard.heroku.com/terms-of-service
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/cli/browser/07524ac7-3ccd-41
07-8460-a148e10e89d9?requestor=SFMyNTY.g2gDbQAAAA0xMDMuMTAxLjE3Mi43bgYAMMinZXYBY
gABUYA.IOeuCo6RDwV8IH6r4rxKMTX26p-7BynvPuv4UQ_tEw8
heroku: Waiting for login...
Logging in... done
Logged in as amarnathdhinakar27@gmail.com

```

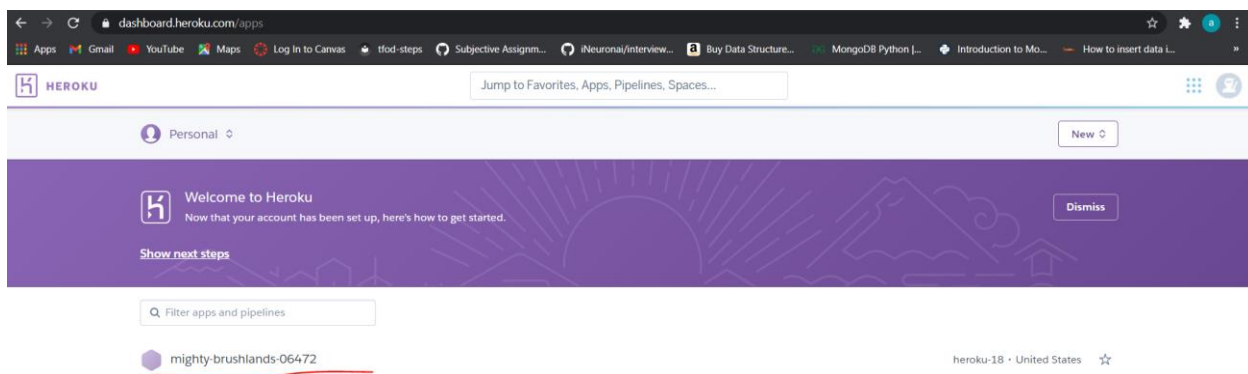
Step 11:

Type heroku create in git bash to create one blank Heroku on a Heroku cloud

```

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ heroku create
Creating app... done, mighty-brushlands-06472
https://mighty-brushlands-06472.herokuapp.com/ | https://git.heroku.com/mighty-brushlands-06472.git

```



The screenshot shows the Heroku dashboard in a web browser. The address bar displays 'dashboard.heroku.com/apps'. The page header includes the Heroku logo and a search bar. Below the header, there's a 'Personal' section with a 'New' button. A purple banner welcomes the user to Heroku, stating 'Now that your account has been set up, here's how to get started.' Below this, there's a search bar for 'Filter apps and pipelines'. At the bottom, the app 'mighty-brushlands-06472' is listed, with a red underline under its name. The status 'heroku-18 · United States' is visible next to it.

Step 12:

Type : git remote -v

It gives me the url for push something to the cloud

```

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git remote -v
heroku https://git.heroku.com/mighty-brushlands-06472.git (fetch)
heroku https://git.heroku.com/mighty-brushlands-06472.git (push)
origin https://dogcattd1.scm.azurewebsites.net:443/dogcattd1.git (fetch)
origin https://dogcattd1.scm.azurewebsites.net:443/dogcattd1.git (push)

```

Step 13:

Type : git push heroku master

Whenever we try to push or fetch some data's in the cloud instead of using the whole url I can type just Heroku push or fetch as here. Here Heroku is the alias for this whole url.

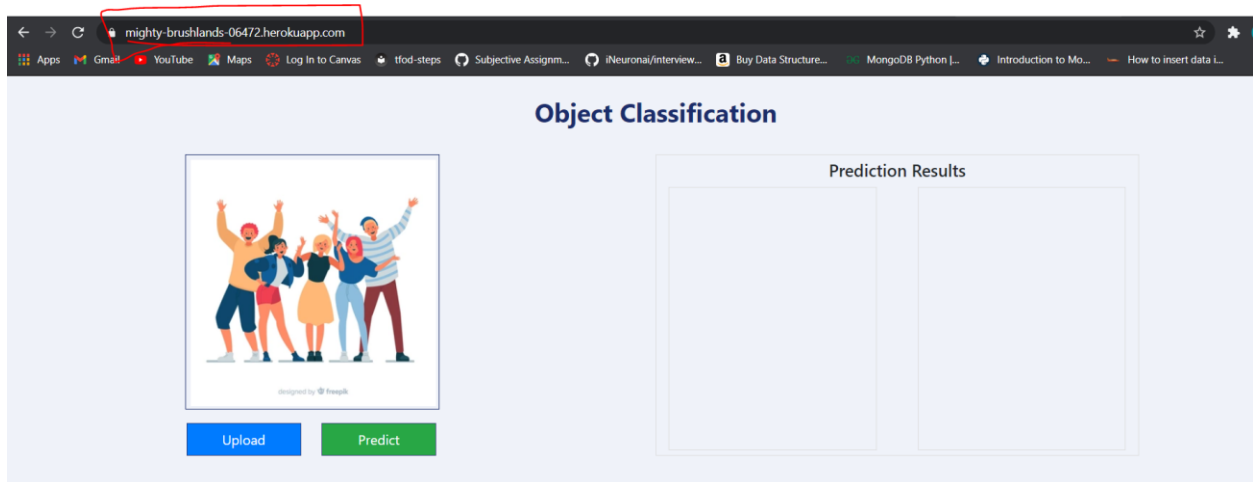
And it will install the necessary packages in the cloud app.

```
Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$ git push heroku master
Enumerating objects: 29, done.
Counting objects: 100% (29/29), done.
Delta compression using up to 8 threads
Compressing objects: 100% (23/23), done.
Writing objects: 100% (29/29), 29.89 MiB | 1.34 MiB/s, done.
Total 29 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote: Building source:
remote:
remote: -----> Python app detected
remote: -----> Installing python-3.6.12
remote: -----> Installing pip 20.1.1, setuptools 47.1.1 and wheel 0.34.2
remote: -----> Installing SQLite3
remote: Collecting PyYAML==5.3.1
remote:   Downloading PyYAML-5.3.1.tar.gz (269 kb)
remote: Collecting scipy==1.4.1
remote:   Downloading scipy-1.4.1-cp36-cp36m-manylinux1_x86_64.whl (26.1 MB)
remote: Collecting six==1.14.0
remote:   Downloading six-1.14.0-py2.py3-none-any.whl (10 kb)
remote: Collecting tensorflow==1.14.0
remote:   Downloading tensorflow-1.14.0-cp36-cp36m-manylinux1_x86_64.whl (109.2 MB)
remote: Collecting tensorflow-estimator==1.14.0
remote:   Downloading tensorflow-estimator-1.14.0-py2.py3-none-any.whl (488 kb)
remote: Collecting termcolor==1.1.0
remote:   Downloading termcolor-1.1.0.tar.gz (3.9 kb)
remote: Collecting Werkzeug==1.0.0
remote:   Downloading Werkzeug-1.0.0-py2.py3-none-any.whl (298 kb)
remote: Collecting wrapt==1.12.1
remote:   Downloading wrapt-1.12.1.tar.gz (27 kb)
remote: Building wheels for collected packages: absl-py, PyYAML, termcolor, wrapt
remote:   Building wheel for absl-py (setup.py): started
remote:   Building wheel for absl-py (setup.py): finished with status 'done'
remote:   Created wheel for absl-py: filename=absl_py-0.9.0-py3-none-any.whl size=121931 sha256=a7c0d272ae2b623763445646c4463b9eea5713812a408ca07da096bf8e497324
remote:   Stored in directory: /tmp/pip-ephem-wheel-cache-caachnyc/wheels/c3/af/84/3962a6a7b4ab336e931b787dcfb758cf94548bb1771e0679
remote:   Building wheel for PyYAML (setup.py): started
remote:   Building wheel for PyYAML (setup.py): finished with status 'done'
remote:   Created wheel for PyYAML: filename=PyYAML-5.3.1-cp36-cp36m-linux_x86_64.whl size=402176 sha256=2ec56a34d76cabice5431f6c5ea829698d8e87ee62dcd71997c5f1403f0975
remote:   Stored in directory: /tmp/pip-ephem-wheel-cache-caachnyc/wheels/e5/9d/ad/2ee53cf762cbaff8afe1487eeef78ea3f260b7e6232a80fc
remote:   Building wheel for termcolor (setup.py): started
remote:   Building wheel for termcolor (setup.py): finished with status 'done'
remote:   Created wheel for termcolor: filename=termcolor-1.1.0-py3-none-any.whl size=4830 sha256=dcfc0d8d391527cc6acdca11c7157001af5436a63b59afa8ad13359a46174bc
remote:   Stored in directory: /tmp/pip-ephem-wheel-cache-caachnyc/wheels/93/2a/eb/e58dbcb963549ee4f065ff80a59f274cc7210b6eab962acd
remote:   Building wheel for wrapt (setup.py): started
remote:   Building wheel for wrapt (setup.py): finished with status 'done'
remote:   Created wheel for wrapt: filename=wrapt-1.12.1-cp36-cp36m-linux_x86_64.whl size=69731 sha256=2e619371f7afa0f53e72a2ba56c27bb7e9085286d97fefad7d0b90dec3b8a259
remote:   Stored in directory: /tmp/pip-ephem-wheel-cache-caachnyc/wheels/32/42/7f/23cae9ff6ef66798d00cd659088e57d8ba01566f6c60db63
remote: Successfully built absl-py PyYAML termcolor wrapt
remote: Installing collected packages: six, absl-py, astor, certifi, click, MarkupSafe, Jinja2, itsdangerous, Werkzeug, Flask, Flask-Cors, gast, google-pasta, grpcio, gunicorn, numpy, h5py, scipy, PyYAML
remote:   Successfully installed Flask-1.1.1 Flask-Cors-3.0.8 Jinja2-2.11.1 Keras-2.3.1 Keras-Applications-1.0.8 Keras-Preprocessing-1.1.0 Markdown-3.2.1 MarkupSafe-1.1.1 Pillow-7.0.0 PyYAML-5.3.1 Werkzeug-1.0.0 absl-py-0.9.0 astor-0.8.1 certifi-2019.11.28 click-7.1.1 gast-0.3.3 google-pasta-0.2.0 grpcio-1.27.2 gunicorn-20.0.4 h5py-2.10.0 itsdangerous-1.1.0 numpy-1.18.2 protobuf-3.11.3 scipy-1.4.1 six-1.14.0 tensorflow-1.14.0 tensorflow-estimator-1.14.0 termcolor-1.1.0 wrapt-1.12.1
remote: -----> Discovering process types
remote:   Profile declares types -> web
remote:
remote: -----> Compressing...
remote:   Done: 245.7M
remote: -----> Launching...
remote:   Released v3
remote:   https://mighty-brushlands-06472.herokuapp.com/ deployed to Heroku
remote: Verifying deploy... done.
To https://git.heroku.com/mighty-brushlands-06472.git
 * [new branch] master -> master

Asus-2020@LAPTOP-3VKRL6IT MINGW64 ~/Downloads/DogCatClassifier (master)
$
```

```
remote:
remote: ----> Compressing...
remote: Done: 245.7M
remote: ----> Launching...
remote: Released v3
remote: https://mighty-brushlands-06472.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/mighty-brushlands-06472.git
* [new branch]      master -> master
```

Finally we will be able to see this deployed our application url.



<https://mighty-brushlands-06472.herokuapp.com/>