***DAY 10 ASSIGNMENT PRESENTED***

***BY***

***POTUKANUMA JEEVITHA***

|  |
| --- |
| **1.WRITE THE TWO POINTS DISCUSSED ABOUT INHERITANCE IN THE CLASS?** |
| **ANS:** |
| * Inheritance is the process of Re-usability base class methods in the desired class. |
| * Inheritance main goal is: Re-usability and Remove duplicate code. |

|  |
| --- |
| **2.WRITE EXAMPLE CODE FOR:** |
| **A.SINGLE INHERITANCE:-** |
| **CODE:** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace SINGLE\_INHERITANCE  {  class Algebra  {  public int add(int a, int b)  {  return a + b;  }  public int sub(int a, int b)  {  return a - b;  }  }  class TotalMaths : Algebra  {  public int mul(int a, int b)  {  return (a \* b);  }  }  internal class Program  {  static void Main(string[] args)  {  TotalMaths tm = new TotalMaths();  Console.WriteLine(tm.add(7, 8));  Console.WriteLine(tm.sub(78, 30));  Console.WriteLine(tm.mul(8, 7));  Console.ReadLine();  }  }  }  **OUTPUT:** |
| **B.MULTILEVEL INHERITANCE:-** |
| **CODE:** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace multi\_level  {  class Algebra  {  public int add(int a, int b)  {  return a + b;  }  public int sub(int a, int b)  {  return a - b;  }  }  class TotalMaths : Algebra  {  public int mul(int a, int b)  {  return (a \* b);  }  }  class Allsubjects : TotalMaths  {  public string methane()  {  return "CH4";  }  }  internal class Program  {  static void Main(string[] args)  {  Allsubjects obj = new Allsubjects();  Console.WriteLine(obj.add(8, 7));  Console.WriteLine(obj.sub(30, 14));  Console.WriteLine(obj.mul(3, 7));  Console.WriteLine(obj.methane());  Console.ReadLine();  }  }  }  **OUTPUT:** |

|  |
| --- |
| **3.PICTORIALLY REPRESENT 3 TYPES OF INHERITANCE DISCUSSED IN THE CLASS.** |
| **ANS:** |
| **SINGLE INHERITANCE:-** |
| **MULTIPLE INHERITANCE:-** |
| **MULTILEVEL INHERITANCE:-** |

|  |
| --- |
| **4.WHY MULTIPLE INHERITANCE IS NOT SUPPORTED FOR CLASSES IN C#.** |
| **ANS:** |
| C# Compiler is designed not to support multiple inheritance because it causes ambiguity of methods from different base class.  In Multiple inheritance, one class can have more than one superclass and inherit features from all its parent classes. **child class** inherits the features of **parent class 1 and 2.** |
| Due to Diamond problem , but in C# we can solve the Diamond problem with the help of **INTERFACES** in some cases. |

|  |
| --- |
| **5.WHAT IS POLYMORPHISM?** |
| **ANS:** |
| * For reusability with different implementations. * Polymorphism is the ability of an object to take on many forms. |

|  |
| --- |
| **6.WRITE SAMPLE CODE FOR METHOD OVERLOADING?** |
| **ANS:** |
| **CODE:-** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace OVERLOADING  {  class Algebra  {  public int add(int a, int b)  {  return a + b;  }  public int add(int a, int b, int c)  {  return a + b + c;  }  public int add(int a, int b, int c, int d)  {  return a + b + c + d;  }  }  internal class Program  {  static void Main(string[] args)  {  Algebra obj = new Algebra();  Console.WriteLine(obj.add(7, 8));  Console.WriteLine(obj.add(7, 14, 11));  Console.WriteLine(obj.add(11, 30, 14, 3));  Console.ReadLine();  }  }  }  **OUTPUT:** |

|  |
| --- |
| **7.WRITE SAMPLE CODE FOR METHOD OVERRIDING [USING NEW KEY WORD]?** |
| **ANS:** |
| **CODE:-** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace OVERRIDING\_using\_NEW\_keyword  {  class EnglishMessage  {  public void PrintHi()  {  Console.WriteLine("Hi");  }  public void PrintHello()  {  Console.WriteLine("Hello");  }  public void PrintGM()  {  Console.WriteLine("Good Morning");  }  }  class TeluguMessage : EnglishMessage  {  public new void PrintGM()  {  Console.WriteLine("Subhodayam");  }  }  internal class Program  {  static void Main(string[] args)  {  TeluguMessage teluguMessage = new TeluguMessage();  teluguMessage.PrintHi();  teluguMessage.PrintHello();  teluguMessage.PrintGM();  Console.ReadLine();  }  }  }  **OUTPUT:** |

|  |
| --- |
| **8.RESEARCH AND WRITE SAMPLE CODE FOR METHOD OVERRIDING USING VIRTUAL, OVERRIDE KEYWORD?** |
| **ANS:** |
| **CODE:-** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace OVERRIDING\_using\_VIRAL\_\_OVERRIDE\_keyword  {  class EnglishMessage  {  virtual public void PrintHi()  {  Console.WriteLine("Hi");  }  virtual public void PrintHello()  {  Console.WriteLine("Hello");  }  virtual public void PrintGM()  {  Console.WriteLine("Good Morning");  }  }  class TeluguMessage : EnglishMessage  {  override public void PrintGM()  {  Console.WriteLine("Subodhayam");  }  }  internal class Program  {  static void Main(string[] args)  {  TeluguMessage msg = new TeluguMessage();  msg.PrintHi();  msg.PrintHello();  msg.PrintGM();  Console.ReadLine();  }  }  }  **OUTPUT:** |