TUTORIAL 8: ENCAPSULATION

Encapsulation: Hide sensitive content from outer world. Example: "Driver can drive the car but don't know how it works or build"

Main reason to use Encapsulation for code security and make code more protected .

Access specifier used here for change attribute accessibility.

>> Private member of the class can not directly accessible from outside of class. For set the value of this private member/attribute either we can use function/method declared into same class or use constructor, same way to read value of private member we can use same class function/method to get that value. This methodology known as "GET SET" method.

Example

```cpp
#include<iostream>

using namespace std;

class employee{
//private member
private:
   int salary;


//Public member
public:
   string name;


   //method to set salary (private member)
   void set_salary(int x=5000)  //minimum amount
   {
```

```cpp
        salary = x;

    }


    //method to get salary  (private member)

    int get_salary(void)

    {


        return salary;

    }



};



int main()

{


//Object create

    employee AIPL2430;


    //Set name of employee

    AIPL2430.name="Jaydeep Shah";


    //Set salary

    AIPL2430.set_salary(50000);


    //Print data
```

```cpp
    cout<<"Employee name is "<<AIPL2430.name<<" and salary = "<<AIPL2430.get_salary();

    return 0;

}
```