



## C++ Objects Memory Allocation & using Arrays in Classes | C++ Tutorials for Beginners #23



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

## C++ Objects Memory Allocation & using Arrays in Classes | C++ Tutorials for Beginners #23

In this tutorial, we will discuss objects memory allocation and using arrays in C++

### Objects Memory Allocation in C++

The way memory is allocated to variables and functions of the class is different even though they both are from the same class.

The memory is only allocated to the variables of the class when the object is created. The memory is not allocated to the variables when the class is declared. At the same time, single variables can have different values for different objects, so every object has an individual copy of all the variables of the class. But the memory is allocated to the function only once when the class is declared. So the objects don't have individual

copies of functions only one copy is shared among each object.

## Arrays in Classes

Arrays are used to store multiple values of the same type. An array is very helpful when multiple variables are required, instead of making multiple variables one array can be used which can store multiple values. Array stores data in sequential order. An example program to demonstrate the use of arrays in classes is shown below.

```
class Shop
{
    int itemId[100];
    int itemPrice[100];
    int counter;

public:
    void initCounter(void) { counter = 0; }
    void setPrice(void);
    void displayPrice(void);
};
```

### Code Snippet 1: Shop Class

As shown in Code Snippet 1, we created a shop class which has, “itemId[100]” and “itemPrice” as integer array variable and “counter” variable as private class members; and “initCounter” void function, “setPrice” void function, and “displayPrice” void function as public class members. The definitions of these functions are shown below.

```
void Shop ::setPrice(void)
```

```
{
    cout << "Enter Id of your item no " << counter + 1 << endl;
    cin >> itemId[counter];
    cout << "Enter Price of your item" << endl;
    cin >> itemPrice[counter];
    counter++;
}
```

#### **Code Snippet 2: Set Price Function**

As shown in Code Snippet 2, we have created a “setPrice” function. This function will take input for “itemId” and “ItemPrice” from the user at runtime. The value of the counter will be incremented by one every time this function will run.

```
void Shop ::displayPrice(void)
{
    for (int i = 0; i < counter; i++)
    {
        cout << "The Price of item with Id " << itemId[i] << " is " << itemPrice[i] << endl;
    }
}
```

#### **Code Snippet 3: Display Price Function**

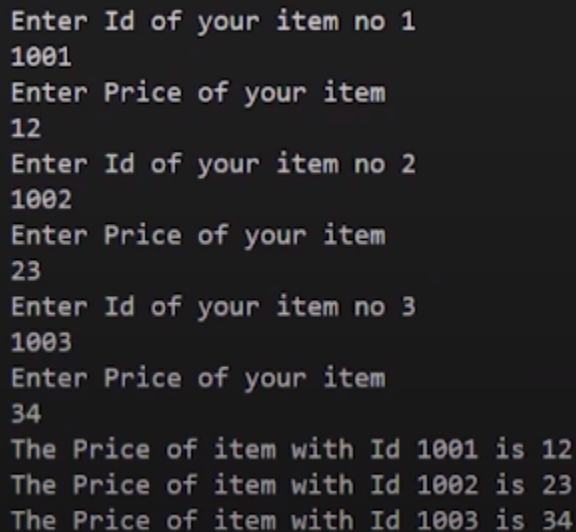
As shown in Code Snippet 3, the “for” loop inside the “displayPrice” function runs till the length of the counter and prints values of the array “itemId” and “ItemPrice”.

```
int main()
```

```
{  
    Shop dukaan;  
    dukaan.initCounter();  
    dukaan.setPrice();  
    dukaan.setPrice();  
    dukaan.setPrice();  
    dukaan.displayPrice();  
    return 0;  
}
```

#### **Code Snippet 4: Main Function**

As shown in Code Snippet 4, we created an object “dukaan” of the shop data type, and the functions “initCounter” is called. The function “setPrice” is called three times. Loops can also be used to call the function multiple times. The “displayPrice” function is also called in the main function. The output of the following program is shown in figure 1.



```
Enter Id of your item no 1  
1001  
Enter Price of your item  
12  
Enter Id of your item no 2  
1002  
Enter Price of your item  
23  
Enter Id of your item no 3  
1003  
Enter Price of your item  
34  
The Price of item with Id 1001 is 12  
The Price of item with Id 1002 is 23  
The Price of item with Id 1003 is 34
```

**Figure 1: Program Output**

As shown in figure 1, for the item 1 we entered the ID “1001” and price “12”; for the item 2 we entered the ID “1002” and price “23”; for the item 3 we entered the ID “1003” and price “34”. The Output of the program has displayed the ID and the price of each item.

**Code as described/written in the video**

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

class Shop
{
    int itemId[100];
    int itemPrice[100];
    int counter;

public:
    void initCounter(void) { counter = 0; }
    void setPrice(void);
    void displayPrice(void);
};

void Shop ::setPrice(void)
{
    cout << "Enter Id of your item no " << counter + 1 << endl;
    cin >> itemId[counter];
    cout << "Enter Price of your item" << endl;
```

```
        cin >> itemPrice[counter];
        counter++;
    }

    void Shop ::displayPrice(void)
    {
        for (int i = 0; i < counter; i++)
        {
            cout << "The Price of item with Id " << itemId[i] << " is " << itemPrice[i] << endl;
        }
    }

    int main()
    {
        Shop dukaan;
        dukaan.initCounter();
        dukaan.setPrice();
        dukaan.setPrice();
        dukaan.setPrice();
        dukaan.displayPrice();
        return 0;
    }
```

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



CodeWithHarry

Copyright © 2022 CodeWithHarry.com

