



Revisiting Pointers: new and delete Keywords in CPP | C++ Tutorials for Beginners #50



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

Revisiting Pointers: new and delete Keywords in CPP | C++ Tutorials for Beginners #50

In this tutorial, we will discuss pointers and new, delete keywords in C++

Pointers in C++

Pointers are variables that are used to store the address. Pointers are created using “*”. An example program of pointers is shown below

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

```
int main(){
    // Basic Example
    int a = 4;
    int* ptr = &a;
    cout<<"The value of a is "<<*(ptr)<<endl;

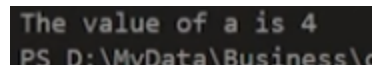
    return 0;
}
```

Code Snippet 1: Pointer Example Program 1

As shown in a code snippet 1,

1. We created an integer variable “a” and assign the value “4” to it
2. We created an integer pointer “ptr” and assign the address of variable “a”
3. And printed the value at the address of pointer “ptr”

The output of the following program is shown below,



```
The value of a is 4
PS D:\MyData\Business\c
```

Figure 1: Pointer Program 1 Output

As shown in figure 1, we get the output value “4” because pointer “ptr” is pointing to the variable “a” and the value of the variable “a” is “4” that is why we get the output “4”.

New Keyword

Another example program for pointers and the use of a “new” keyword is shown below.

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

```
int main(){

    float *p = new float(40.78);
    cout << "The value at address p is " << *(p) << endl;

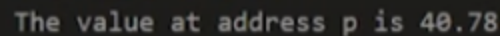
    return 0;
}
```

Code Snippet 2: Pointer Example Program 2

As shown in code snippet 2,

1. We created a float pointer “p” and dynamically created a float which has value “40.78” and assigned that value to pointer “p”
2. And printed the value at the address of pointer “p”

The output of the following program is shown below,



```
The value at address p is 40.78
```

Figure 2: Pointer Program 2 Output

As shown in figure 2, we get the output value “40.78” because pointer “p” is pointing to an address whose value is “40.78”.

Another example program for pointers array and the use of a “new” keyword with an array is shown below.

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

int main(){

    int *arr = new int[3];
```

```
arr[0] = 10;
arr[1] = 20;
arr[2] = 30;
cout << "The value of arr[0] is " << arr[0] << endl;
cout << "The value of arr[1] is " << arr[1] << endl;
cout << "The value of arr[2] is " << arr[2] << endl;

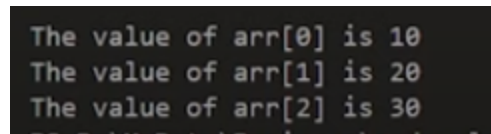
return 0;
}
```

Code Snippet 3: Pointer Example Program 3

As shown in a code snippet 3,

1. We created an integer pointer “arr” and dynamically created an array of size three which is assigned to the pointer “arr”
2. The values “10”, “20”, and “30” are assigned to the “1”, “2”, and “3” indexes of an array
3. And printed the value at the array indexes “1”, “2”, and “3”

The output of the following program is shown below,



```
The value of arr[0] is 10
The value of arr[1] is 20
The value of arr[2] is 30
```

Figure 3: Pointer Program 2 Output

As shown in figure 3, we get the output values “10”, “20”, and “30”.

Delete Keyword

Another example program for pointers array and the use of the “delete” keyword with an array is shown below.

```
#include<iostream>
```

```
using namespace std;

int main(){

    int *arr = new int[3];
    arr[0] = 10;
    arr[1] = 20;
    arr[2] = 30;
    delete[] arr;
    cout << "The value of arr[0] is " << arr[0] << endl;
    cout << "The value of arr[1] is " << arr[1] << endl;
    cout << "The value of arr[2] is " << arr[2] << endl;

    return 0;
}
```

Code Snippet 4: Pointer Example Program 4

As shown in code snippet 4,

1. We created an integer pointer “arr” and dynamically created an array of size three which is assigned to the pointer “arr”
2. The values “10”, “20”, and “30” are assigned to the “1”, “2”, and “3” indexes of an array
3. Before printing the values we used the “delete” keyword
4. And printed the value at the array indexes “1”, “2”, and “3”

The output of the following program is shown below,

```
The value of arr[0] is 15339192
The value of arr[1] is 15335616
The value of arr[2] is 30
```

Figure 4: Pointer Program 2 Output

As shown in figure 2, we get the garbage value in the output instead of “10”, “20”, and “30” because we have used “delete” keyword before printing the values due to which the space used by an array gets free and we get the garbage value in return.

[Previous](#)[Next](#)**CodeWithHarry**

Copyright © 2022 CodeWithHarry.com

