**Practice Problem on Single Inheritance**

Create a class “Vehicle” with a method “start()” that prints “Vehicle started”.
Create a subclass “Car” that extends “Vehicle” and overrides the “start()” method to print “Car started”.
Create an object of the “Vehicle” class and call the “start()” method.
Create an object of the “Car” class and call the “start()” method.

Create a class “Person” with fields “name” and “age” and a method “display()” that prints the name and age of the person.
Create a subclass “Employee” that extends “Person” and adds a field “salary” and a method “display()” that prints the name, age, and salary of the employee.
Create an object of the “Person” class and call the “display()” method.
Create an object of the `Employee` class and call the “display()” method.

Create a class “Shape” with a method “getArea()” that returns the area of the shape.
Create a subclass “Rectangle” that extends “Shape” and adds fields “length” and “width” and overrides the “getArea()” method to return the area of the rectangle.
Create an object of the “Shape” class and call the “getArea()” method.
Create an object of the “Rectangle” class and call the “getArea()” method.

Create a class “Animal” with a method “makeSound()” that prints a sound.
Create a subclass “Dog” that extends “Animal” and overrides the “makeSound()” method to print “Woof!”.
Create an object of the “Animal” class and call the “makeSound()” method.
Create an object of the “Dog” class and call the “makeSound()” method.

Create a class “BankAccount” with fields “accountNumber”, “balance”, and “interestRate” and a method “deposit()” that adds an amount to the balance.
Create a subclass “SavingsAccount” that extends “BankAccount” and adds a field “minimumBalance” and a method “withdraw()” that subtracts an amount from the balance.
Create an object of the “BankAccount” class and call the “deposit()” method.
Create an object of the “SavingsAccount” class and call the “deposit()” and “withdraw()” methods.

# Practice Problem on Multilevel Inheritance

Create a class “Person” with fields “name” and “age” and a method “display()” that prints the name and age of the person.
Create a subclass “Employee” that extends “Person” and adds a field “salary” and a method “display()” that prints the name, age, and salary of the employee.
Create a subclass “Manager” that extends “Employee” and adds a field “department” and a method “display()” that prints the name, age, salary, and department of the manager.
Create an object of the “Person” class and call the “display()” method.
Create an object of the “Employee” class and call the “display()” method.
Create an object of the “Manager” class and call the “display()” method.

Create a class “Animal” with a method “makeSound()” that prints a sound.
Create a subclass “Mammal” that extends “Animal” and adds a method “giveBirth()” that prints “Giving birth”.
Create a subclass “Dog” that extends “Mammal” and overrides the “makeSound()” method to print “Woof!”.
Create an object of the “Animal” class and call the “makeSound()” method.
Create an object of the “Mammal” class and call the “makeSound()” and “giveBirth()” methods.
Create an object of the “Dog” class and call the “makeSound()” and “giveBirth()” methods.

Create a class “Vehicle” with a method “start()” that prints “Vehicle started”.
Create a subclass “Car” that extends “Vehicle” and adds a method “drive()” that prints “Car driving”.
Create a subclass “SportsCar” that extends “Car” and overrides the “drive()” method to print “Sports car driving fast”.
Create an object of the “Vehicle” class and call the “start()” method.
Create an object of the “Car” class and call the “start()” and “drive()” methods.
Create an object of the “SportsCar” class and call the “start()” and “drive()” methods.

Create a class “Shape” with a method “getArea()” that returns the area of the shape.
Create a subclass “Rectangle” that extends “Shape” and adds fields “length” and “width” and overrides the “getArea()” method to return the area of the rectangle.
Create a subclass “Box” that extends “Rectangle” and adds a field “height” and overrides the “getArea()” method to return the surface area of the box.
Create an object of the “Shape” class and call the “getArea()” method.
Create an object of the “Rectangle” class and call the “getArea()” method.
Create an object of the “Box” class and call the “getArea()” method.

Create a class “BankAccount” with fields “accountNumber”, “balance”, and “interestRate” and a method “deposit()” that adds an amount to the balance.
Create a subclass “SavingsAccount” that extends “BankAccount” and adds a field “minimumBalance” and a method “withdraw()” that subtracts an amount from the balance.
Create a subclass “FixedDepositAccount” that extends “SavingsAccount” and adds a field “term” and a method “getInterest()” that returns the interest earned on the account.
Create an object of the “BankAccount” class and call the “deposit()” method.
Create an object of the “SavingsAccount” class and call the “deposit()” and “withdraw()” methods.
Create an object of the “FixedDepositAccount” class and call the “deposit()”

# Practice Problem on Hierarchical Inheritance

Create a class “Shape” with a method “calculateArea()” that calculates and returns the area of the shape. Create two subclasses “Rectangle” and “Triangle” that extend “Shape” and implement the “calculateArea()” method.

Create a class “Vehicle” with a method “drive()” that displays a message “Driving a vehicle”. Create two subclasses “Car” and “Bike” that extend “Vehicle” and implement the “drive()” method to display “Driving a car” and “Driving a bike” respectively.

Create a class “Animal” with a method “move()” that displays a message “The animal is moving”. Create two subclasses “Bird” and “Fish” that extend “Animal” and implement the “move()” method to display “The bird is flying” and “The fish is swimming” respectively.

Create a class “BankAccount” with a method “deposit()” that adds money to the account. Create two subclasses “SavingsAccount” and “CheckingAccount” that extend “BankAccount” and implement the “deposit()” method to add interest to the account balance for savings account and display a message for checking account.

Create a class “Person” with a method “speak()” that displays a message “The person is speaking”. Create two subclasses “Student” and “Teacher” that extend “Person” and implement the “speak()” method to display “The student is asking a question” and “The teacher is giving a lecture” respectively.

https://www.studytrigger.com/article/practice-problems-on-inheritance-in-java/