Inheritance in C++

By:
freeCSIT

C++ course with PDF Notes

OOPs in C++

Classes and objects are the two main aspects of object-oriented programming.

Features of OOP

1. Classes
2. Object
3. Inheritance
4. Polymorphism
5. Encapsulation

OOPs in C++

Classes and objects are the two main aspects of object-oriented programming.

Features of OOP

1. Classes
2. Object
3. Inheritance
4. Polymorphism
5. Encapsulation

OOPs in C++

Classes and objects are the two main aspects of object-oriented programming.

Features of OOP

3. Inheritance
4. Polymorphism
5. Encapsulation

OOPs in C++

Classes and objects are the two main aspects of object-oriented programming.

Features of OOP

3 Inheritance

Inheritance is one of the key principles of Object-Oriented Programming (OOP). It allows a class (called the derived class) to inherit properties and behavior (data members and member functions) from another class (called the base class). This promotes code reuse and helps in creating hierarchical relationships between classes.

OOPs in C++

Classes and objects are the two main aspects of object-oriented programming.

Features of OOP

3 Inheritance

Access Specifiers in Inheritance

The access specifiers (public, protected, and private) of the base class control how the members of the base class are inherited in the derived class:

Base Class Member	public Inheritance	protected Inheritance	private Inheritance
public	Remains public	Becomes protected	Becomes private
protected	Remains protected	Remains protected	Becomes private
private	Not inherited	Not inherited	Not inherited

OOPs in C++

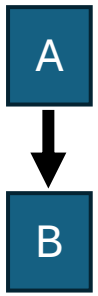
Classes and objects are the two main aspects of object-oriented programming.

Features of OOP

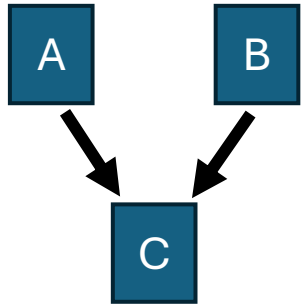
Types

3 Inheritance

1 Single Inheritance

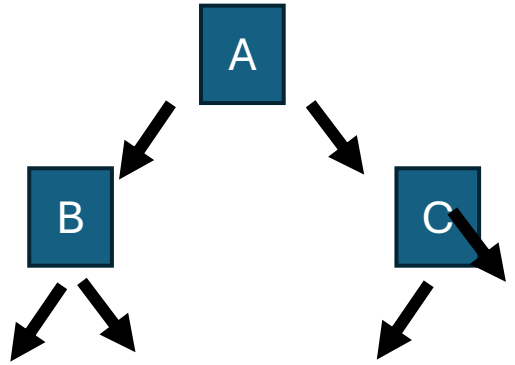


2 Multiple Inheritance



3 Multilevel Inheritance

4 Hierarchical Inheritance



5 Hybrid Inheritance

Combination of one two types

Thanks for watching

Please Subscribe