

**Subject Name: Object Oriented Programming Using C++**

**Subject Code: BCA-301 N**

**Subject Topic: Introduction to Inheritance**

**Abhishek Dwivedi**

Assistant Professor

Department of Computer Application

UIET, CSJM University, Kanpur

# Inheritance in C++

- Inheritance is the capability of one class to acquire properties and characteristics from another class. The class whose properties are inherited by other class is called the **Parent** or **Base** or **Super** class. And, the class which inherits properties of other class is called **Child** or **Derived** or **Sub** class.
- Inheritance makes the code reusable. When we inherit an existing class, all its methods and fields become available in the new class, hence code is reused.

# Purpose of Inheritance in C++

- Code Reusability
- Method Overriding (Runtime Polymorphism)
- Use of Virtual Keyword

## Why and when to use inheritance?

- Consider a group of vehicles. You need to create classes for Bus, Car and Truck. The methods `fuelAmount()`, `capacity()`, `applyBrakes()` will be same for all of the three classes. If we create these classes avoiding inheritance then we have to write all of these functions in each of the three classes as shown in below figure:

### Class Bus

```
fuelAmount()  
capacity()  
applyBrakes()
```

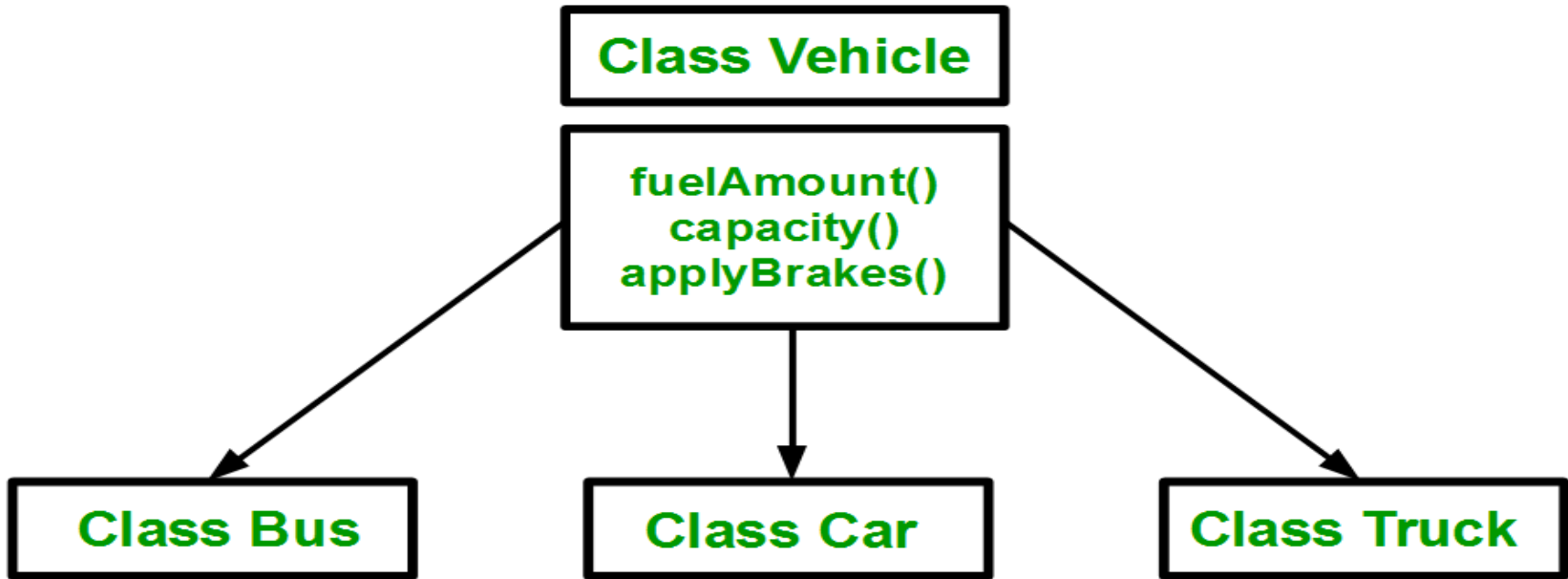
### Class Car

```
fuelAmount()  
capacity()  
applyBrakes()
```

### Class Truck

```
fuelAmount()  
capacity()  
applyBrakes()
```

- You can clearly see that above process results in duplication of same code 3 times. This increases the chances of error and data redundancy. To avoid this type of situation, inheritance is used. If we create a class Vehicle and write these three functions in it and inherit the rest of the classes from the vehicle class, then we can simply avoid the duplication of data and increase re-usability. Look at the below diagram in which the three classes are inherited from vehicle class.



- Using inheritance, we have to write the functions only one time instead of three times as we have inherited rest of the three classes from base class(Vehicle).

# Implementing inheritance in C++

- For creating a sub-class which is inherited from the base class we have to follow the below syntax.

- **Syntax:**

```
class subclass_name : access_mode base_class_name
{
    //body of subclass
};
```

- Here, **subclass\_name** is the name of the sub class, **access\_mode** is the mode in which you want to inherit this sub class for example: public, private etc. and **base\_class\_name** is the name of the base class from which you want to inherit the sub class.

# Example

```
class Parent
{
    public:
        int id_p;
};
```

**//Base Class**

```
class Child : public Parent
```

**// Sub class inheriting from Base Class(Parent)**

```
{
    public:
        int id_c;
};
```

```
void main()
```

```
{
    Child obj1; // An object of class child has all data members and member functions of class parent

    obj1.id_c = 7;
    obj1.id_p = 91;
    cout << "Child id is " << obj1.id_c << endl;
    cout << "Parent id is " << obj1.id_p << endl;
    getch();
}
```



# References:

- [www.studytonight.com](http://www.studytonight.com)
- [www.tutorialpoint.com](http://www.tutorialpoint.com)
- [www.geeksforgeeks.org](http://www.geeksforgeeks.org)
- “Object oriented programming in C++” Robert Lafore
- “Object oriented programming with C++”, E.Balagurusamy