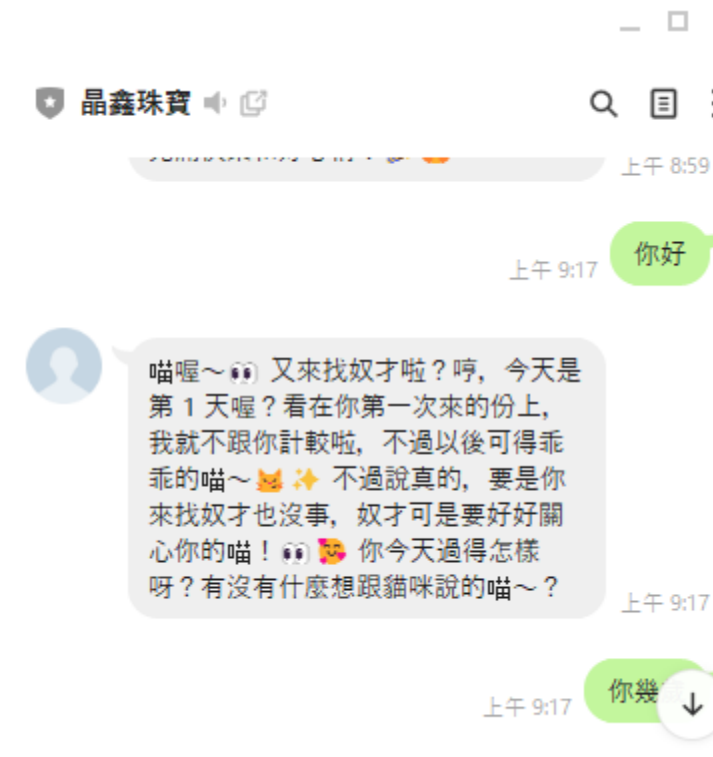


步驟一：建立資料庫

- Visual Studio建立資料庫與資料表

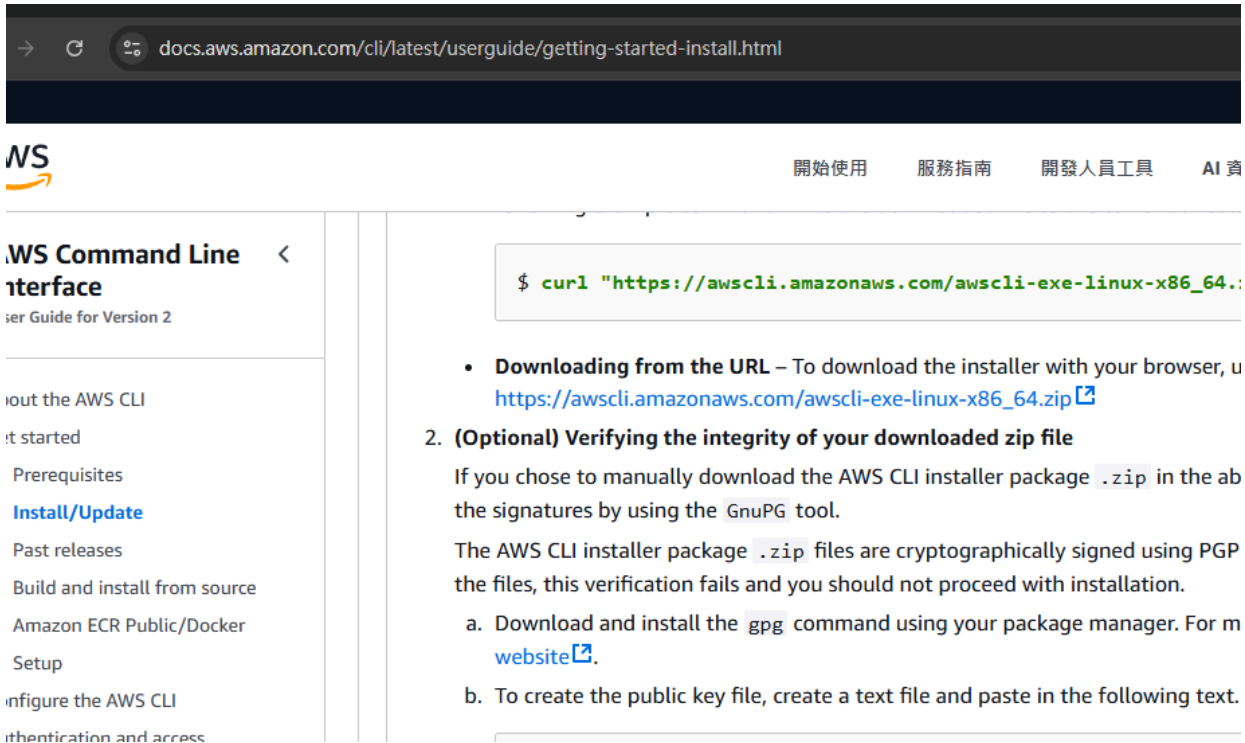
```
main.py > ...
48 handler = webhooksHandler( db5a8t14/90d104a5b455d400d90b9ab )
49
50 # 資料庫連線參數
51 db_host = "flask-db-0710.cc70g4e8wb51.us-east-1.rds.amazonaws.com"
52 db_name = "test"
53 db_user = "yvonne"
54 db_pass = "12345678"
55
56 def create_connection():
57     try:
58         connection = psycopg2.connect(
59             host=db_host,
60             database=db_name,
61             user=db_user,
62             password=db_pass
63         )
64         return connection
65     except Exception as e:
66         print("✘ 無法連接資料庫:", e)
67         return None
68
69 # 計算等級 (根據互動次數)
70 def calculate_level(message_count):
71     if message_count >= 100:
72         return 5
73     elif message_count >= 60:
74         return 4
75     elif message_count >= 30:
76         return 3
77     elif message_count >= 10:
78         return 2
79     return 1
80
81 # 根據角色與等級選擇語氣風格
82 def get_persona_prompt(role_name, level):
83
問題 輸出 偵錯主控台 終端機 連接埠
PS C:\Users\User\Desktop\project> & c:/Users/User/Desktop/project/.venv/Scripts/python.exe c:/Users/User/Desktop/
成功連接到資料庫
成功建立 user_states 表格
```

測試是否連線成功



步驟二：安裝 AWS CLI

- 從 AWS 官網下載 CLI 工具



The screenshot shows the AWS CLI installation guide page. The browser address bar displays the URL: docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html. The page header includes the AWS logo and navigation links: 開始使用, 服務指南, 開發人員工具, and AI 資. The left sidebar contains a navigation menu with the following items: AWS Command Line Interface (selected), User Guide for Version 2, About the AWS CLI, Getting started, Prerequisites, Install/Update (highlighted in blue), Past releases, Build and install from source, Amazon ECR Public/Docker, Setup, Configure the AWS CLI, and Authentication and access. The main content area features a terminal code block with the command: `$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"`. Below the code block, there is a list of instructions: 1. Downloading from the URL – To download the installer with your browser, use https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip. 2. (Optional) Verifying the integrity of your downloaded zip file. If you chose to manually download the AWS CLI installer package `.zip` in the above step, you can verify the integrity of the file by checking the signatures by using the `GnuPG` tool. The AWS CLI installer package `.zip` files are cryptographically signed using PGP. If you do not have the public key file, this verification fails and you should not proceed with installation. a. Download and install the `gpg` command using your package manager. For more information, see [this website](#). b. To create the public key file, create a text file and paste in the following text.

安裝AWS CLI

- 安裝完需重開Visual Studio, 否則會error
- 再執行 `aws --version`指令(注意有空格, 和2個--)
- `aws configure` 指令
- 輸入 Access Key、Secret Key、Region (例:us-east-1)與預設格式(json)

步驟三：安裝 Zappa

- 執行 `pip install zappa`
- 如果出現錯誤訊息, 可以先執行 `set PYTHONUTF8=1`, 再重新執行一次

如果安裝zappa過程有問題

```
set PYTHONUTF8=1  
pip install zappa
```

初始化 Zappa

- 執行 `zappa init` 建立設定檔
- 確認按Enter不是按y
- 成功左側會出現`zappa_settings.json`
- 顯示內容如右

```
> .venv
create_user.py
main.py
requirements.txt
zappa_settings.json

48 handler = web
49
50 # 資料庫連線參
51 db_host = "fl
52 db_name = "te
53 db_user = "yv
54 db_pass = "12
55
56 def create_co
```

```
{ } zappa_settings.json > ...
1 {
2   "dev": {
3     "app_function": "main.app",
4     "aws_region": "us-east-1",
5     "exclude": [
6       "boto3",
7       "dateutil",
8       "botocore",
9       "s3transfer",
10      "concurrent"
11    ],
12    "profile_name": "default",
13    "project_name": "project",
14    "runtime": "python3.12",
15    "s3_bucket": "zappa-yvonne-0711"
16  }
17 }
```

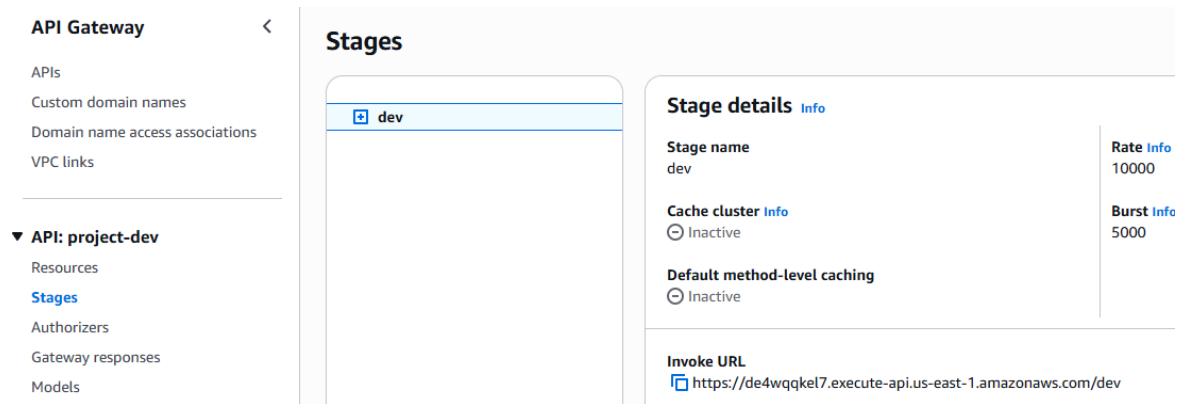
Zappa建立一個dev的部署階段

- 執行 'zappa deploy dev 進行首次部署
- 可使用 'zappa update dev, 再次部署要用update
- 如果忘記了deploy的url, zappa status dev

```
問題 輸出 偵錯主控台 終端機 連接埠
PS C:\Users\User\Desktop\project> zappa status dev
Status for project-dev:
  Lambda Versions:      3
  Lambda Name:          project-dev
  Lambda ARN:           arn:aws:lambda:us-east-1:383425823448:function:project-dev
  Lambda Role ARN:      arn:aws:iam::383425823448:role/project-dev-ZappaLambdaExecutionRole
  Lambda Code Size:     44397848
  Lambda Version:       $LATEST
  Lambda Last Modified: 2025-07-11T02:49:12.000+0000
  Lambda Memory Size:   512
  Lambda Timeout:       30
  Lambda Handler:       handler.lambda_handler
  Lambda Runtime:       python3.12
```

取得 Lambda URL

- Zappa 部署完成後產生的URL
- 此 URL 可用於Line Messaging API
- 要先找api gateway, 找到部署的那支lambda點擊進去



The screenshot displays the AWS API Gateway console interface. On the left, a navigation menu is visible under the heading 'API Gateway', with 'APIs', 'Custom domain names', 'Domain name access associations', and 'VPC links' listed. Below this, a dropdown menu is expanded for 'API: project-dev', showing 'Resources', 'Stages', 'Authorizers', 'Gateway responses', and 'Models'. The 'Stages' section is selected, showing a list with one entry: 'dev'. To the right of the list, the 'Stage details' for the 'dev' stage are displayed. The 'Stage name' is 'dev'. The 'Cache cluster' is 'Inactive'. The 'Default method-level caching' is also 'Inactive'. The 'Invoke URL' is shown as <https://de4wqqkel7.execute-api.us-east-1.amazonaws.com/dev>. On the far right, performance metrics are shown: 'Rate' is 10000 and 'Burst' is 5000.

Stage name	Rate
dev	10000

Cache cluster	Burst
Inactive	5000

Invoke URL: <https://de4wqqkel7.execute-api.us-east-1.amazonaws.com/dev>

步驟四：整合到 Linebot

- 登入 Line Developers
- 在 Messaging API 的 Webhook 欄位填入 Lambda URL
- 檢驗成果, OK

The image shows two screenshots side-by-side. The left screenshot is from the AWS Lambda console, displaying the 'Webhook settings' for a function. It shows a 'Webhook URL' field with the value 'https://de4wqqkel7.execute-api.us-east-1.amazonaws.com/dev/callback' and two buttons: 'Verify' and 'Edit'. Above the URL field, there is a note: 'Scan this QR code with LINE to add your LINE Official Account as a friend. You can share the code with others.' The right screenshot is a LINE chat interface. At the top, the contact name is '晶鑫珠寶'. A green bubble contains the message '你好' with a timestamp of '上午 11:41'. A grey bubble contains a longer message: '喵喔~👁️ 又來找奴才啦，真是讓人又驚又喜喵！哼，不過你這次來得倒也挺及時喵，奴才正想要和你多聊一點呢~🐱💎 你今天過得怎樣喵？有沒有什麼好玩的事情要跟我分享喵？不過啊，說真的啦，奴才可是很怕你長時間待著喵，一不小心就會被你折騰壞啦🤪❤️ 你可要乖乖的喵，不然奴才可是會用萌萌的眼神狠狠地看你喵~🐾' with a timestamp of '上午 11:41'.