Devops Flask

Development

Step 1: Create App

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello():
    return "Hell World"


if __name__ == "__main__":
    app.run(host ='0.0.0.0', port = 5001, debug = True)
```

> name it as demo.py

```
click==8.0.3
Flask==2.0.2
itsdangerous==2.0.1
Jinja2==3.0.2
MarkupSafe==2.0.1
Werkzeug==2.0.2
```

Containerize the app

Step 2: Create Docker file

```
FROM python:alpine3.7
COPY . /app
WORKDIR /app
RUN pip install -r requirements.txt
EXPOSE 5001
ENTRYPOINT [ "python" ]
CMD [ "demo.py" ]
```

Step 3: Write a compose file

```
version: "3.9"
services:
  web:
    build: .
    ports:
      - "8000:5001"
```

Push to GitHub

Step 1: Install git for windows

How to Install Git on Windows {Step-by-Step Tutorial} - PhoenixNAP

Introduction Git is a widely used open-source software tracking application used to track projects across different teams and revision levels. This guide will show you how to install Git on Windows. Prerequisites Administrator

https://phoenixnap.com/kb/how-to-install-git-windows

Step 3: Add the files to staging

```
git add .
```

Step 4: Commit the changes

```
git commit -m "first release"
```

Step 5: Push changes to Github

```
git push -u origin main
```

> use can push with api key or ssh keys

Step up a server for the app deployment

Step 6: Update Server

```
apt update -y && apt upgrade -y
```

Step 7: Install Docker and Docker compose

```
curl https://get.docker.com/ | bash
```

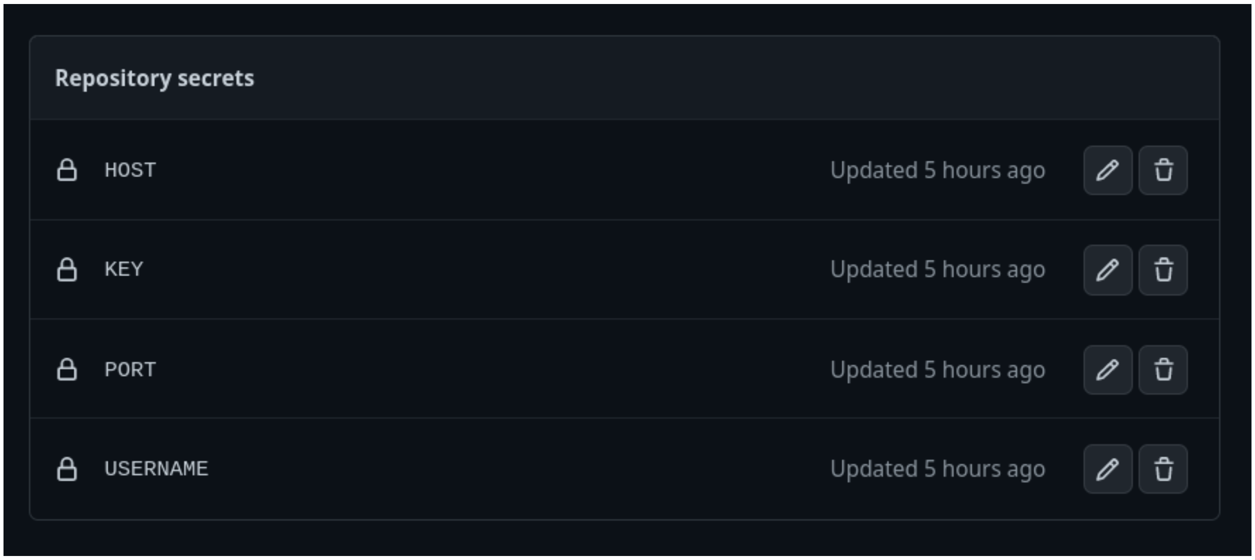
Step 8: Setup GIthub Deployment Key & clone

```
ssh-keygen
git clone repo_name
```

Create GitHub actions

```
name: Docker Image CI

on:
  push:
    branches: [ "main" ]
  pull_request:
    branches: [ "main" ]

jobs:

  build:
    runs-on: ubuntu-latest
    steps:
    - name: Checkout to Code
      uses: actions/checkout@v3

    - name: Deploy to server
      uses: appleboy/ssh-action@v0.1.6
      with:
        host: ${{ secrets.HOST }}
        username: ${{ secrets.USERNAME }}
        key: ${{ secrets.KEY }}
        port: ${{ secrets.PORT }}
        script: |
          cd /root/fla/
          eval `ssh-agent -s`
          ssh-add /root/deploy
          git pull
          docker compose up -d --force-recreate --no-deps --build web

    - name: Docker Login
      uses: docker/login-action@v2
      with:
          registry: ghcr.io
          username: ${{ github.actor }}
          password: ${{ secrets.GITHUB_TOKEN }}

    - name: Build and push
      uses: docker/build-push-action@v3
      with:
          push: true
          tags: ghcr.io/${{ github.repository }}
```

Step 9: Add secret



Repository secrets

HOST	Updated 5 hours ago
KEY	Updated 5 hours ago
PORT	Updated 5 hours ago
USERNAME	Updated 5 hours ago

Add Deep Source to monitor the code

Notes

```
eval `ssh-agent -s`
ssh-add key
```