|  |
| --- |
| **Day 14 Assignment By M.Pallavi**  **10-02-2022** |

|  |
| --- |
| 1.Research and write what is the use of sealed class. WACP to illustrate sealed class. |
| * Sealed method is implemented so that no other class can overthrow it and implement its own method. * Uses:  1. Sealed class is to override all the methods of our class in sub-classes. 2. Sealed class will the extend class functionality. |

|  |  |
| --- | --- |
| 2. Research and write what is the difference between normal properties and auto-implemented properties | |
| Normal properties | auto-implemented properties |
| Normal properties are used to access private variables | Auto properties are used to access private and public variables |
| It needs declaration of set and get methods. | No need to use filed declaration or any code for the get and set methods. |
| In this get and set accessor are optional. | In this get accessor must be specified. |

|  |
| --- |
| Program:  Write a C# program, to illustrate sealed class. |
| using System;  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Author: M.Pallavi  purpose: program on sealed class.  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  namespace pgmonSealedClass  {  sealed class police  {  public static int helpline = 100;  public int secret()  {  return 89;  }    }  internal class Program  {  static void Main(string[] args)  {  police p=new police();  Console.WriteLine(p.secret());  }  }  } |
| Output: |

|  |
| --- |
| 3. WACP to illustrate normal properties |
| Code:  using System;  namespace ConsoleApp1  {  public class Program  {  private int a;  private int b;  public int A  {  set { a = value; }  get { return a; }  }  public int B  {  set { b = value; }  get { return b; }  }  public static void Main(string[] args)  {  Program p = new Program();  p.A = 23;  p.B = 54;  Console.WriteLine("value of a is" + p.A);  Console.WriteLine("value of b is"+ p.B);  }  }  } |
|  |

|  |
| --- |
| Program: 4. WACP to check if the number is prime or not using logic discussed in the class using break; |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace pgmonPrime\_num  { /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Author: program to check given number is prime or not  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int n = 81;    for (int i = 2; i < n; i++)//check for prime  {  if (n % i == 0)  break;  if (i == n)  {  Console.WriteLine("prime number");  }  else  Console.WriteLine("not prime number");  }  Console.ReadLine();  }  }  } |
| output: |

|  |
| --- |
| 5. print numbers from 1 to 3 and skip the numbers divisible by 3HINT : use continue; |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace pgmOnContinue  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Author: M.Pallavi  purpose: WACP to print numbers from 1 to 30 and skip the numbers divisible by 3 using continue;  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int n;  Console.WriteLine("enter n value");  n=Convert.ToInt32(Console.ReadLine());  Console.WriteLine("numbers not divisible by 30 are");  for (int i = 0; i <= n; i++)  {    if (i % 3 == 0)  continue;  Console.Write(i, + "\t");  }  Console.ReadLine();  }  }  } |
|  |

|  |
| --- |
| Program : 6. Find the first number after 1000 which is divisible by 97 by using for loop and break |
| Code:  using System.Collections.Generic;  namespace pgmOnbreak  {  internal class Program  {  static void Main(string[] args)  {  for (int i = 1000; i <= 1097; i++)  {  if (i % 97 == 0)  {  Console.WriteLine(i);  break;  }  }  Console.ReadLine();  }  }  } |
| Output: |