



Pointers to Objects and Arrow Operator in CPP | C++ Tutorials for Beginners #51



[Overview](#) [Q&A](#) [Downloads](#) [Announcements](#)

Pointers to Objects and Arrow Operator in CPP | C++ Tutorials for Beginners #51

In this tutorial, we will discuss pointers to objects and arrow operator in C++

Pointer to objects in C++

As discussed before pointers are used to store addresses of variables which have data types like int, float, double etc. But pointer can also store the address of an object. An example program is shown below to demonstrate the concept of pointer to objects.

```
#include<iostream>
using namespace std;
```

```
class Complex{
    int real, imaginary;
public:
    void getData(){
        cout<<"The real part is "<< real<<endl;
        cout<<"The imaginary part is "<< imaginary<<endl;
    }

    void setData(int a, int b){
        real = a;
        imaginary = b;
    }

};

int main(){
    Complex *ptr = new Complex;
    (*ptr).setData(1, 54); is exactly same as
    (*ptr).getData(); is as good as

    return 0;
}
```

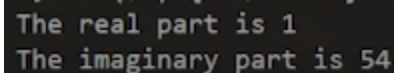
Code Snippet 1: Pointer to objects Example Program 1

As shown in a code snippet 1,

1. We created a class “Complex”, which contains two private data members “real” and “imaginary”.

2. The class “complex” contains two member functions “getdata” and “setdata”
 3. The Function “setdata” will take two parameters and assign the values of parameters to the private data members “real” and “imaginary”
 4. The Function “getdata” will print the values of private data members “real” and “imaginary”
 5. In the main program object is created dynamically by using the “new” keyword and its address is assigned to the pointer “ptr”
 6. The member function “setdata” is called using the pointer “ptr” and the values “1, 54” are passed.
 7. The member function “getdata” is called using the pointer “ptr” and it will print the values of data members.
- The main thing to note here is that we called the member function with pointers instead of object but still it will give same result because pointer is pointing to the address of that object.

The output of the following program is shown below,



```
The real part is 1
The imaginary part is 54
```

Figure 1: Pointer to Objects Program 1 Output

Arrow Operator in C++

Another example program for the pointer to Objects and the use of the “Arrow” Operator is shown below.

```
#include<iostream>
using namespace std;

class Complex{
    int real, imaginary;
public:
    void getData(){
        cout<<"The real part is "<< real<<endl;
        cout<<"The imaginary part is "<< imaginary<<endl;
    }
};
```

```
    }

    void setData(int a, int b){
        real = a;
        imaginary = b;
    }

};

int main(){
    Complex *ptr = new Complex;
    ptr->setData(1, 54);
    ptr->getData();

    // Array of Objects
    Complex *ptr1 = new Complex[4];
    ptr1->setData(1, 4);
    ptr1->getData();
    return 0;
}
```

Code Snippet 2: Pointer to Objects with Arrow Operator Example Program 2

As shown in code snippet 2,

1. We created a class “Complex”, which contains two private data members “real” and “imaginary”.
2. The class “complex” contains two member functions “getdata” and “setdata”
3. The Function “setdata” will take two parameters and assign the values of parameters to the private data

members “real” and “imaginary”

4. The Function “getdata” will print the values of private data members “real” and “imaginary”
5. In the main program object is created dynamically by using the “new” keyword and its address is assigned to the pointer “ptr”
6. The member function “setdata” is called using the pointer “ptr” with the arrow operator “->” and the values “1, 54” are passed.
7. The member function “getdata” is called using the pointer “ptr” with the arrow operator “->” and it will print the values of data members.
8. Array of objects is created dynamically by using the “new” keyword and its address is assigned to the pointer “ptr1”
9. The member function “setdata” is called using the pointer “ptr1” with the arrow operator “->” and the values “1, 4” are passed.
10. The member function “getdata” is called using the pointer “ptr1” with the arrow operator “->” and it will print the values of data members.

The main thing to note here is that we called the member function with pointers by using arrow operator “->” instead of the dot operator “.” but still it will give the same results.

The output of the following program is shown below,

```
The real part is 1
The imaginary part is 54
The real part is 1
The imaginary part is 4
```

Figure 2: Pointer to Objects Program 2 Output

[Previous](#)[Next](#)



CodeWithHarry

Copyright © 2022 CodeWithHarry.com

