|  |
| --- |
| **Day 7 Morning Assignments**  **By Manoj yekolla**  **1-Feb-2022** |

|  |
| --- |
| **1. Create Employee class with three variables and two methods**  **ReadEmployee and PrintEmployee and create an object and call methods ?** |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7\_Project1  {  class Employee  {  private int id;  private string name;  private int salary;  public void ReadEmployee()  {  Console.WriteLine("Enter Id Num :");  id =Convert.ToInt32( Console.ReadLine());  Console.WriteLine("Enter Your Name :");  name = Console.ReadLine();  Console.WriteLine("Enter Your Salary :");  salary = Convert.ToInt32(Console.ReadLine());  }  public void PrintEmployee()  {  Console.WriteLine($"Id={id},Name={name},Salary={salary}");  }  }  internal class Program  {  static void Main(string[] args)  {  Employee emp1 = new Employee();  emp1.ReadEmployee();  emp1.PrintEmployee();  Console.ReadLine();  }    }  } |
| Output :  Screenshot (98) |

|  |
| --- |
| **2. Write the 3 def of class and 4 points about object discussed in the class ?** |
| **Class :**  -> A Class Consists Of State and Behaviour ,A Class enables we to create your custom types by Grouping variables of other types ,methods and events.  -> A Class is Group of Variables and Methods.  -> A Class is like a blueprint to Create Objects. |
| **Object :**  -> We Can Create any number of objects.  -> An object is an instance of a Class.  ->Objects are occupy memory.  ->Objects are Reference Type. |

|  |
| --- |
| **3.Pictorially represent class and multiple objects ?** |
| ed2a2bfb-95eb-473f-af7c-aa9f1d4c055e (2) |

**4. Create below classes:**

**1. Customer**

**2. Product**

**3. Seller**

**4. Department**

|  |
| --- |
| 1. WACP Customer Class ? |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project4  {  class Customer  {  private string name;  private string email;  private long mobile;  private string address;  public void ReadCustomer()  {  Console.WriteLine("Enter your Name :");  name = Console.ReadLine();  Console.WriteLine("Enter Your Email :");  email = Console.ReadLine();  Console.WriteLine("Enter Your Mobile Number :");  mobile = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Your Address :");  address = Console.ReadLine();  }  public void PrintCustomer()  {  Console.WriteLine($"Name={name},Email={email},Mobile={mobile},Address={address}");  }  }  internal class Program  {  static void Main(string[] args)  {  Customer cust = new Customer();  cust.ReadCustomer();  cust.PrintCustomer();  Console.ReadLine();  }  }  } |
| Output :  Screenshot (100) |

|  |
| --- |
| 1. WACP to Product Class ? |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project4\_b\_  {  class Product  {  private int id;  private string name;  private string description;  private double price;  public void ReadProduct()  {  Console.WriteLine("Enter Your Product Id : ");  id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("enter your Product name : ");  name= Console.ReadLine();  Console.WriteLine("enter your Product Description :");  description= Console.ReadLine();  Console.WriteLine("enter your Product price :");  price= Convert.ToDouble(Console.ReadLine());  }  public void PrintProduct()  {  Console.WriteLine($"id={id},name={name},Descr={description},price={price}");  }  }  internal class Program  {  static void Main(string[] args)  {  Product Pro=new Product();  Pro.ReadProduct();  Pro.PrintProduct();  Console.ReadLine();  }  }  } |
| Output :  Screenshot (102) |

|  |
| --- |
| **4(c) WACP to print Seller Class ?** |
| Code: |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project\_4\_c\_  {  class Seller  {  private int id;  private string name;  private long mobile;  private string address;  public void ReadSeller()  {  Console.WriteLine("Enter Your seller Id : ");  id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("enter your Seller name : ");  name = Console.ReadLine();  Console.WriteLine("enter your seller mobile :");  mobile = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("enter your Seller Address :");  address = Console.ReadLine();  }  public void PrintProduct()  {  Console.WriteLine($"id={id},name={name},mobile={mobile},address={address}");  }  }  internal class Program  {  static void Main(string[] args)  {  Seller sell=new Seller();  sell.ReadSeller();  sell.PrintProduct();  Console.ReadLine();  }  }  } |
| Output :  Screenshot (104) |

|  |
| --- |
| **4(d) WACP Department Class ?** |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project\_4\_d\_  {  class Department  {  private int id;  private string name;  private long mobile;  private string branch;  public void ReadDepartment()  {  Console.WriteLine("Enter Your Department Id : ");  id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("enter your Department name : ");  name = Console.ReadLine();  Console.WriteLine("enter your Department mobile :");  mobile = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("enter your Department branch :");  branch = Console.ReadLine();  }  public void PrintDepartment()  {  Console.WriteLine($"id={id},name={name},mobile={mobile},branch={branch}");  }  }  internal class Program  {  static void Main(string[] args)  {  Department department = new Department();  department.ReadDepartment();  department. PrintDepartment();  Console.ReadLine();  }  }  } |
| Output :  Screenshot (107) |

|  |
| --- |
| **5. Create Employee class with 3 public variables.**  **Create Employee object and initialize with values while creating object**  **and print the values.** |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project5  {  class Employee  {  public int id;  public string name;  public int salary;  }  internal class Program  {  static void Main(string[] args)  {  Employee emp1 = new Employee() { id = 1, name = "yekolla", salary = 10000 };  Console.WriteLine($"id={emp1.id},name={emp1.name},salary={emp1.salary}");  Console.ReadLine();  }  }  } |
| Output :  Screenshot (109) |

|  |
| --- |
| **6.Create Employee class as shown below , now create employees array object and initialize with 4 employees ,write code using**  **a. for loop**  **b. foreach loop**  **c. lambda expression.** |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project6  {  class Employee  {  public int id;  public string name;  public int salary;  }  internal class Program  {  static void Main(string[] args)  {  Employee[] employees = new Employee[]  {  new Employee(){id=1,name="manojyek",salary=5000},  new Employee(){id=2,name="murali",salary=7000},  new Employee(){id=3,name="dhawan",salary=15000},  new Employee(){id=4,name="nithesh",salary=4000},  };  //for loop  for (int i=0;i<employees.Length;i++)  {  Console.WriteLine($"id={employees[i].id},name={employees[i].name},salary={employees[i].salary}");    }  //forEach  foreach(var e in employees)  {  Console.WriteLine($"id={e.id},name={e.name},salary={e.salary}");  }  //lamda experssion  employees.ToList().ForEach(e => Console.WriteLine($"id={e.id},name={e.name},salary={e.salary}"));  Console.ReadLine();  }  }  } |
| Output :  Screenshot (112) |

|  |
| --- |
| **7. For the above project,**  **write code to print employees who is getting salary >=5000 using**  **for loop ,foreach loop ,lambda expression** |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project7  {  class Employee  {  public int id;  public string name;  public int salary;  }  internal class Program  {  static void Main(string[] args)  {  Employee[] employees = new Employee[]  {  new Employee(){id=1,name="manojyek",salary=5000},  new Employee(){id=2,name="murali",salary=7000},  new Employee(){id=3,name="dhawan",salary=15000},  new Employee(){id=4,name="nithesh",salary=4000},  };  //for loop  for (int i = 0; i < employees.Length; i++)  {  if(employees[i].salary>=5000)  Console.WriteLine($"id={employees[i].id},name={employees[i].name},salary={employees[i].salary}");  }  //forEach    foreach (var e in employees)    {  if (e.salary >= 5000)  Console.WriteLine($"id={e.id},name={e.name},salary={e.salary}");  }  //lamda experssion  employees.ToList().Where(e=>e.salary>=5000).ToList().ForEach(e => Console.WriteLine($"id={e.id},name={e.name},salary={e.salary}"));    Console.ReadLine();  }  }  } |
| Output :  Screenshot (114) |

|  |
| --- |
| **8. Similar to 6 and 7 projects create list of Customer an Product Arrays**  **and practice for, foreach and lambda expression ?** |
| 1. Customer Program |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project\_8\_a\_  {  class Customer  {  public int id;  public string name;  public long mobile;  public string email;  }  internal class Program  {  static void Main(string[] args)  {  Customer[] cust = new Customer[]  {  new Customer(){id=1,name="manojyek",mobile=9010203045,email="manoj@gmail.com"},  new Customer(){id=2,name="murali",mobile=8994939923,email="murali33@gmail.com"},  new Customer(){id=3,name="dhawan",mobile=7899938545,email="dhawan@gmail.com"},  new Customer(){id=4,name="nithesh",mobile=9049599599,email="nith@gmail.com"},  };  //for loop  for (int i = 0; i < cust.Length; i++)  {  Console.WriteLine($"id={cust[i].id},name={cust[i].name},mobile={cust[i].mobile},email={cust[i].email}");  }  //forEach  foreach (var e in cust)  {  Console.WriteLine($"id={e.id},name={e.name},mobile={e.mobile},email={e.email}");  }  //lamda experssion  cust.ToList().ForEach(e => Console.WriteLine($"id={e.id},name={e.name},mobile={e.mobile},email={e.email}"));  Console.ReadLine();  }  }  } |
| Output :  Screenshot (116) |

|  |
| --- |
| **8(b) Product Array Program using for ,foreach ,loop ?** |
| Code : |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7\_Project8\_b\_  {  class Product  {  public int id;  public string name;  public double price;  public string colour;  }  internal class Program  {  static void Main(string[] args)  {  Product[] prod = new Product[]  {  new Product(){id=1,name="manojyek",price=10000,colour="red"},  new Product(){id=2,name="murali",price=20000,colour="blue"},  new Product(){id=3,name="dhawan",price=30000,colour="green"},  new Product(){id=4,name="nithesh",price=40000,colour="yellow"},  };  //for loop  for (int i = 0; i < prod.Length; i++)  {  Console.WriteLine($"id={prod[i].id},name={prod[i].name},price={prod[i].price},colour={prod[i].colour}");  }  //forEach  foreach (var e in prod)  {  Console.WriteLine($"id={e.id},name={e.name},price={e.price},colour={e.colour}");  }  //lamda experssion  prod.ToList().ForEach(e => Console.WriteLine($"id={e.id},name={e.name},price={e.price},colour={e.colour}"));  Console.ReadLine();  }  }  } |
| Output :  Screenshot (118) |