Practice Day 8.5

Project Name:

Smart Vehicle Classifier

Description

This project simulates a simple AI-based vehicle classifier using Object-Oriented Programming (OOP).
It demonstrates **inheritance**, **protected and private members**, **method overriding**, and **encapsulation**.

The program manages two types of vehicles: **Car** and **Bike**. Each class inherits from a base class Vehicle.
A simple classifier method predicts the type of vehicle based on its speed threshold.

Task Statement

Design and implement a simple program that classifies vehicles based on their type and speed using Object-Oriented Programming (OOP) principles.

You must:

1. Create a base class Vehicle that stores vehicle type, name, and speed.

2. Use **protected** and **private** attributes in the base class to demonstrate encapsulation.

3. Implement a method classify() that categorizes vehicles as **High-speed Vehicle** if the speed is greater than 100, otherwise **Normal Vehicle**.

4. Derive two subclasses — Car and Bike — from Vehicle. Each should override the show_info() method to display detailed information.

5. Read multiple inputs representing different vehicles, create corresponding objects, and print their details along with their classification results.

Input

3

Car Toyota 150

Bike Yamaha 90

```
Car Honda 120
```

Output

```
Car Added: Toyota
Bike Added: Yamaha
Car Added: Honda

--- Vehicle Details ---
Type: Car, Name: Toyota, Speed: 150 km/h, Predicted: High-speed Vehicle
Type: Bike, Name: Yamaha, Speed: 90 km/h, Predicted: Normal Vehicle
Type: Car, Name: Honda, Speed: 120 km/h, Predicted: High-speed Vehicle
```

Short Hint

Use a base class Vehicle with **protected** and **private** attributes.
Create subclasses Car and Bike that override show_info() and add a simple speed-based classifier.
Speed above 100 is considered a "High-speed Vehicle".