**OBJECT-ORIENTED PROGRAMMING**

|  |  |
| --- | --- |
| Lab 13 | |
| **Topic** | Multilevel Inheritance and polymorphism |
| **Objective** | inheritance polymorphism and its types. |

**Task: Zoo Management System**

Develop a Zoo Management System in C++ that utilizes different types of inheritance and demonstrates polymorphism. Design the following classes: **Animal**, **Mammal**, **Bird**, and **Fish**. The **Mammal**, **Bird**, and **Fish** classes should inherit publicly from the **Animal** class. Additionally, create a **Zoo** class that contains a collection of animals.

**Base Class: Animal**

* **Private Member Variables:**
  + **name** (string): to store the name of the animal.
  + **type** (string): to represent the type of animal (e.g., Mammal, Bird, Fish).
* **Public Member Functions:**
  + Default constructor: Initializes **name** as "Unknown" and **type** as "Animal".
  + Parameterized constructor: Accepts parameters for **name** and **type** and initializes the **Animal**.
  + **virtual void displayInfo()**: Displays the name and type of the animal.

**Derived Class 1: Mammal**

* **Inherits publicly from Animal.**
* **Additional Private Member Variables:**
  + **furColor** (string): to store the color of the mammal's fur.
* **Additional Public Member Functions:**
  + Default constructor: Calls the base class default constructor and sets **furColor** to "Not specified".
  + Parameterized constructor: Accepts parameters for **name**, **type**, and **furColor** and initializes the **Mammal**.
  + **void displayInfo()**: Overrides the base class method to display mammal-specific information.

**Derived Class 2: Bird**

* **Inherits publicly from Animal.**
* **Additional Private Member Variables:**
  + **wingSpan** (double): to store the wingspan of the bird.
* **Additional Public Member Functions:**
  + Default constructor: Calls the base class default constructor and sets **wingSpan** to 0.0.
  + Parameterized constructor: Accepts parameters for **name**, **type**, and **wingSpan** and initializes the **Bird**.
  + **void displayInfo()**: Overrides the base class method to display bird-specific information.

**Derived Class 3: Fish**

* **Inherits publicly from Animal.**
* **Additional Private Member Variables:**
  + **scaleColor** (string): to store the color of the fish's scales.
* **Additional Public Member Functions:**
  + Default constructor: Calls the base class default constructor and sets **scaleColor** to "Not specified".
  + Parameterized constructor: Accepts parameters for **name**, **type**, and **scaleColor** and initializes the **Fish**.
  + **void displayInfo()**: Overrides the base class method to display fish-specific information.

**Zoo Class:**

Create a **Zoo** class that contains a collection of animals. Demonstrate polymorphism by storing objects of different derived classes in the same collection.

* **Private Member Variable:**
  + **animals** (Animal\*): to store a collection of animal objects.
* **Public Member Functions:**
  + **void addAnimal(Animal\* newAnimal)**: Adds an animal to the zoo's collection.
  + **void displayZooInfo()**: Displays information about all the animals in the zoo.

**Main Function:**

* Create objects of the **Mammal**, **Bird**, and **Fish** classes.
* Create a **Zoo** object and add the created animals to it.
* Display information about the animals using polymorphism.

**Task Requirements:**

* Utilize different types of inheritance (public inheritance for Mammal, Bird, Fish from Animal).
* Implement polymorphism by using virtual functions and demonstrating the ability to store objects of derived classes in a collection (vector) of the base class type.

**Task2: Vehicle Rental System**

Create a simplified Vehicle Rental System in C++ to practice different types of inheritance and polymorphism. Design the following classes: **Vehicle**, **Car**, **Motorcycle**, and **Truck**. The **Car**, **Motorcycle**, and **Truck** classes should inherit publicly from the **Vehicle** class. Additionally, create a **RentalAgency** class that manages a fleet of vehicles.

**Base Class: Vehicle**

* **Private Member Variables:**
  + **brand** (string): to store the brand of the vehicle.
  + **model** (string): to store the model of the vehicle.
* **Public Member Functions:**
  + Default constructor: Initializes **brand** as "Unknown" and **model** as "Not specified".
  + Parameterized constructor: Accepts parameters for **brand** and **model** and initializes the **Vehicle**.
  + **virtual void displayInfo()**: Displays the brand and model of the vehicle.

**Derived Class 1: Car**

* **Inherits publicly from Vehicle.**
* **Additional Private Member Variables:**
  + **numDoors** (int): to store the number of doors in the car.
* **Additional Public Member Functions:**
  + Default constructor: Calls the base class default constructor and sets **numDoors** to 4.
  + Parameterized constructor: Accepts parameters for **brand**, **model**, and **numDoors** and initializes the **Car**.
  + **void displayInfo()**: Overrides the base class method to display car-specific information.

**Derived Class 2: Motorcycle**

* **Inherits publicly from Vehicle.**
* **Additional Private Member Variables:**
  + **numWheels** (int): to store the number of wheels on the motorcycle.
* **Additional Public Member Functions:**
  + Default constructor: Calls the base class default constructor and sets **numWheels** to 2.
  + Parameterized constructor: Accepts parameters for **brand**, **model**, and **numWheels** and initializes the **Motorcycle**.
  + **void displayInfo()**: Overrides the base class method to display motorcycle-specific information.

**Derived Class 3: Truck**

* **Inherits publicly from Vehicle.**
* **Additional Private Member Variables:**
  + **cargoCapacity** (double): to store the cargo capacity of the truck.
* **Additional Public Member Functions:**
  + Default constructor: Calls the base class default constructor and sets **cargoCapacity** to 0.0.
  + Parameterized constructor: Accepts parameters for **brand**, **model**, and **cargoCapacity** and initializes the **Truck**.
  + **void displayInfo()**: Overrides the base class method to display truck-specific information.

**RentalAgency Class:**

Create a **RentalAgency** class that contains a collection of vehicles. Demonstrate polymorphism by storing objects of different derived classes in the same collection.

* **Private Member Variable:**
  + **vehicles** (Vehicle\*): to store a collection of vehicle objects.
* **Public Member Functions:**
  + **void addVehicle(Vehicle\* newVehicle)**: Adds a vehicle to the rental agency's collection.
  + **void displayRentalInfo()**: Displays information about all the vehicles in the rental agency.

**Main Function:**

* Create objects of the **Car**, **Motorcycle**, and **Truck** classes.
* Create a **RentalAgency** object and add the created vehicles to it.
* Display information about the vehicles using polymorphism.

**Task Requirements:**

* Utilize different types of inheritance (public inheritance for Car, Motorcycle, Truck from Vehicle).
* Implement polymorphism by using virtual functions and demonstrating the ability to store objects of derived classes in a collection (vector) of the base class type.