

Youblike即時地圖資訊

# 撰寫程式的順序

# 製作程式的順序

- 1.打開google瀏覽器
- 2.搜尋gemini AI
- 3.將Python(爬蟲) + Folium(畫地圖用) + Flask(網頁伺服器)等...關鍵字與需求輸入AI對話框
- 4.Ctrl+C and Ctrl+V ，貼到該貼的對應副檔名上
- 5.測試是否可以正常執行到想要的結果
- 6.結束去玩自己的遊戲了

# 撰寫程式的順序

- 總共分五部份
  - 一、先找尋、整理Youbike資料來源
  - 二、使用folium顯示指定區域的地圖，找到站點畫出指定圖形、標記、youbike資訊
  - 三、使用OSMnx套件載入道路節點經緯度座標資訊，並利用內建方法(最短路徑算法) 找出最短路徑
  - 四、用leaflet在地圖上畫紅線標示
  - 五、使用Flask(網頁伺服器)動態更新地圖

# 一、找尋、整理Youbike資料來源

- 1. 台北市youbike查詢網站

<https://data.taipei/dataset/detail?id=c6bc8aed-557d-41d5-bfb1-8da24f78f2fb>

- 2.

檔案下載				
資料項目	檔案大小	檔案格式	更新時間	操作
YouBike2.0臺北市公共自行車即時資訊		JSON	系統介接	<a href="#">下載</a> <a href="#">預覽</a>
詮釋資料				
主題分類	交通			
數位發展部服務分類	交通及通訊			
數位發展部資料集類型	原始資料			
主要權位說明	sno(站點代號)、sna(場站中文名稱)、quantity(場站總停車格)、available_rent_bikes(場站目前車輛數量)、sarea(場站區域)、mday(資料更新時間)、latitude(緯度)、longitude(經度)、ar(地點)、sareaen(場站區域英文)、snaen(場站名稱英文)、aren(地址英文)、available_return_bikes(空位數量)、act(全站禁用狀態)、srcUpdateTime(YouBike2.0系統發布資料更新的時間)、updateTime(大數據平台經過處理後將資料存入DB的時間)、infoTime(各場站來源資料更新時間)、infoDate(各場站來源資料更新時間)			
資料集描述	YouBike2.0臺北市公共自行車即時資訊介接網址為 <a href="https://tcgbusfs.blob.core.windows.net/dotapp/youbike/v2/youbike_immediate.json">https://tcgbusfs.blob.core.windows.net/dotapp/youbike/v2/youbike_immediate.json</a> ，檔案格式為json檔。			
資料集提供機關	交通局			
資料授權	公開			
更新頻率	每1分			
詮釋資料更新時間	2025-06-24 09:49:16			
資料集語系	中文			
計費方式	免費			
機關聯絡人	李彥安			
機關聯絡人電話	02-27208889#6907			
機關聯絡人電子郵件	pn9607@gov.taipei			

```
[
  {
    "sno": "500101001",
    "sna": "YouBike2.0 捷運科技大樓站",
    "sarea": "大安區",
    "mday": "2025-06-26 01:10:02",
    "ar": "復興南路二段235號前",
    "sareaen": "Daan Dist.",
    "snaen": "YouBike2.0 MRT Technology Bldg. Sta.",
    "aren": "No.235, Sec. 2, Fuxing S. Rd.",
    "act": "1",
    "srcUpdateTime": "2025-06-26 01:26:30",
    "updateTime": "2025-06-26 01:26:52",
    "infoTime": "2025-06-26 01:10:02",
    "infoDate": "2025-06-26",
    "Quantity": 28,
    "available_rent_bikes": 8,
    "latitude": 25.02605,
    "longitude": 121.5436,
    "available_return_bikes": 19
  },
  {
    "sno": "500101002",
    "sna": "YouBike2.0 復興南路二段273號前",
    "sarea": "大安區",
    "mday": "2025-06-26 00:30:04",
    "ar": "復興南路二段273號西側",
    "sareaen": "Daan Dist.",
    "snaen": "YouBike2.0 No.273, Sec. 2, Fuxing S. Rd.",
    "aren": "No.273, Sec. 2, Fuxing S. Rd. (West)",
    "act": "1",
    "srcUpdateTime": "2025-06-26 01:26:30",
    "updateTime": "2025-06-26 01:26:52",
    "infoTime": "2025-06-26 00:30:04",
    "infoDate": "2025-06-26",
    "Quantity": 21,
    "available_rent_bikes": 5,
    "latitude": 25.02565,
    "longitude": 121.54357,
    "available_return_bikes": 16
  },
  {
    "sno": "500101003",
    "sna": "YouBike2.0 國北敦大寶小東側門",
    "sarea": "大安區",
    "mday": "2025-06-26 01:10:02",
    "ar": "和平東路二段96巷7號",
    "sareaen": "Daan Dist.",
    "snaen": "YouBike2.0 NTUE Experiment Elementary School (East)",
    "aren": "No. 7, Ln. 96, Sec. 2, Heping E. Rd.",
    "act": "1",
    "srcUpdateTime": "2025-06-26 01:26:30",
    "updateTime": "2025-06-26 01:26:52",
    "infoTime": "2025-06-26 01:10:02",
    "infoDate": "2025-06-26"
```

# 一、Youbike資料來源 程式碼

```
# 讀取youbike資料

def load_youbike():
    # # 爬蟲目標:Youbike臺北市公共自行車即時資訊
    url = "https://tcgbusfs.blob.core.windows.net/dotapp/youbike/v2/youbike\_immediate.json"

    response = requests.get(url)
    global youbike_Data
    # 若請求狀態為200則輸出respinse物件內容
    if response.status_code == requests.codes.ok :
        print("取得網頁內容成功")
        print("網頁內容:")
        youbike_Data = response.json()
    else:
        print("取得網頁內容失敗")
```

## 二、使用folium顯示地圖

- 是基於leaflet.js的python地圖套件
- #安裝folium ==> pip install folium





## 二、程式碼

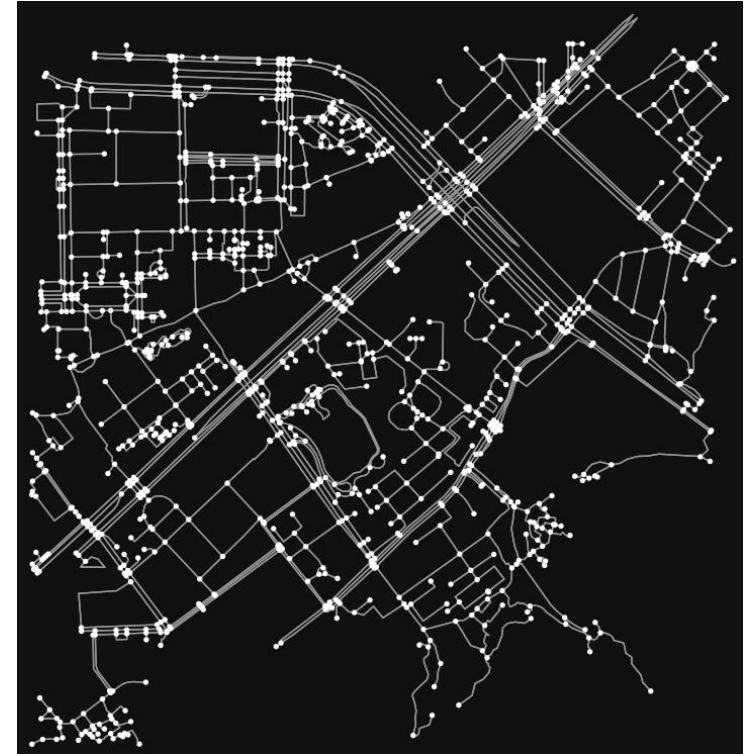
- 大部分都是在javascript.js檔上面使用leaflet命令是去畫圖。

```
function updata_youbike(youbikeData) {  
  // 清除所有現有的 YouBike 標記，避免重複添加  
  youbikeMarkers.forEach(function(marker) {  
    map.removeLayer(marker);  
  });  
  youbikeMarkers = []; // 清空陣列  
  
  if (!youbikeData || youbikeData.length === 0) {  
    console.warn("YouBike 數據為空或未定義，無法添加標記。");  
    return;  
  }  
  
  // 遍歷 YouBike 數據中的每個站點  
  youbikeData.forEach(function(station) {  
    // 假設每個站點物件有 'lat' (緯度) 和 'lng' (經度) 屬性  
    // 你需要根據你的實際 JSON 結構來調整這裡的屬性名  
    console.log("station:", station);  
    const lat = station.latitude; // 請確認實際的緯度屬性名  
    const lon = station.longitude; // 請確認實際的經度屬性名  
    console.log("lat:", lat, ", lon:", lon);  
    // 確保經緯度存在且為數字  
    if (typeof lat === 'number' && typeof lon === 'number') {  
      // 構建彈出視窗的內容  
      // 你可以根據 station 物件的實際屬性來構建更豐富的資訊  
      let popupContent = `${station.sna}</b><br>`; // 站點名稱 (sna)  
      popupContent += `總停車格: ${station.quantity}<br>`; // 總停車格數 (sbi)  
      popupContent += `可借車輛: ${station.available_rent_bikes}<br>`; // 可借車輛數 (bemp)  
      popupContent += `車位空位: ${station.available_return_bikes}<br>`; // 空位數量 (available_return_bike)  
      popupContent += `該站資料更新時間: ${station.mday}`; // 該站車輛更新時間 (mday)  
      popupContent += `獲取資料時間: ${station.updateTime}`; // 獲取資料時間 (mday)  
  
      // 創建一個 Leaflet Marker  
      const marker = L.marker([lat, lon])  
        .bindPopup(popupContent) // 綁定彈出視窗  
        .addTo(map); // 添加到地圖上  
  
      // 將標記儲存到陣列中，以便管理  
      youbikeMarkers.push(marker);  
    } else {  
      console.warn("YouBike 站點數據缺少經緯度或格式不正確:", station);  
    }  
  });  
  
  console.log(`已添加 ${youbikeMarkers.length} 個 YouBike 站點標記。`);  
  // 你可以在頁面載入後立即呼叫這個函數，或者透過點擊按鈕來觸發它  
  // 這裡我們假設你希望點擊 "更新youbike時間" 按鈕時才顯示  
  // 如果想在頁面載入時就顯示，可以將這行代碼放在函數定義之後，或在 DOMContentLoaded 事件中觸發  
  // addYouBikeMarkers(youbike_data_js);  
}
```



## 三、使用OSMnx套件

- OSMnx結合了Overpass API，讓開發者可以在python中快速取得OpenStreetMap的資料。



# 三、OSMnx程式碼

```
def calculate_route():
    """
    接收前端發送的起點和終點座標，
    計算最短路徑，並返回路徑的經緯度點。
    """

    data = request.get_json()
    # 確保座標類型正確
    try:
        origin_lat = float(data['origin']['lat'])
        origin_lon = float(data['origin']['lon'])
        dest_lat = float(data['destination']['lat'])
        dest_lon = float(data['destination']['lon'])
    except ValueError:
        return jsonify({'error': 'Invalid coordinate format.'}), 400
    # 更新G圖層的中心位置
    print("origin=", origin_lat, origin_lon)
    print("dest=", dest_lat, dest_lon)
    # dist_var=
    G = ox.graph_from_point(( (origin_lat+dest_lat)/2, (origin_lon+dest_lon)/2 ), dist=4000, network_type='all')
    print("new_G_center=", (origin_lat+dest_lat)/2, ",", (origin_lon+dest_lon)/2)

    if G is None:
        return jsonify({'error': 'OSMnx graph not loaded.'}), 500

    try:
        # 尋找最近的圖形節點 (注意 OSMnx nearest_nodes 接受 (經度, 緯度))
        origin_node = ox.nearest_nodes(G, origin_lon, origin_lat)
        destination_node = ox.nearest_nodes(G, dest_lon, dest_lat)

        # 計算最短路徑 (這裡使用 'length' 作為權重, 您也可以嘗試 'travel_time')
        route = nx.shortest_path(G, origin_node, destination_node, weight="length")

        # 提取路徑上所有節點的經緯度座標
        route_coordinates = []
        for node_id in route:
            node_data = G.nodes[node_id]
            # Folium PolyLine 通常需要 (緯度, 經度) 格式
            route_coordinates.append([node_data['y'], node_data['x']])

        return jsonify({'route_coords': route_coordinates})

    except Exception as e:
        print(f"計算路徑時發生錯誤: {e}")
        # 如果找不到路徑, 可能是因為點不在網路中或網路不連通
        return jsonify({'error': f'無法計算路徑: {str(e)} - 請確保所選點位於道路網中並相互連通.'}), 500
```

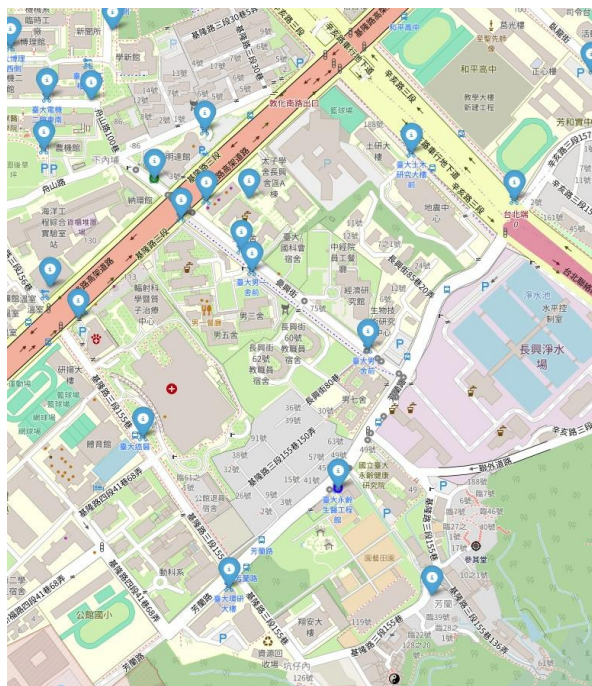
## 四、用leaflet在地圖上畫紅線標示

- 這邊基本都是在javascript上寫leaflet寫程式碼

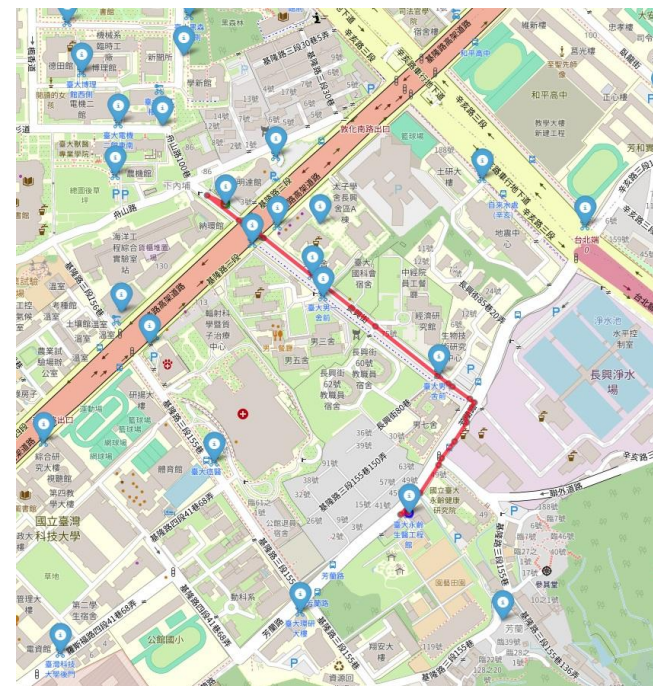
道路圖層



地理圖層



路徑圖





## 四、用leaflet畫線程式碼

```
function calculateAndDrawRoute() {
  if (!originPoint || !destinationPoint) {
    errorMessage.innerHTML = '請先設定起點和終點!';
    return;
  }

  // 清除舊的路徑線條
  if (routeLine) {
    map.removeLayer(routeLine);
  }
  errorMessage.innerHTML = '正在計算路徑...';

  // 發送 AJAX 請求到 Flask 後端
  $.ajax({
    url: '/calculate_route',
    type: 'POST',
    contentType: 'application/json',
    data: JSON.stringify({
      origin: originPoint,
      destination: destinationPoint
    }),
    success: function(response) {
      errorMessage.innerHTML = ''; // 清除計算中訊息
      if (response.route_coords && response.route_coords.length > 0) {
        // 收到新的路徑座標, 重新繪製線條
        //console.log("response.route_coords=", response.route_coords)
        routeLine = L.polyline(response.route_coords, {
          color: 'red',
          weight: 5,
          opacity: 0.7
        }).addTo(map);

        // 自動調整地圖視窗以顯示整條路徑
        map.fitBounds(routeLine.getBounds());
      } else {
        errorMessage.innerHTML = '未找到路徑或路徑數據為空。';
      }
    },
    error: function(xhr, status, error) {
      errorMessage.innerHTML = '計算路徑失敗: ' + (xhr.responseJSON ? xhr.responseJSON.error : error);
      console.error("AJAX Error: ", xhr.responseText);
    }
  });
}
```

# 五、使用Flask(網頁伺服器)動態更新地圖

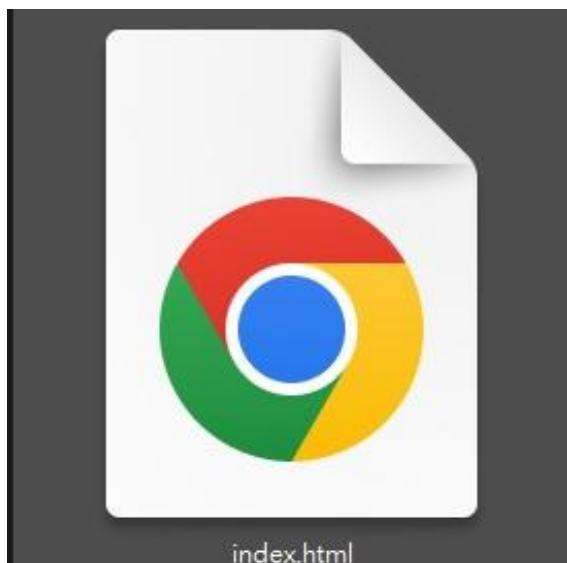
專案結構

建立一個基本Flask專案，須包含以下檔案：

```
your_project_folder/  
├── app.py          # Flask 應用程式主文件  
├── templates/  
│   └── index.html  # 網頁模板  
└── static/  
    └── script.js   # 前端 JavaScript 文件 (處理地圖互動和 AJAX)
```

本機端

<http://127.0.0.1:5000>



# Flask架構

- UI(使用者介面配置)

- 畫地圖
- 動態響(On\_click)

- 啟動服務器
- 建立API接口
- 處理資料

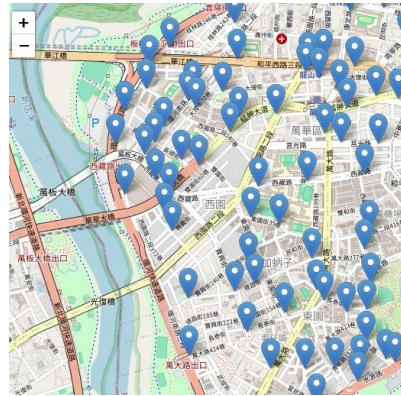
點選地圖設定起點和終點

起點: 未設定 | 終點: 未設定

重設點位

計算並繪製路徑

顯示youbike站點



```
@app.route('/calculate_route', methods=['POST'])
def calculate_route():
    """
    接收前端發送的起點和終點座標，
    計算最短路徑，並返回路徑的經緯度點。
    """
```



index.html



script.js



app.py

# 參考資料

## 一、台北市youbike查詢網站

<https://data.taipei/dataset/detail?id=c6bc8aed-557d-41d5-bfb1-8da24f78f2fb>

## 二、folium

1.小狐狸事務所 <https://yhhuang1966.blogspot.com/2022/08/python-folium.html>

2.Gemini(Google AI) <https://gemini.google.com>

## 三、OSMnx

1.Day19、20 OSMnx取得OpenStreetMap資料(2019 iT 邦幫忙鐵人賽)

<https://ithelp.ithome.com.tw/articles/10207749>

2.Gemini(Google AI) <https://gemini.google.com>

3.python 包介绍：osmnx (CSDN) [https://blog.csdn.net/qq\\_40206371/article/details/120407459](https://blog.csdn.net/qq_40206371/article/details/120407459)

4.OSMnx 2.0.4 documentation <https://osmnx.readthedocs.io/en/stable/user-reference.html>

## 四、Python Flask

1.Gamma Ray 軟體工作室(youtuber) <https://www.youtube.com/watch?v=AiUzsr5JZgQ&t=36s>

2.Gemini(Google AI) <https://gemini.google.com>