Week 2: Build Web apps with Flask: Part1

- Introduction to Flask
- Creating simple flask app
- Creating flask templates

Introduction to Flask

Flask is a **micro web framework** written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

Flask depends on the **Jinja** template engine and the **Werkzeug** WSGI toolkit.

What is Werkzeug?

Werkzeug is a comprehensive WSGI web application library. It began as a simple collection of various utilities for WSGI applications and has become one of the most advanced WSGI utility libraries.

Jinja

Jinja is a modern and designer-friendly templating language for Python, modelled after Django's templates. It is fast, widely used and secure with the optional sandboxed template execution environment

Dhambaal App Structure

```
DHAMBAAL
 ├──── Pipfile
 ├──── Pipfile.lock
 ├──── README.adoc
 ├──── README.pdf
 ├──── Resources
 │    ├──── app.js
 │    ├──── app.py
 │    ├──── classes_1.py
 │    ├──── classes_2.py
 │    ├──── decorator.py
 │    ├──── dict.py
 │    ├──── functions_1.py
 │    ├──── generator_1.py
 │    ├──── lists.py
 │    ├──── project_homework.py
 │    ├──── sample.py
 │    ├──── set_1.py
 │    └──── tuple_1.py
 ├──── app.py
 └──── dhambaal
     ├──── __init__.py
     ├──── static
     │    ├──── css
     │    ├──── images
     │    └──── js
     └──── templates
         └──── index.html
```