

Encapsulation in C++ with example

Encapsulation is a process of combining data members and functions in a single unit called class. This is to prevent the access to the data directly, the access to them is provided through the functions of the class. It is one of the popular feature of Object Oriented Programming(OOPs) that helps in **data hiding**.

How Encapsulation is achieved in a class

To do this:

- 1) Make all the data members private.
- 2) Create public setter and getter functions for each data member in such a way that the set function set the value of data member and get function get the value of data member.

Let's see this in an example Program:

Encapsulation Example in C++

Here we have two data members num and ch, we have declared them as private so that they are not accessible outside the class, this way we are hiding the data. The only way to get and set the values of these data members is through the public getter and setter functions.

```
#include<iostream>
using namespace std;
class ExampleEncap{
private:
    /* Since we have marked these data members private,
     * any entity outside this class cannot access these
     * data members directly, they have to use getter and
     * setter functions.
     */
    int num;
    char ch;
public:
    /* Getter functions to get the value of data members.
     * Since these functions are public, they can be accessed
     * outside the class, thus provide the access to data members
     * through them
     */
    int getNum() const {
        return num;
    }
    char getCh() const {
        return ch;
    }
    /* Setter functions, they are called for assigning the values
     * to the private data members.
     */
    void setNum(int num) {
        this->num = num;
    }
}
```

```
void setCh(char ch) {  
    this->ch = ch;  
}  
};  
int main(){  
    ExampleEncap obj;  
    obj.setNum(100);  
    obj.setCh('A');  
    cout<<obj.getNum()<<endl;  
    cout<<obj.getCh()<<endl;  
    return 0;  
}
```

Output:

```
100  
A
```