Objectives

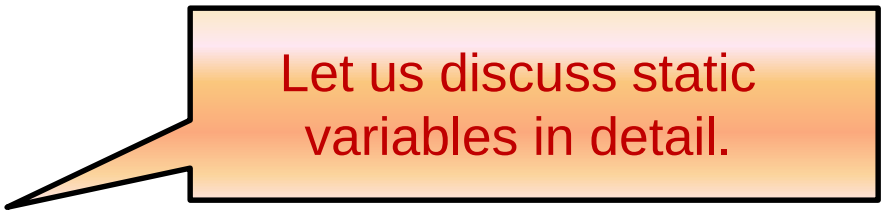
- In this session, you will learn to:
  - Use static variables and static functions

Using Static Variables and Static Functions

AlmaMate

- If you want a variable to retain its value throughout the program, you can declare it as a static variable in your program.
- To manipulate and use the values of static variables, you can define a function as a static function.

Static Variables



Static Variables (Contd.)

AlmaMate

- The keyword `static` means that only one instance of a given variable exists for a class.
- Static variables:
  - Are used to define constants.
  - Can be initialized outside the member function or class definition.
  - Have only one copy of the variable existing in the memory for all the objects of that class.

Static Variables (Contd.)

- The following code shows the declaration and initialization of a static variable:

```
using System;
class Program
{
  static void Main(string[] args)
   {
    StaticExample.ivar = 1;
    Console.WriteLine(StaticExample.
    ivr.ToString());
    Console.ReadLine();
   }
  }
```

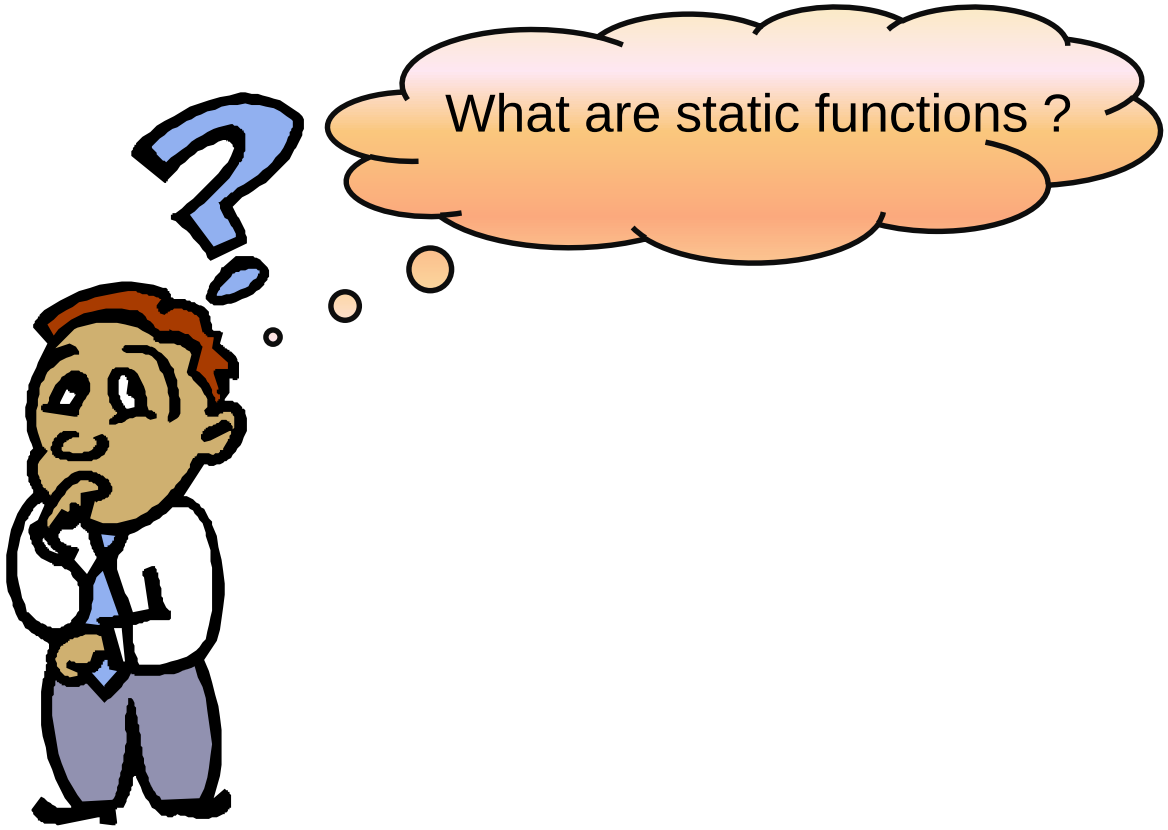
Static variable is initialized outside the class definition

Static Variables (Contd.)

```
public class StaticExample
   {
     public static int ivar;
       StaticExample()
       {
       Console.WriteLine("Object
                  created");
       }
    }
```

Static variable declared

AlmaMate



Static Functions (Contd.)

AlmaMate

Let us understand the concept of static functions.

Static Functions (Contd.)

AlmaMate

- Static functions:
  - Can access only static variables.
  - Can be used to check whether an object of a class has been created.
  - Exist even before the object is created.

Static Functions (Contd.)

AlmaMate

- The following code shows an example of static functions:

```
using System;
public class StaticExample
{
    public static int s;
    public void count()
    {
        s++;
    }
    public static int display()
    {
        return s;
    }
 }
```

Static Functions (Contd.)

AlmaMate

```
class Static
{
  static int Main(string[] args)
 {
    StaticExample s = new StaticExample();
    s.count();
    s.count();
    s.count();
    Console.WriteLine("The value of variable
          is {0}",StaticExample.display());
    return 0;
 }
}
```

Summary

- In this session, you learned that:
  - The static variable retains its value even after the function to which it belongs has been executed.
  - The static functions can access only static variables. Non-static variables cannot be accessed by using static functions.