Practical 11

Aim: Illustrate delegate, Indexer and property in C#

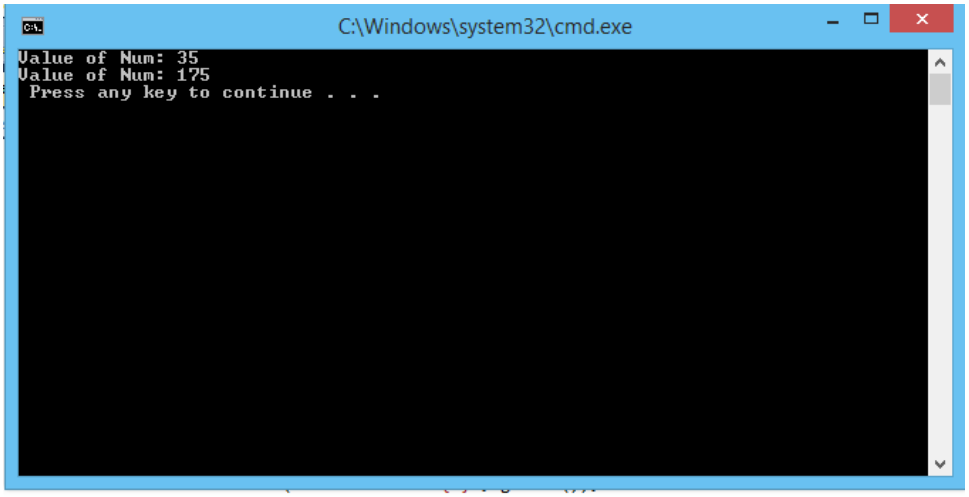
#PROGRAM (Delegate)

```
using System;
delegate int NumberChanger(int n);
namespace DelegateAppl
{
   class TestDelegate
   {
      static int num = 10;
      public static int AddNum(int p)
      {
         num += p;
         return num;
      }

      public static int MultNum(int q)
      {
         num *= q;
         return num;
      }
      public static int getNum()
      {
         return num;
      }
      static void Main(string[] args)
      {
         NumberChanger nc1 = new NumberChanger(AddNum);
         NumberChanger nc2 = new NumberChanger(MultNum);

         //calling the methods using the delegate objects
         nc1(25);
         Console.WriteLine("Value of Num: {0}", getNum());
         nc2(5);
         Console.WriteLine("Value of Num: {0}", getNum());
         Console.ReadKey();
      }
   }
}
```

#OUTPUT



#PROGRAM (Indexer)

```
using System;
namespace IndexerApplication
{
   class IndexedNames
   {
      private string[] namelist = new string[size];
      static public int size = 10;
      public IndexedNames()
      {
         for (int i = 0; i < size; i++)
            namelist[i] = "N. A.";
      }

      public string this[int index]
      {
         get
         {
            string tmp;

            if (index >= 0 && index <= size - 1)
            {
               tmp = namelist[index];
            }
            else
            {
               tmp = "";
            }
```

```
            return (tmp);
        }
        set
        {
            if (index >= 0 && index <= size - 1)
            {
                namelist[index] = value;
            }
        }
    }
    static void Main(string[] args)
    {
        IndexedNames names = new IndexedNames();
        names[0] = "aasf";
        names[1] = "abd";
        names[2] = "fsf";
        names[3] = "fsf";
        for (int i = 0; i < IndexedNames.size; i++)
        {
            Console.WriteLine(names[i]);
        }

        Console.ReadKey();
    }
  }
}
```

#OUTPUT