Inheritance

Inheritance

- it is possible to inherit attributes and methods from one class to another. We group the "inheritance concept" into two categories:
  - **subclass** (child) - the class that inherits from another class
  - **superclass** (parent) - the class being inherited from

Inheritance

Mom and daughter

Some properties of mom inherits by her daughter

Types of Inheritance:

- **Single Inheritance:** A class can inherit from only one superclass.
- **Multiple Inheritance:** A class can inherit from more than one superclass.
- **Multilevel Inheritance:** A class can inherit from another class, and then another class can inherit from the derived class.
- **Hierarchical Inheritance:** Multiple classes inherit from a single superclass.


heritence.java > Vehicle
// Base class (superclass)
class Vehicle {
String brand;
int year;
public Vehicle(String brand, int year) {
this.brand = brand;
this.year = year;
}
public void displayInfo() {
System.out.println("Brand: " + brand);
System.out.println("Year: " + year);
}
}
// Derived class 1 (subclass)
class Car extends Vehicle {
int numOfDoors;
public Car(String brand, int year, int numOfDoors) {
super(brand, year);
this.numOfDoors = numOfDoors;
}
public void displayCarInfo() {
displayInfo(); // Accessing the method from the superclass
System.out.println("Number of Doors: " + numOfDoors);
}
}
// // Derived class 2 (subclass)
class Bicycle extends Vehicle {
int numOfGears;
public Bicycle(String brand, int year, int numOfGears) {
super(brand, year);
this.numOfGears = numOfGears;
}
public void displayBicycleInfo() {
displayInfo(); // Accessing the method from the superclass
System.out.println("Number of Gears: " + numOfGears);
}
}
public class Inheritence {
Run | Debug
public static void main(String[] args) {
Car myCar = new Car("Toyota", 2022, 4);
Bicycle myBicycle = new Bicycle("Giant", 2022, 6);
// Displaying information using subclass methods
System.out.println("Car Information:");
myCar.displayCarInfo();
System.out.println("Use Parent Class function.....!");
myCar.displayInfo();
System.out.println("\nBicycle Information:");
myBicycle.displayBicycleInfo();
System.out.println("Use Parent Class function.....!");
myBicycle.displayInfo();
}
}

Task.

- Task 2: Employee Inheritance
  - Create a base class Employee with attributes like name and employeeID. Implement a constructor to initialize these attributes. Create a subclass Manager that inherits from the Employee class. The Manager class should have an additional attribute department and a method to display the manager's information.