Static Variables and Static Function

Static variable means variable allocate its memory before creating object of class and allocate memory at the time of class loading and non static or instance variable allocate its memory after creating object of class.

```
class ABC
{
  private:
   static int x;      static variable
   int y;             instance variable
   public:

};
int ABC::x=0;
```

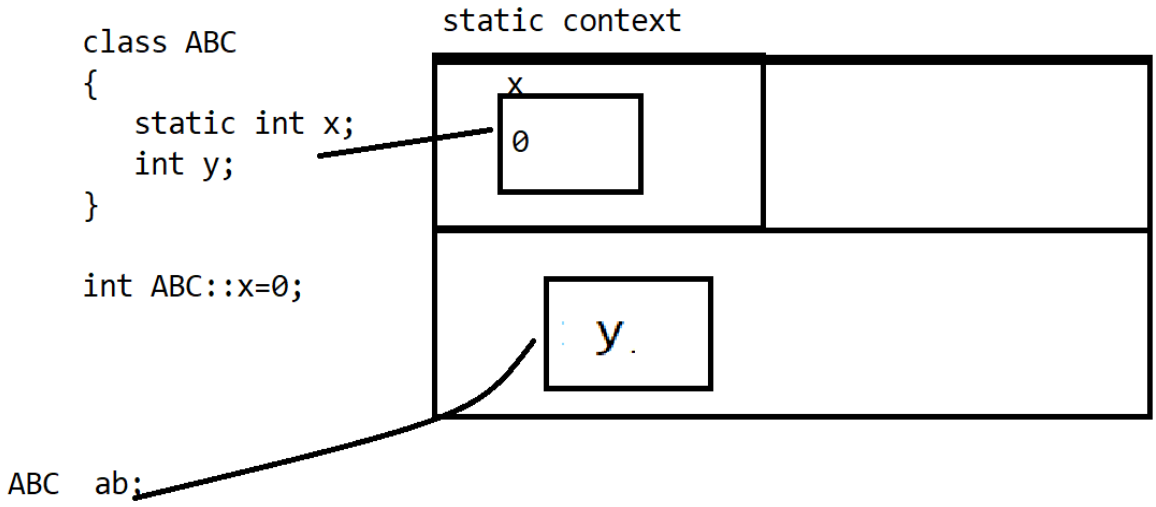
If we want to work with static variable we have the some important points

i) static variable allocate its memory without creating object of class .

Q. who is responsible for allocate memory of static variable ?

Class is responsible for allocate memory of static variable means we can say class loader is responsible for allocate memory of static variable means static variable allocated at class level .

Following Diagram shows how to allocate memory static variable



If we think about above diagram then x is static variable and it is allocated memory when class loaded in memory before instance and y is allocated in memory when we create object of ab class.

2) static variable used by using classname and scope resolution operator in c++

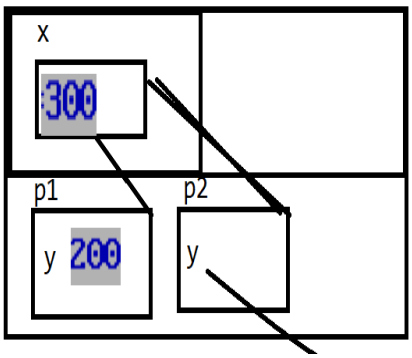
```
#include<iostream.h>
#include<conio.h>
class PQR
{
  private:

  public:
  static int x;_
   int y;
};
int PQR::x=0;
```

```
void main()
{
   clrscr();
   PQR p;
   PQR::x=100;
   cout<<"X is  "<<PQR::x<<"\n";
   p.y=200;
   cout<<"Y is  "<<p.y<<"\n";
   getch();
}
```

3) static variable share its common copy between more than one object of same class and non static variable share its separate copy for every object of class.

```
#include<iostream.h>
#include<conio.h>
class PQR
{ public:
  static int x;
  int y;
};
int PQR::x=0;
```




```
void main()
{
   clrscr();
   PQR p1,p2;
   cout<<"with First  Object"<<"\n";
   p1.y=200;
   p1.x=300;
   cout<<"X is "<<p1.x<<"\n";
   cout<<"Y is "<<p1.y<<"\n";
   cout<<"With Second Object"<<"\n";
   cout<<"X is  "<<p2.x<<"\n"; //300
   cout<<"Y is  "<<p2.y<<"\n";
   getch();
}
```

```
With First Object
X is 300
Y is 200
With Second Object
X is 300
Y is 0 (garbage)
```

Static Function

Static function means function can call by using classname:: operator and static function can work with only static variables. Means we cannot use the non static variable in static function.

```
class ABC
{
  public:
  static int x;
  static void show()
  {
     x=100;
    cout<<"I am static function "<<x<<"\n";
  }
};
int ABC::x=0;
void main()
{
  clrscr();
  ABC::show();
   getch();
}
```

Q. Why non static variable not use in static function?

Because static function can call without using object and non static variable cannot allocate memory without object so if we use the non static variable in static function and if we call the static function using classname without object so compiler not found non static variable in memory so it will generate the error.