

## **Lecture 1: Static Data members and Static member function**



# Today's Agenda

Today we are going to cover -

- Static Data members
- Static member functions
- MCQ Questions

**Let's Get Started-**

# Static Data Members

A data member of a class can be qualified as static. The properties of a static member variable are similar to that of C's static variable. A static data member has certain special characteristics.

They are:-

- It is initialized to zero when the first object of its class is created. No other initialization is permitted.
- Only one copy of that member is created for the entire class and is shared by all the objects of that class, no matter how many objects are created.
- It is visible only within the class, but its lifetime is the entire program.

# Static Data Members

A static variable is normally used to maintain value common to the entire class. For e.g, to hold the count of objects created. Note that the type and scope of each static member variable must be declared outside the class definition. This is necessary because the static data members are stored separately rather than as a part of

# Static Data Members

```
#include <iostream>
using namespace std;
class Demo
{
public:
    static int ABC;
};
```

```
//defining
int Demo :: ABC =10;
```

# Static Data Members

```
int main()
{

cout<<"\nValue of ABC: "<<Demo::ABC;
return 0;

}
```

# Static Member Function

A static member function is a special member function, which is used to access only static data members, any other normal data member cannot be accessed through static member function. Just like static data member, static member function is also a class function; it is not associated with any class object.



# Static Member Function

We can access a static member function with class name, by using following syntax:

```
class_name:: function_name(parameter);
```

# Static Member Function

```
#include <iostream>
using namespace std;

class Demo
{
private:
static int X;
static int Y;

public:
static void Print()
{
cout << "Value of X: " << X << endl;
cout << "Value of Y: " << Y << endl;
}
};
```

# Static Member Function

```
//static data members initializations
```

```
int Demo :: X =10;
```

```
int Demo :: Y =20;
```

```
int main()
```

```
{
```

```
Demo OB;
```

```
//accessing class name with object name
```

```
cout<<"Printing through object name:"<<endl;
```

```
OB.Print();
```

```
//accessing class name with class name
```

```
cout<<"Printing through class name:"<<endl;
```

```
Demo::Print();
```

```
return 0;
```

```
}
```

## MCQ 1

1. If a class contains static variable, then every object of the class has its copy of static variable.

a. True

b. False

# Solution

1. If a class contains static variable, then every object of the class has its copy of static variable.

a. True

**b. False**

**ANSWER: b. False**

**Explanation: Only one copy of static variable is created for entire class and is shared by all the objects of that class.**

## MCQ 2

2. Default value of static variable is \_\_\_\_\_

- a. 0
- b. 1
- c. Garbage value
- d. Compiler dependent

# Solution

2. Default value of static variable is \_\_\_\_\_

**a. 0**

b. 1

c. Garbage value

d. Compiler dependent

**ANSWER: a. 0**

## MCQ 3

3. Static variable in a class is initialized when

- a. every object of the class is created
- b. last object of the class is created
- c. first object of the class is created
- d. No need to initialize static variable



# Solution

3. Static variable in a class is initialized when

a. every object of the class is created

b. last object of the class is created

**c. first object of the class is created**

d. No need to initialize static variable

**ANSWER: c. first object of the class is created**

## MCQ 4

4. Static variable declared in a class are also called\_\_\_\_\_

- a. instance variable
- b. named constant
- c. global variable
- d. class variable

# Solution

4. Static variable declared in a class are also called\_\_\_\_\_

a. instance variable

b. named constant

c. global variable

**d. class variable**

**ANSWER: d. class variable**

## MCQ 5

5. We can initialize a value of static variable of a class only when its object is created. No other initialization is permitted.

a. True

b. False

# Solution

5. We can initialize a value of static variable of a class only when its object is created. No other initialization is permitted.

a. True

**b. False**

**ANSWER: b. False**

**Explanation: It is not mandatory that static variable must be initialized only after first class object is created.**

Any Questions ??  
**Any Questions??**

# Thank You!

**See you guys in next class.**