

部屬應用程式至 **Amazon Web Services**

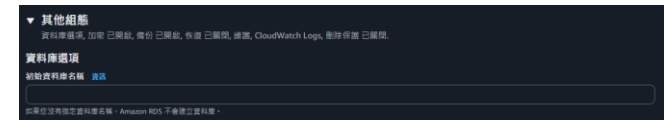
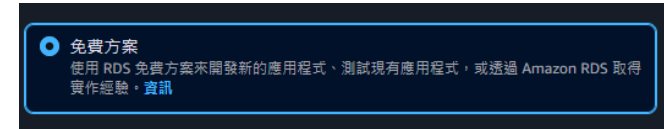
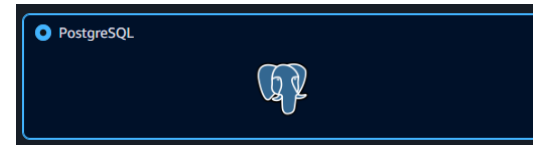
# 工具

- IDE : Visual studio code
- Amazon web services : Aurora and RDS, S3, Lambda, API Gateway
- PYPI :
  - flask, psycopg2-binary, openai, line-bot-sdk, zappa

# 建立關聯式資料庫

- Aurora and RDS

- 選擇：PostgreSQL
- 選擇：範本 - 免費方案
- 選擇：可用性與耐久性 - 1個執行續
- 設定：資料庫名稱設定
  - 個體識別 ID
  - 使用者名稱
  - 密碼
- 選擇：執行個體組態 - t4g.micro
- 設定：其他組態 - 初始資料庫名稱



# 設定本地 AWS 組態

- 下載：對應電腦版本的 AWS CLI
- 設定：AWS config
  - Key
  - Region
  - format

## AWS CLI install and update instructions

For installation instructions, expand the section for your operating system.

▶ Linux

▶ macOS

▶ Windows

```
C:\Users\User>aws configure
AWS Access Key ID [*****R6NA]:
AWS Secret Access Key [*****SOON]:
Default region name [us-east-1]:
Default output format [json]:
```

# 程式

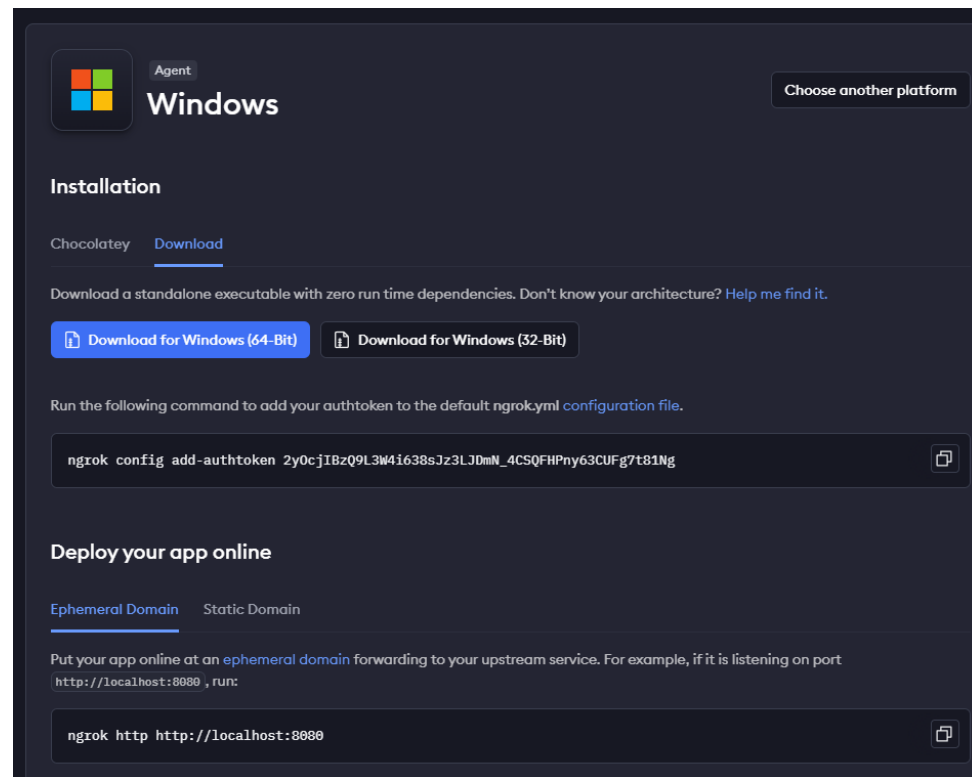
- 使用 Visual studio code
  - 建置環境
    - venv
  - 安裝套件
    - line-bot-sdk
    - flask
    - openai
    - zappa

```
(.venv) C:\Users\User\Desktop\project>pip list
Package            Version
-----
aenum               3.1.16
aiohappyeyeballs   2.6.1
aiohttp             3.12.14
aiosignal           1.4.0
annotated-types     0.7.0
anyio               4.9.0
argcomplete         3.6.2
attrs               25.3.0
blinker             1.9.0
boto3               1.39.4
botocore            1.39.4
certifi             2025.7.9
```

# 本地測試

- 使用 ngrok

- 下載：對應電腦版本
- 設定：config
- 開啟：啟動服務
- 測試：測試程式狀態



Agent Windows Choose another platform

### Installation

Chocolatey Download

Download a standalone executable with zero run time dependencies. Don't know your architecture? [Help me find it.](#)

[Download for Windows \(64-Bit\)](#) [Download for Windows \(32-Bit\)](#)

Run the following command to add your authtoken to the default ngrok.yml configuration file.

```
ngrok config add-authtoken 2y0cjIBzQ9L3W4i638sJz3LJDmN_4CSQFHPny63CUFG7t81Ng
```

### Deploy your app online

Ephemeral Domain Static Domain

Put your app online at an ephemeral domain forwarding to your upstream service. For example, if it is listening on port `http://localhost:8080`, run:

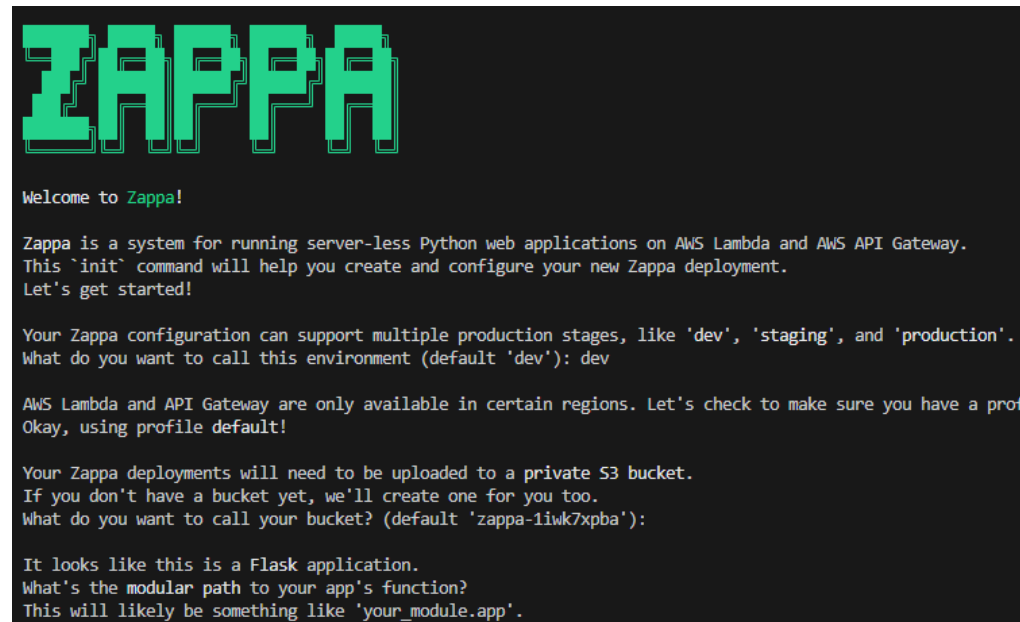
```
ngrok http http://localhost:8080
```

```
ersion 3.23.3
egion Japan (jp)
eb Interface http://127.0.0.1:4040
orwarding https://1610d7a98aad.ngrok-free.app -> http://localhost:5000

Connections      ttl    opn    rtl    rt5    p50    p90
                  0      0      0.00  0.00  0.00  0.00
```

# 部屬

- 使用 zappa
  - 初始化
  - 設定
  - 產生 format



```

ZAPPA

Welcome to Zappa!

Zappa is a system for running server-less Python web applications on AWS Lambda and AWS API Gateway.
This `init` command will help you create and configure your new Zappa deployment.
Let's get started!

Your Zappa configuration can support multiple production stages, like 'dev', 'staging', and 'production'.
What do you want to call this environment (default 'dev'): dev

AWS Lambda and API Gateway are only available in certain regions. Let's check to make sure you have a profile.
Okay, using profile default!

Your Zappa deployments will need to be uploaded to a private S3 bucket.
If you don't have a bucket yet, we'll create one for you too.
What do you want to call your bucket? (default 'zappa-1iwk7xpba'):

It looks like this is a Flask application.
What's the modular path to your app's function?
This will likely be something like 'your_module.app'.

```



```

{
  "dev": {
    "app_function": "main.app",
    "aws_region": "us-east-1",
    "exclude": [
      "boto3",
      "dateutil",
      "botocore",
      "s3transfer",
      "concurrent"
    ],
    "profile_name": "default",
    "project_name": "project",
    "runtime": "python3.12",
    "s3_bucket": "zappa-1iwk7xpba"
  }
}

```

# Linebot 應用

- 串接 openai
  - 設定：不同的角色帶入不同 prompt
  - 互動：讓使用者體驗沉浸式互動

