

# Scope resolution Operator

- In C++ language, the scope resolution operator is written " ::".
- It is used for following purposes.
  - (a) To define a function outside a class.
  - (b) To access a class's static variables.
  - (c) In case of multiple Inheritance.
  - (d) To access a global variable when there is a local variable with same name.
  - (e) To refer a class inside another class.

for example:-

1 > If the global variable name is same as local variable, the scope resolution operator will be used to call the global variable.

```
#include <iostream.h>
#include <conio.h>
int a; // Global a
{
```

```

clrscr();
int a = 10;           // local a
cout << "Value of global a is : " << ::a;
cout << "In value of local a is : " << a;
getch();
}

```

Output: value of global a is 10 value of local a 10
--

Example 2

It is also used to define a function outside the class.

```

#include <iostream.h>
#include <conio.h>
class jweb
{
public:
void func();           // declaration
};
void jweb::func() // definition outside
                    // class using ::
{
cout << "fun() is called";
}
void main()
{
clrscr();
}

```

```

jPweb j;
j.fun(),
getch(),
}

```

Output: - fun() is called

Example: - It is used to access the static variable of a class.

```

#include <iostream.h>
#include <conio.h>
class abc {
static int a;
public:
static int b;
void fun (int a)
{

```

Note

We can access a class's static var even if there is a local var.

```

cout << "value of static a" << abc::a;
cout << "\n value of local a" << a;
} };
```

```

int abc::a = 1;
int abc::b = 2;
void main ()
{
abc ab;
int a = 3;
ab.fun(a);
cout << "\n value of b" << abc::b;
getch(), }

```