

PROJECT 6

Objectives

1. To model real-world flowers as objects with attributes (price, color, smell) and behaviors (input, display) using object-oriented programming principles.
2. To practice encapsulation by grouping related data (properties) and functions (methods) into a single class.
3. To demonstrate reusability by creating multiple flower objects (lilly, rose, hibiscus) without rewriting code.

I. Write a class with following criteria

Class name: Flower

Objects: lilly, rose, hibiscus

Properties: price,color,smell

Methods: get(), display()

II. Draw the flowchart for the above

Submission date: Each class to submit on the day we will meet from 11th to 16th August 2025

PROJECT 7

Objective

1. Automate and organize logging across the application using decorators for function tracking and factory functions for module-specific loggers.

I. You are building a logging system for a scientific computation application. To reduce boilerplate logging code, you decide to use a decorator that logs function entry and exit, and a factory function to generate customized loggers based on module name.

carry out the following task:

- i. Write a decorator that prints "Entering <function_name>" before the function executes and "Exiting <function_name>" afterward.
- ii. Create a factory function `make_logger(prefix)` that returns such a decorator with a custom log prefix (e.g., "[MathModule]" or "[NetworkModule]").
- iii. Apply the generated decorator to a sample function.

II. Draw the flowchart

Submission date: Each class to submit on the day we will meet from 11th to 16th August 2025

PROJECT 8

To be submitted before 2:00 pm on Wednesday 20th August 2025

Objectives

1. **Learn Django basics** – Understand the main building blocks like models, views, templates, and routing to create backend applications.
2. **Apply Django in practice** – Build and run a working web app using Django's features and workflow.

Watch the video with this url and implement same

<https://youtu.be/rHux0gMZ3Eg>

Also save it in your GitHub account and write the link in your logbook