

Minimal Serverless and Flask example

This example shows the use of **Serverless Framework** — a *CloudFormation-based* plugin — with **serverless-wsgi** (for flask proxy routing) and **serverless-python-requirements** plugins, in a Windows environment.

- Install **nodejs** using **choco install nodejs** or any other package manager
- Install **python**
- Install **docker** (only required for local testing or **dockerizePip**)
 - Share your **C:** drive or where you have your python packages in **docker** using **WSL 1.0**
- Run **setup.bat**
- Setup your aws profile credentials in **~/.aws/credentials**
- If you have private repositories add the following to **requirements.txt**:

```
git+{{url_en_github}}@{{branch}}#egg={{nombre_del_paquete_a_instalar}}&subdirectory={{subdirectorio/relativo/tomando/como/base/la/raiz/del/proyecto/hasta/el/setup.py/del/paquete}}
```

- Run **deploy_sls.bat**

Recommended Readings

- [The Official Guide to Serverless Flask](#)
- [Build a Python REST API with Serverless, Lambda, and DynamoDB](#)
- [How to Test Serverless Applications](#)
- [Stages in Serverless Framework](#)
- [Using a private Python package as a dependency](#)
- Plugins
 - [Serverless Python Requirements](#)
 - [Serverless WSGI](#)