|  |
| --- |
| **Object – Oriented Programming Assignment**  **By**  **Nanam Vaishnavi**  **04 – Feb - 2022** |

|  |
| --- |
| **1. Write the two points discussed about inheritance in the class.** |
| **Ans:**  **1) Inheritance is a process of reusing base class method in the derived class.**  **2) Inheritance main goal is: Reusability and to remove duplicate code.** |

|  |
| --- |
| **2. Write example code for:**  **a. Single inheritance**  **b. Multi level inheritance** |
| 1. **Single Inheritance Code:** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // Author : Nanam Vaishnavi  // Purpose : Single Inheritance  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace 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(6, 6));  Console.WriteLine(tm.Sub(14, 9));  Console.WriteLine(tm.Mul(4, 7));  Console.ReadLine();  }  }  } |
| **OUTPUT** |
|  |
| 1. **Multi Level Inheritance Code:** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // Author : Nanam Vaishnavi  // Purpose : Multi Level Inheritance  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace 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);  }  }  class Allsubjects : TotalMaths  {  public string benzene()  {  return "c6h6";  }  }  internal class Program  {  static void Main(string[] args)  {  Allsubjects all = new Allsubjects();  Console.WriteLine(all.Add(7, 5));  Console.WriteLine(all.Sub(9, 8));  Console.WriteLine(all.Mul(5, 6));  Console.WriteLine(all.benzene());  Console.ReadLine();  }  }  } |
| **OUTPUT** |
|  |

|  |
| --- |
| **3. Pictorially represent 3 types of inheritance discussed**  **in the class.** |
| 1. **Single Inheritance**   **Super class**  **Sub Class** |
| 1. **Multi-level inheritance**   **Super Class**  **Super Class**  **Sub Class** |
| 1. **Multiple Inheritance**   **Super Class**  **Super Class**  **Sub Class** |

|  |
| --- |
| **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.** |

|  |
| --- |
| **5. What is polymorphism.** |
| **Ans :**  **Ability of an object to take on many forms.** |

|  |
| --- |
| **6. Write sample code for Method Overloading** |
| **CODE** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // Author : Nanam Vaishnavi  // Purpose : Method Overloading  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace Inheritance  {  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 ob = new Algebra();  Console.WriteLine(ob.Add(4, 9));  Console.WriteLine(ob.Add(8, 3, 5));  Console.WriteLine(ob.Add(9, 2, 4, 7);  Console.ReadLine();  }  }  } |
| **OUTPUT** |
|  |

|  |
| --- |
| **7. Write sample code for method overriding**  **[ using new key word ]** |
| **CODE** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // Author : Nanam Vaishnavi  // Purpose : Method Overriding [Use new keyword]  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace Inheritance  {  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("Subodhayam");  }  }  internal class Program  {  static void Main(string[] args)  {  TeluguMessage msg = new TeluguMessage();  msg.PrintHi();  msg.PrintHello();  msg.PrintGM();  Console.ReadLine();  }  }  } |
| **OUTPUT** |
|  |

|  |
| --- |
| **8. Research and write sample code for method overriding**  **using virual, override keyword.** |
| **CODE** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // Author : Nanam Vaishnavi  // Purpose : Method Overriding [use override, virtual keyword]  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace Inheritance  {  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** |
|  |