Day 10 Morning Assignment

By k. sanjay

|  |
| --- |
| 1. Write the two points discussed about inheritance in the class. |
| * Inheritance is the process of reusing of base class methods in the derived class * Inheritance main goal is Re- usability and to remove duplicate code |

|  |
| --- |
| 2. Write example code for:  a. Single inheritance  b. Multi level inheritance |
| 1. Single inheritance   using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day\_10\_mng\_4thfeb\_2022assignment  {  class Addition  {  public int add(int a, int b)  {  return a + b;  }  }  class Multiplication : Addition  {  public int mul(int a, int b)  {  return a \* b;  }  }  internal class Program  {  static void Main(string[] args)  {  Multiplication mt = new Multiplication();  Console.WriteLine(mt.add(10,5));  }  }  } |
| 1. Multi level inheritance   using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day\_10\_mng\_4thfeb\_2022assignment  {  class Addition  {  public int add(int a, int b)  {  return a + b;  }  }  class Multiplication : Addition  {  public int mul(int a, int b)  {  return a \* b;  }  }  class Twooperations : Multiplication  {  public string water()  {  return "H2o";  }  }  internal class Program  {  static void Main(string[] args)  {  Twooperations to = new Twooperations();  Console.WriteLine(to.add(10,5));  Console.WriteLine(to.mul(5,2));  Console.WriteLine(to.water());  }  }  } |

|  |
| --- |
| 3. Pictorially represent 3 types of inheritance discussed  in the class. |
| Single Inheritance Multilevel Inheritance Multiple Inheritance |
| 4. Why multiple inheritance is not supported for classes in  C# |
| C# compiler is designed not to support multiple inheritence because it causes ambiguity of methods from different base class. This is Cause by diamond Shape problems of two classes If two classes B and C inherit from A, and class D inherits from both B and C. ... So., multiple inheritance is not possible in C# |

|  |
| --- |
| 5. What is polymorphism. |
| The word polymorphism means having many forms. In object-oriented programming paradigm, polymorphism is often expressed as 'one interface, multiple functions'.  Polymorphism can be static or dynamic. In static polymorphism, the response to a function is determined at the compile time. In dynamic polymorphism, it is decided at run-time. |

|  |
| --- |
| 6. Write sample code for method overloading |
| class Opearations  {  public int add(int a, int b)  {  return a + b;  }  public int diff(int a , int b , int c)  {  return a - b - c;  }  public int mul(int a , int b, int c, int d)  {  return a \* b \* c \* d;  }  } |

|  |
| --- |
| 7. Write sample code for method overriding  [ using new key word ] |
| class Message  {  public void printHi()  {  Console.WriteLine("HI");  }  public void printHello()  {  Console.WriteLine("Hello");  }  public void PrintGm()  {  Console.WriteLine("Good Morning");  }  }  class Newmessage : Message  {  public new void PrintGm()  {  Console.WriteLine("Subhodhayam");  }  } |

|  |
| --- |
| 8. Research and write sample code for method overriding  using virual, override keyword. |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day\_10\_mng\_4thfeb\_2022assignment  {  class Message  {  public void printHi()  {  Console.WriteLine("HI");  }  public void printHello()  {  Console.WriteLine("Hello");  }  public virtual void PrintGm()  {  Console.WriteLine("Good Morning");  }  }  class Newmessage : Message  {  public override void PrintGm()  {  Console.WriteLine("Subhodhayam");  }  }      internal class Program  {  static void Main(string[] args)  {  Newmessage msg = new Newmessage();  msg.printHi();  msg.printHello();  msg.PrintGm();  Console.ReadLine();    }  }  } |