

# Steps to run Flask App in GAE Devappserver

1. Setup your flask app folder with
  - app.py
  - templates/index.html
  - static/css/main.css
2. Add the following code in app.py if not present

```
if __name__ == "__main__":  
    app.run()
```

3. Rename app.py to main.py
4. Create initialize\_gae.py inside the folder with the following code

```
from google.appengine.ext.webapp.util import run_wsgi_app  
from main import app  
  
run_wsgi_app(app)
```

5. Create app.yaml with the following code

```
application: flaskapp  
version: 1  
runtime: python  
api_version: 1  
  
handlers:  
- url: .*  
  script: initialize_gae.py
```

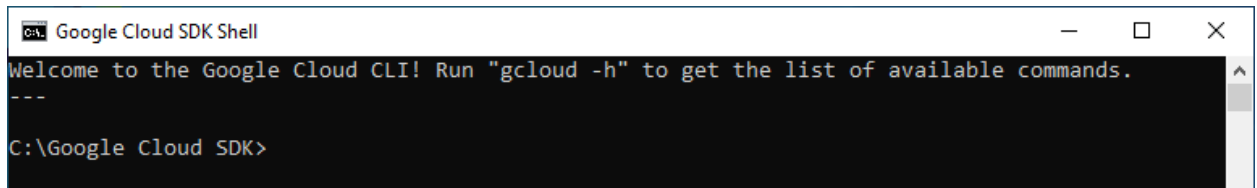
6. Final folder structure would be like this

- main.py
- initialize\_gae.py
- app.yaml
- templates/
  - index.html
- static/
  - css/
    - main.css

7. Paste the following folders inside your flask app folder (*These folder will be available in the **\\cdcserver** as a zip file*)

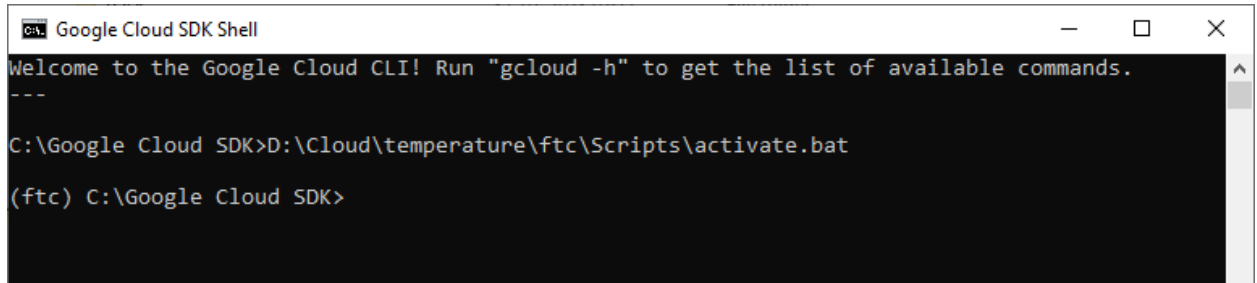
This PC > HDD (D:) > Cloud > temperature				
	Name	Date modified	Type	Size
ess	flask	31-01-2023 07:11	File folder	
	jinja2	31-01-2023 07:13	File folder	
ads	simplejson	31-01-2023 07:12	File folder	
nts	static	31-10-2022 18:11	File folder	
	templates	30-01-2023 16:36	File folder	
	werkzeug	31-01-2023 07:12	File folder	
b_CS871	app	31-01-2023 07:09	Yaml Source File	1 KB
	initialize_gae	31-01-2023 07:09	Python Source File	1 KB
1	main	31-01-2023 07:09	Python Source File	1 KB
der	main	31-01-2023 07:10	Compiled Python ...	2 KB
ture	ftc	30-01-2023 13:09	File folder	

## 8. Start the Google Cloud SDK Shell

A screenshot of a Windows command prompt window titled "Google Cloud SDK Shell". The window has standard Windows window controls (minimize, maximize, close) in the top right corner. The text inside the window reads: "Welcome to the Google Cloud CLI! Run 'gcloud -h' to get the list of available commands." followed by three dashes "---" and a prompt "C:\Google Cloud SDK>".

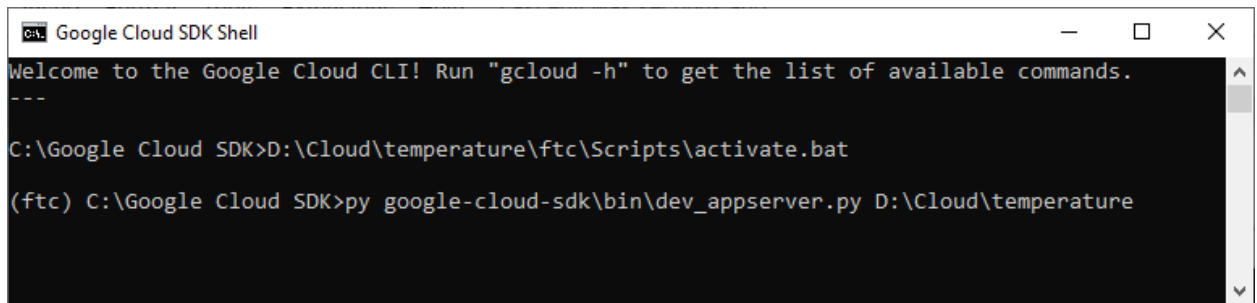
```
Google Cloud SDK Shell
Welcome to the Google Cloud CLI! Run "gcloud -h" to get the list of available commands.
---
C:\Google Cloud SDK>
```

## 9. Activate a virtual env with flask already installed(if not use **pip install flask** to install flask).

A screenshot of a Windows command prompt window titled "Google Cloud SDK Shell". The text inside the window shows the execution of a batch file: "C:\Google Cloud SDK>D:\Cloud\temperature\ftc\Scripts\activate.bat" followed by "(ftc) C:\Google Cloud SDK>".

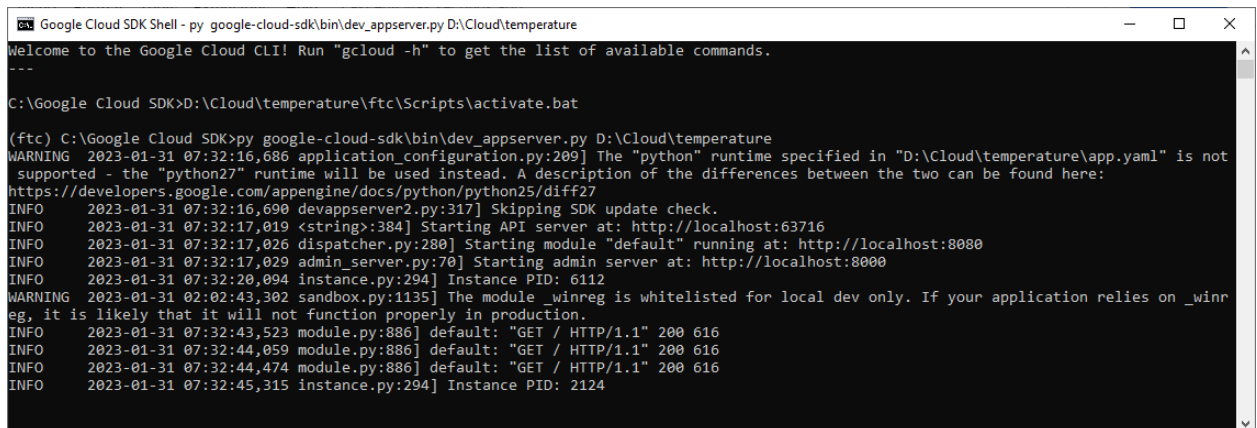
```
Google Cloud SDK Shell
Welcome to the Google Cloud CLI! Run "gcloud -h" to get the list of available commands.
---
C:\Google Cloud SDK>D:\Cloud\temperature\ftc\Scripts\activate.bat
(ftc) C:\Google Cloud SDK>
```

## 10. Run **py google-cloud-sdk\bin\dev-appserver.py** "*path to app folder*"

A screenshot of a Windows command prompt window titled "Google Cloud SDK Shell". The text inside the window shows the command being entered: "C:\Google Cloud SDK>D:\Cloud\temperature\ftc\Scripts\activate.bat" followed by "(ftc) C:\Google Cloud SDK>py google-cloud-sdk\bin\dev\_appserver.py D:\Cloud\temperature".

```
Google Cloud SDK Shell
Welcome to the Google Cloud CLI! Run "gcloud -h" to get the list of available commands.
---
C:\Google Cloud SDK>D:\Cloud\temperature\ftc\Scripts\activate.bat
(ftc) C:\Google Cloud SDK>py google-cloud-sdk\bin\dev_appserver.py D:\Cloud\temperature
```

## 11. The app will be running at **http://localhost:8080/**

A screenshot of a Windows command prompt window titled "Google Cloud SDK Shell - py google-cloud-sdk\bin\dev\_appserver.py D:\Cloud\temperature". The window shows the output of the command, including a warning about the python runtime, several INFO messages about starting the API server, dispatcher, and admin server, and a final INFO message about the instance PID. The text inside the window is as follows:

```
Google Cloud SDK Shell - py google-cloud-sdk\bin\dev_appserver.py D:\Cloud\temperature
Welcome to the Google Cloud CLI! Run "gcloud -h" to get the list of available commands.
---
C:\Google Cloud SDK>D:\Cloud\temperature\ftc\Scripts\activate.bat
(ftc) C:\Google Cloud SDK>py google-cloud-sdk\bin\dev_appserver.py D:\Cloud\temperature
WARNING 2023-01-31 07:32:16,686 application_configuration.py:209] The "python" runtime specified in "D:\Cloud\temperature\app.yaml" is not
supported - the "python27" runtime will be used instead. A description of the differences between the two can be found here:
https://developers.google.com/appengine/docs/python/python25/diff27
INFO 2023-01-31 07:32:16,690 devappserver2.py:317] Skipping SDK update check.
INFO 2023-01-31 07:32:17,019 <string>:384] Starting API server at: http://localhost:63716
INFO 2023-01-31 07:32:17,026 dispatcher.py:280] Starting module "default" running at: http://localhost:8080
INFO 2023-01-31 07:32:17,029 admin_server.py:70] Starting admin server at: http://localhost:8000
INFO 2023-01-31 07:32:20,094 instance.py:294] Instance PID: 6112
WARNING 2023-01-31 02:02:43,302 sandbox.py:1135] The module _winreg is whitelisted for local dev only. If your application relies on _winr
eg, it is likely that it will not function properly in production.
INFO 2023-01-31 07:32:43,523 module.py:886] default: "GET / HTTP/1.1" 200 616
INFO 2023-01-31 07:32:44,059 module.py:886] default: "GET / HTTP/1.1" 200 616
INFO 2023-01-31 07:32:44,474 module.py:886] default: "GET / HTTP/1.1" 200 616
INFO 2023-01-31 07:32:45,315 instance.py:294] Instance PID: 2124
```

