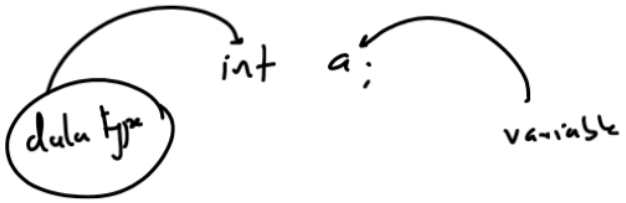
How do we create an object ?



Before creating an object , we first have to create its DATA TYPE and this DATA TYPE for object is ALWAYS a CLASS.

C

```
struct Student
{
    int roll;
    char grade;
    float per;
};
int main()
{
    struct Student S;
    S.roll = 10;
    S.grade = 'A';
    S.per = 61.7;
    printf("%d", S.roll);
}
```

S

r	10
g	'A'
p	61.7

C++

```
class Student
{
    int roll;
    char grade;
    float per;
public:
    void get();
    void show();
};
int main()
{
    Student S;
    S.roll = 10;
}
```

S

r	?
g	?
p	?

object

What are access specifiers in C++ ?

Ans: Access specifiers are keywords which are used to control the accessibility of class members

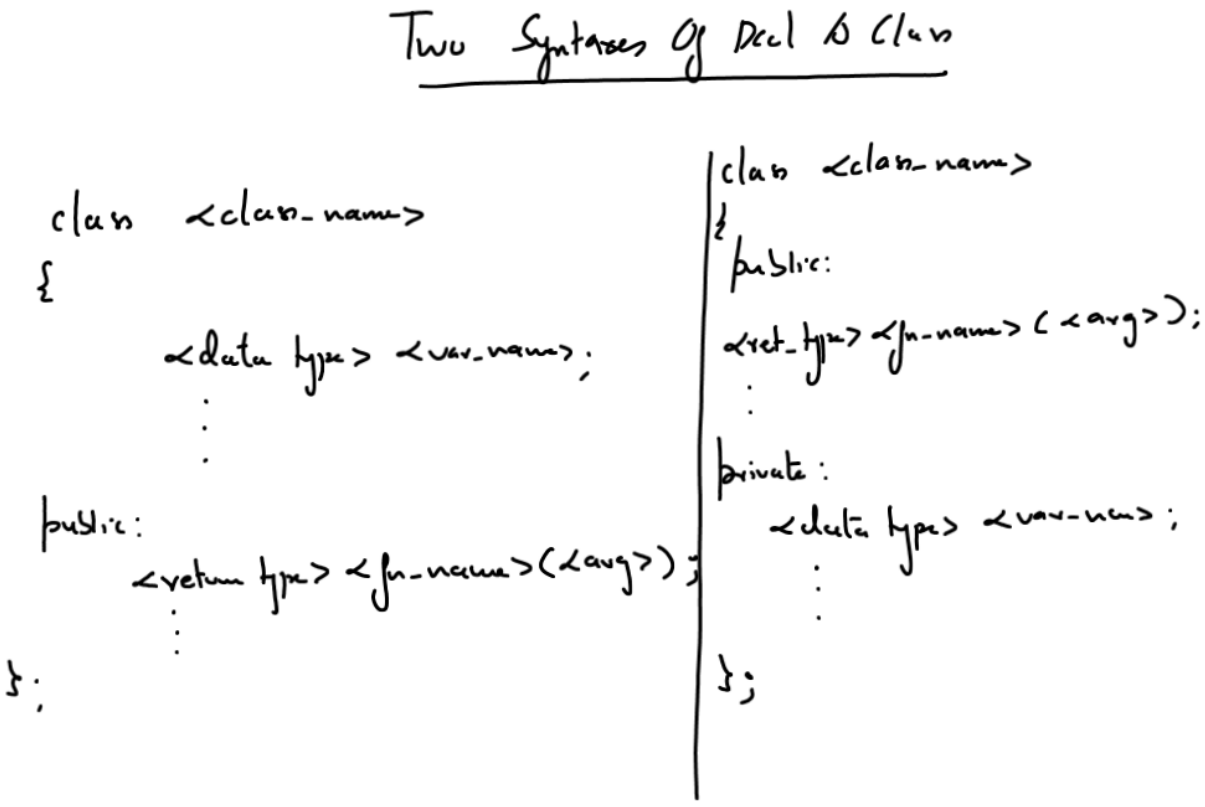
There are 3 access specifiers provided by C++ language and they are:

a. **private**: This is default access specifier of class members in C++ and if any member of the class is private it means that IT CANNOT BE DIRECTLY ACCESSED FROM OUTSIDE THE CLASS.

b. **public**: This access specifier allows a class member to be accessed from outside the class. In other words we can say that public members of a class are directlly accessible from outside the class.

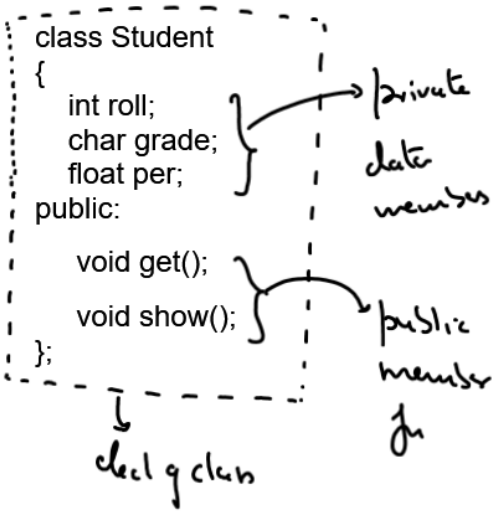
General Rule: We must always declare data member of a class in private section while member functions of a class in public section.

c. **protected**: We will discuss this keyword during the topic INHERITANCE



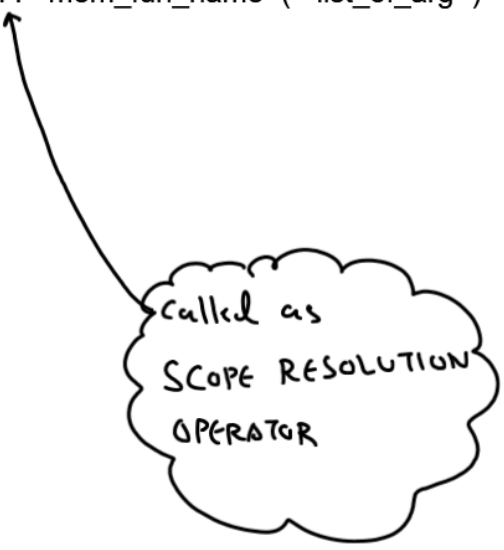
Developing the First Object Oriented Code
==========

```
#include <iostream.h>
#include <conio.h>
class Student
{
   int roll;
   char grade;
   float per;
public:
    void get();
    void show();
};
```

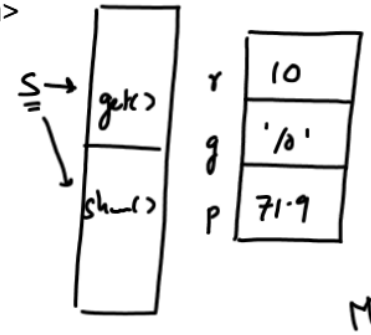


Syntax Of Defining A Member Function

```
<return type> <class_name > : : <mem_fun_name>( <list_of_arg>)
{

    // body

}
```



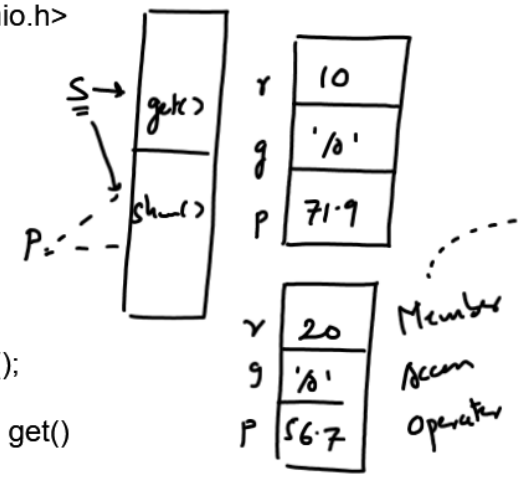
```
#include <iostream.h>
#include <conio.h>

class Student
{
   int roll;
   char grade;
   float per;
public:
    void get();
    void show();
};
void Student:: get()
{
  cout<<"Enter roll,grade and per:";
  cin>>roll>>grade>>per;
}
void Student:: show()
{
  cout<<"roll="<<roll<<",grade="<<grade<<",per="<<per<<endl;
}
```




```
int main()
{
   clrscr();
   Student S;
   S.get();
   S.show();
   getch();
   return 0;
}
```

Member Access Operator

```
#include <iostream.h>
#include <conio.h>

class Student
{
   int roll;
   char grade;
   float per;
public:
    void get();
    void show();
};
void Student:: get()
{
  cout<<"Enter roll,grade and per:";
  cin>>roll>>grade>>per;
}
void Student:: show()
{
  cout<<"roll="<<roll<<",grade="<<grade<<",per="<<per<<endl;
}
```




```
int main()
{
   clrscr();
   Student S,P;
   S.get();
   S.show();
   P.get();
   P.show();
   getch();
   return 0;
}
```

Member Access Operator

```
int main()
{
    clrscr();
    Student S,P;
    S.get();
    P.show();
    getch();
    return 0;
}
```

Will this code compile ?
Ans: Yes

Will this codwe run?
Ans: Yes

Will this code accept data ?
Ans: Yes

Will this code show proper output ?
Ans: No

Qn. Will this code take 2 inputs ?
Ans :yes

Qn. the second input will overwrite first input?
Ans: No

Qn. Will this code show proper output ?
Ans: Yes

```
int main()
{
    clrscr();
    Student S,P;
    S.get();
    P.get();
    S.show();
    P.show();
    getch();
    return 0;
}
```

Qn. Will this code take 2 inputs ?
Ans :yes

Qn. The second input will overwrite first input?
Ans: Yes

Qn. Will this code show proper output ?
Ans: Yes for S , No for P

```
int main()
{
   clrscr();
   Student S,P;
   S.get();
   S.get()
   S.show();
   P.show();
   getch();
   return 0;
}
```

Assignment
=========
Write an object oriented program for accepting 2 integerts from the user , add them and display their result ?

Your next class will be on Wednesday(24th Aug) at 6:30 pm.