

利用 Flask 建立支援 Line Bot 的後勤網站

學習目標：

- 了解 Flask 的用法！
- 建立支援 Line Bot 機器人的網站！

利用 Bootstrap 製作 Flask Web 站台網頁

1. 開啟 templates/home.html 檔案，導入 Bootstrap 連結：

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-EVSTON3/azprG1Anm3QDgpJLIm9Nao8Yz1ztC0TfwFpsD3D65VohhpuuCOMLAsJc" crossorigin="anonymous">
  <title>MySite</title>
</head>
<body>
  Hello , Hacker !!
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-Mrc0WZHYLzclABNL+HUVF6sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM" crossorigin="anonymous"></script>
</body>
</html>
```

2. 使用導航列！

```
(前方略過....)
<body>
  <nav class="navbar navbar-expand-sm navbar-dark bg-primary">
    <ul class="navbar-nav">
      <li class="nav-item">
        <a class="nav-link" href="#">Home</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Link</a>
      </li>
    </ul>
  </nav>
(後方略過....)
```

3. 使用折疊式按鍵，方便做成 RWD 網頁

```
(前方略過....)
<body>
  <nav class="navbar navbar-expand-sm navbar-dark bg-primary">
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#collapsibleNavbar">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="collapsibleNavbar">
      <ul class="navbar-nav">
(後方略過....)
```

4. 使用 Jumbotron

```
(前方略過....)
</nav>
<div class="jumbotron">
  <h1>Hello , Hacker !!</h1>
  <p>這是測試頁...</p>
</div>
(後方略過....)
```

5. 使用 Google fonts：

```
(前方略過....)
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Noto+Sans+TC:wght@300&display=swap" rel="stylesheet">

<title>MySite</title>
<style> body {font-family: 'Noto Sans TC', sans-serif;}</style>
</head>
(後方略過....)
```

利用 Jinja2 模板引擎打造網頁

1. 將 home.html 檔案，複製到 base.html 檔案
2. 修改 base.html 檔案中，可能異動的部份：

```
(前方略過....)
<title>{% block title %}{% endblock %}</title>
(中間略過....)
<!-- 取代 jumbotron 那一段 -->
<main>
  {% block main %}{% endblock %}
</main>

<footer>
</footer>
(中間略過....)
{% block script %}{% endblock %}
</body>
```

3. 修改 home.html 檔案，去掉重複的部份，並加入 base.html 樣板檔：

```
{% extends "base.html" %}
{% block title %}My Web Site{% endblock %}
{% block main %}
  <div class="jumbotron">
    <h1>Hello , Hacker !!</h1>
    <p>這是測試頁...</p>
  </div>
  <div class="container my-5 py-5">
    <div class="row">
      <div class="col">
        <h2>左側</h2>
      </div>
      <div class="col">
        <h2>中間</h2>
      </div>
      <div class="col">
        <h2>右側</h2>
      </div>
    </div>
  </div>
{% endblock %}
```

4. 送上 Heroku 主機，測試看看內容！

5. 修改 app/router.py 檔案內容，加入導向查詢菜單的網頁位置：

```
# 導向查詢菜單內容的網頁
app.route("/showmenu")
def showmenu():
    return render_template("menu.html")
```

6. 新增菜單的網頁 app/templates/menu.html :

```
{% extends "base.html" %}
{% block title %}大學書店菜單系統{% endblock %}

{% block main %}
<div class="container my-5 py-5">
  <div class="row">
    <div class="col">
      <h1>菜單列表</h1>
    </div>
  </div>

  {% for menu in menuList %}
  <div class="row">
    <div class="col">
      <p>{{ menu[0] }}</p>
    </div>
    <div class="col">
      <p>{{ menu[1] }}</p>
    </div>
    <div class="col">
      <p>{{ menu[2] }}</p>
    </div>
  </div>
  {% endfor %}
</div>
{% endblock %}
```

7. 送上 Heroku 主機，測試看看內容！