21-08-24

Javascript

Oops concept

1. Inheritance

Problem: Create a base class `Animal` with the following properties:

- `name` (string)
- `age` (number)

Add a method `speak` that returns `"Animal sound"`.

Create a subclass called `Dog` that inherits from `Animal`. Override the `speak` method to return `"Woof"`.

Additional Requirement: Create another subclass `Cat` that also inherits from `Animal`. Override the `speak` method to return `"Meow"`.

2. Encapsulation

Problem: Create a class `BankAccount` that encapsulates the following properties:

- `accountNumber` (string)
- `balance` (number)

Implement the following methods:

- `deposit(amount)`: Adds the given amount to the balance.
- `withdraw(amount)`: Subtracts the given amount from the balance if sufficient funds are available.
- `getBalance()`: Returns the current balance.

Ensure that the `balance` property is private and cannot be accessed directly from outside the class.

3. Polymorphism

Problem: Create a base class `Shape` with a method `calculateArea` that returns `0`.

Create two subclasses: `Rectangle` and `Circle`.

- `Rectangle` should have properties `width` and `height` and override `calculateArea` to return the area of the rectangle (`width * height`).
- `Circle` should have a property `radius` and override `calculateArea` to return the area of the circle (`Math.PI * radius^2`).

Create an array of `Shape` objects (including both `Rectangle` and `Circle`) and print the area of each shape by calling `calculateArea`.