Anonymous classes in C++

Trie Data Structure using smart pointer and OOP in C++

C++ program to compare two Strings using Operator Overloading

Derived Data Types in C++

Go vs C++

Pointers and References in C++

fill in C++ STL

Deque vs Vector in C++ STL

Types of Literals in C/C++ with Examples

Strings in C++ and How to Create them?

Conditional or Ternary

Operator (?:) in C/C++

C++ Programming Basics

not1 and not2 function templates in C++ STL with Examples

C++ | asm declaration

Enum Classes in C++ and Their Advantage over Enum DataType

How to Insert an element at a specific position in an Array in C++

C++ Program to print an Array using Recursion

Count of distinct remainders when N is divided by all the numbers from the range [1, N]

Difference Between

Constructor and Destructor in C++

How to erase an element from a vector using erase() and reverse_iterator?

How to implement our own Vector Class in C++?

Difference between Inheritance and Polymorphism

Introduction to C++ Programming Language

OpenCV | Hands on Image Contrast

Difference between Abstraction and Encapsulation in C++

ios eof() function in C++ with Examples

return

statement in C/C++ with Examples

OpenCV | Saving an Image

ios rdstate() function in C++ with Examples

fill() function in C++ STL with examples

Hmm. We having trouble finding th site.

We can't connect to t server at googleads.g.doublecl

If that address is cor here are three other you can try:

- Try again later.
- Check your netw connection.
- If you are connec but behind a fire check that Firefo permission to ac the Web.

Anonymous classes in C++

Anonymous class is a class which has no name given to it. C++ supports this feature.

- These classes cannot have a constructor but can have a destructor.
- These classes can neither be passed as arguments to functions nor can be used as return values from functions.

Examples to illustrate Anonymous Classes

1. **Creating single object of Anonymous Class :** In the first Example, Anonymous class is created with object name obj1. The scope of the obj1 is throughout the program. So, we can access this into the main function. In main, using obj1, a call is given to member functions of the anonymous class.

```
// CPP program to illustrate
// concept of Anonymous Class
#include <iostream>
using namespace std;

// Anonymous Class : Class is not having any name
class
{
    // data member
    int i;
public:
    void setData(int i)
    {
        // this pointer is used to differentiate
        // between data member and formal argument.
        this->i = i;
    }
    void print()
    {
        cout << "Value for i : " << this->i << endl;
    }

} obj1;     // object for anonymous class

// Driver function
int main()
{
    obj1.setData(10);
    obj1.print();
    return 0;
}
```

Output :

```
Value for i : 10
```

2. **Creating two objects of Anonymous Class :** In the Second example, we have created two objects obj1 and obj2 for Anonymous class and given a call to member functions of the class. The scope of the obj1 and obj2 is through out the program. Likewise, we can create multiple objects for an anonymous class.

```cpp
// CPP program to illustrate
// concept of Anonymous Class
#include <iostream>
using namespace std;

// Anonymous Class : Class is not having any name
class
{
    // data member
    int i;
public:
    void setData(int i)
    {
        // this pointer is used to differentiate
        // between data member and formal argument.
        this->i = i;
    }
    void print()
    {
        cout << "Value for i : " << this->i << endl;
    }

} obj1, obj2;    // multiple objects for anonymous class

// Driver function
int main()
{
    obj1.setData(10);
    obj1.print();

    obj2.setData(20);
    obj2.print();
    return 0;
}
```

Output :

```
Value for i : 10
Value for i : 20
```

3. **Restricting the scope of Anonymous class :** To restrict the scope of the objects for the anonymous class, we can take a help of typedef. In the third example, by using **typedef** we can give a convenient name to class and use that name we have created multiple objects obje1 and obj2 for the anonymous class. Here we can control the scope of the obj1 and obj2 objects, which are inside the main function.

```cpp
// CPP program to illustrate
// concept of Anonymous Class
// by scope restriction
#include<iostream>
using namespace std;

// Anonymous Class : Class is not having any name
typedef class
{
    // data member
    int i;
public:
    void setData(int i)
    {
        // this pointer is used to differentiate
        // between data member and formal argument.
        this->i = i;
    }
    void print()
    {
        cout << "Value for i :" << this->i << endl;
    }

} myClass;      // using typedef give a proper name

// Driver function
int main()
{
    // multiple objects
    myClass obj1, obj2;
    obj1.setData(10);
    obj1.print();

    obj2.setData(20);
    obj2.print();
    return 0;
}
```

Output :

```
Value for i : 10
Value for i : 20
```

Recommended Posts:

Trivial classes in C++

C++ Classes and Objects

Local Classes in C++

Nested Classes in C++

C++ Stream Classes Structure

Storage Classes in C++ with Examples

File Handling through C++ Classes

Virtual functions in derived classes

Pure Virtual Functions and Abstract Classes in C++

Catching base and derived classes as exceptions

Enum Classes in C++ and Their Advantage over Enum DataType

Difference between Virtual function and Pure virtual function in C++

cin get() in C++ with Examples

Difference between Increment and Decrement Operators

AlankarShelar
He completed his master program in parallel computing Currently He is working as Assistant Professor in Textile and Engineering Institute Ichalkaranji He is well known among students to teach the different programming languages

If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please Improve this article if you find anything incorrect by clicking on the "Improve Article" button below.

Article Tags : C++ C++-Class and Object

Practice Tags : CPP

3

2

To-do Done

Based on **1** vote(s)

Feedback/ Suggest Improvement | Add Notes | Improve Article

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments