

Anonymous classes in C++

Vector of
Vectors in C++
STL with
Examples

Which C++
libraries are
useful for
competitive
programming?

Array of
Vectors in C++
STL

Pi(π) in C++
with Examples

Program to
calculate
Electricity Bill

Speed up Code
executions
with help of
Pragma in
C/C++

Modulo
Operator (%) in
C/C++ with
Examples

Setting up a
C++

Competitive

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

How to find the
Entry with
largest Value
in a C++ Map

Writing code
faster during
Competitive
Programming
in C++

Role of
SemiColon in
various
Programming
Languages

Learn C++
Programming
Step by Step -
A 20 Day
Curriculum!

Default values
in a Map in
C++ STL

Priority queue
of pairs in C++
with ordering
by first and
second
element

Program to
print half
Diamond star
pattern

How to flatten
a Vector of
Vectors or 2D
Vector in C++



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

in a String
using
unordered_map
in C++

std::greater in
C++ with
Examples

Top 10
Programming
Languages for
Blockchain
Development

Implementing
upper_bound()
and
lower_bound()
for Ordered Set
in C++

Operator
Overloading
'<<' and '>>'
operator in a
linked list
class

Finding
Median of
unsorted Array
in linear time
using C++ STL

C/C++
program for
calling main()
in main()

Program to
find frequency
of each



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

Default value
of Vector in
C++ STL

Important
functions of
STL
Components in
C++

new vs
malloc() and
free() vs delete
in C++

Average of
Cubes of first
N natural
numbers

Queries for
elements
having values
within the
range A to B in
the given index
range using
Segment Tree



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

Anonymous classes in C++

08-03-2018

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
```



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

```

 {
    // 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 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;
    }
}

```

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

```

    } 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 obj1 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 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

```

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !


```
obj1.setData(10);  
obj1.print();  
  
obj2.setData(20);  
obj2.print();  
return 0;  
}
```

Output :


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

Attention reader! Don't stop learning now. Get hold of all the important DSA concepts with the **DSA Self Paced Course** at a student-friendly price and become industry ready.

Ad



#1 Kids Online Coding Class
One-on-One Online Classes From Home, Curriculum
Specially Designed for Kids 6-14

 code.whitehatjr.com/Fr...

Recommended Posts:

[C++ Classes and Objects](#)

[Trivial classes in C++](#)

[Local Classes in C++](#)

[Nested Classes in C++](#)

[Storage Classes in C++ with Examples](#)

[C++ Stream Classes Structure](#)

[File Handling through C++ Classes](#)

[Virtual functions in derived classes](#)

[How to add reference of an object in C++](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

Pure Virtual Functions and Abstract Classes in C++

Enum Classes in C++ and Their Advantage over Enum Data Type

lvalues references and rvalues references in C++ with Examples



AlankarShelar

Check out this Author's [contributed articles](#).

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](#)



4

2

☐ To-do ☐ Done

Based on 1 vote(s)

Feedback/ Suggest Improvement

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

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !



5th Floor, A-118,
Sector-136, Noida, Uttar Pradesh - 201305
feedback@geeksforgeeks.org

COMPANY

About Us
Careers
Privacy Policy
Contact Us

LEARN

Algorithms
Data Structures
Languages
CS Subjects
Video Tutorials

PRACTICE

Courses
Company-wise
Topic-wise
How to begin?

CONTRIBUTE

Write an Article
Write Interview Experience
Internships
Videos

@geeksforgeeks, Some rights reserved



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !