

practice coding questions based on object-oriented programming (OOP) in Python:

### Question 1:

Implement a simple class **Rectangle** that represents a rectangle. The class should have methods to calculate the area and perimeter of the rectangle.

### Question 2:

Create a class **Student** with attributes name, age, and grade. Implement a method to display the student details. Create instances of the **Student** class and display their details.

### Question 3:

Extend the previous **Student** class by adding a method **calculate\_grade** that takes a percentage as an argument and returns the corresponding grade (A, B, C, etc.).

### Question 4:

Design a class hierarchy for a zoo. Create a base class **Animal** with attributes like name, species, and age. Then, create derived classes for specific types of animals, such as **Lion**, **Elephant**, and **Giraffe**. Implement methods to display information specific to each type of animal.

### Question 5:

Create a class **Bank** with attributes balance and account holders. Implement methods to deposit money, withdraw money, and display the account balance. Create multiple instances of the **Bank** class with different account holders.

### Question 6:

Implement a class **Book** with attributes title, author, and publication year. Create a method to display the book details. Then, create a derived class **EBook** that adds an attribute for the file size. Override the display method in the **EBook** class to include the file size.

### Question 7:

Design a class **Shape** with an abstract method **calculate\_area**. Create derived classes for specific shapes like **Circle** and **Square**. Implement the **calculate\_area** method in each derived class.

### Question 8:

Create a class **Person** with attributes name, age, and address. Implement a method **is\_adult** that returns True if the person is 18 years or older, otherwise False.

### Question 9:

Design a class **Car** with attributes make, model, and year. Implement a method **is\_old** that returns True if the car is more than 10 years old, otherwise False.

### Question 10:

Implement a class **BankAccount** with attributes account\_number, account\_holder, and balance. Create a derived class **SavingsAccount** that adds an attribute for the interest rate. Implement methods for deposit, withdraw, and calculate interest in the **SavingsAccount** class.