1. Explain what inheritance is in object-oriented programming and why it is used.

2. Discuss the concept of single inheritance and multiple inheritance, highlighting their differences and advantages.

3. Explain the terms "base class" and "derived class" in the context of inheritance.

4. What is the significance of the "protected" access modifier in inheritance? How does it differ from "private" and "public" modifiers?

5. What is the purpose of the "super" keyword in inheritance? Provide an example.

6. Create a base class called "Vehicle" with attributes like "make", "model", and "year". Then, create a derived class called "Car" that inherits from "Vehicle" and adds an attribute called "fuel_type". Implement appropriate methods in both classes.

7. Create a base class called "Employee" with attributes like "name" and "salary." Derive two classes, "Manager" and "Developer," from "Employee." Add an additional attribute called "department" for the "Manager" class and "programming_language" for the "Developer" class.

8. Design a base class called "Shape" with attributes like "colour" and "border_width." Create derived classes, "Rectangle" and "Circle," that inherit from "Shape" and add specific attributes like "length" and "width" for the "Rectangle" class and "radius" for the "Circle" class.

9. Create a base class called "Device" with attributes like "brand" and "model." Derive two classes, "Phone" and "Tablet," from "Device." Add specific attributes like "screen_size" for the "Phone" class and "battery_capacity" for the "Tablet" class.

10. Create a base class called "BankAccount" with attributes like "account_number" and "balance." Derive two classes, "SavingsAccount" and "CheckingAccount," from "BankAccount." Add specific methods like "calculate_interest" for the "SavingsAccount" class and "deduct_fees" for the "CheckingAccount" class.

Submission Guidelines:

- Answer all the questions in a single Jupyter Notebook file (.ipynb).
- Include necessary code, comments, and explanations to support your answers and implementation.
- Ensure the notebook's cells containing code are already run.
- Create a GitHub repository to host your assignment files.
- Rename the Jupyter Notebook file using the format "date_month_topic.ipynb" (e.g., "02_July_OOPs_inheritance.ipynb").
- Place the Jupyter Notebook file in the repository.
- Ensure the repository is publicly accessible.

- Submit the link of the corresponding assignment present in your GitHub repository as the assignment submission link.

Note:- Create your assignment in Jupyter notebook and upload it to GitHub & share that uploaded assignment file link through your dashboard. Make sure the repository is public.