|  |
| --- |
| Day 6 Assignment  BY  Nanam VAishnavi  01- 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;  namespace Day7project1  {  internal class Program  {  class Employee  {  public int id;  public string name;  public int salary;  public void ReadEmployee()  {  Console.WriteLine("Enter ID: ");  id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Name: ");  name = Console.ReadLine();  Console.WriteLine("Enter Salary: ");  salary = Convert.ToInt32(Console.ReadLine());  }  public void PrintEmployee()  {  Console.WriteLine($"Id = {Id}, Name = {Name}, Salary = {salary}");  }    }  static void Main(string[] args)  {  Employee emp1 = new Employee();  emp1.ReadEmployee();  emp1.PrintEmployee();  Console.ReadLine();  }  }  } |

|  |
| --- |
| **OUTPUT :** |
|  |

|  |  |
| --- | --- |
| |  | | --- | | 1. **Write the 3 def of class and 4 points about object discussed in the class.** | |
| **Class :**   * A class is a group pf variables and methods. * A class is like a design to create objects. * A class consists of state and behaviour.   **Object :**   * An object is an instance of a class. * We can create any number of objects. * Objects occupied memory. * Objects are reference. |

t

|  |
| --- |
| **3. Pictorially represent class and multiple objects.** |
| **cars**  **Properties**  **methods**  **Green**  **Ford**  **Gasoline**  **Red**  **Toyota**  **Electricity** |

|  |
| --- |
| **4. Create below classes:**  **1. Customer**  **2. Product**  **3. Seller**  **4. Department** |
| **Customer class** |
| using System;  using System.Collections.Generic;  namespace Customer  {  internal class Customer\_1  {  public int cust\_ID;  public string cust\_Name;  public string cust\_Email;  public void ReadCustomer()  {  Console.WriteLine("Enter Cust\_ID: ");  cust\_ID = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Cust\_Name");  cust\_Name = Console.ReadLine();  Console.WriteLine("Enter Cust\_Email: ");  cust\_Email = Console.ReadLine();  }  public void PrintCustomer()  {  Console.WriteLine($"cust\_ID = {cust\_ID},cust\_Name = {cust\_Name}, cust\_Email = {cust\_Email}");  }  static void Main(string[] args)  {  Customer\_1 cust = new Customer\_1();  cust.ReadCustomer();  cust.PrintCustomer();  Console.ReadLine();  }  }  } |
| **OUTPUT:** |
|  |
| **Product class** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Product  {  internal class Product1  {  public int product\_ID;  public string product\_name;  public int price;  public void ReadProduct()  {  Console.WriteLine("Enter product\_ID: ");  product\_ID = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter product\_Name");  product\_name = Console.ReadLine();  Console.WriteLine("Enter price: ");  price = Convert.ToInt32(Console.ReadLine());  }  public void PrintProduct()  {  Console.WriteLine($"product\_ID = {product\_ID},product\_Name = {product\_name}, price = {price}");  }  static void Main(string[] args)  {  Product1 product = new Product1();  product.ReadProduct();  product.PrintProduct();  Console.ReadLine();  }  }  } |

|  |
| --- |
| **OUTPUT** |
|  |
| |  | | --- | | **Seller class** | | using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Seller  {  internal class Seller\_1  {  public int id;  public string name;  public string email;  public void ReadSeller()  {  Console.WriteLine("Enter ID: ");  id = Convert.ToInt16(Console.ReadLine());  Console.WriteLine("Enter Name: ");  name = Console.ReadLine();  Console.WriteLine("Enter Email: ");  email = Console.ReadLine();  }  public void PrintSeller()  {  Console.WriteLine($"ID ={id} , Name ={name}, email{email} ");  }  static void Main(string[] args)  {  Seller\_1 sell = new Seller\_1();  sell.ReadSeller();  sell.PrintSeller();  Console.ReadLine();  }  }  } | | **OUTPUT** | |
|  |
| **Department** |
| using System;  using System.Collections.Generic;  namespace Department  {  internal class Department1  {  public int dept\_id;  public string dept\_name;  public string course\_name;    public void ReadDepartment()  {  Console.WriteLine("Enter Dept\_ID: ");  dept\_id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Dept\_Name: ");  dept\_name = Console.ReadLine();  Console.WriteLine("Enter Course\_Name: ");  course\_name = Console.ReadLine();  }  public void PrintDepartment()  {  Console.WriteLine($"Dept\_ID= {dept\_id}, Dept\_Name= {dept\_name}, Course\_Name= {course\_name}");  }    static void Main(string[] args)  {  Department1 dept = new Department1();  dept.ReadDepartment();  dept.PrintDepartment();    Console.ReadLine();  }  }  } |
| **OUTPUT** |
|  |

|  |
| --- |
| **5. Create Employee class with 3 public variables. Create Employee object and initialize with values while creating object and print the values**. |
| using System;  using System.Collections.Generic;  namespace Day7Project3  {  class Employee  {  public int empid;  public string name;  public int salary;    }  internal class Program  {  static void Main(string[] args)  {  Employee emp = new Employee();  emp.empid = 26;  emp.name = "Vaishu";  emp.salary = 30000;  //{empid =26; name ="vaishu, age= 21, salary=30000"};  Console.WriteLine($"empid= {emp.empid}, name={emp.name}, salary={emp.salary}");  Console.ReadLine();  }  }  } |
| **OUTPUT :** |
|  |

|  |
| --- |
| **6. Create Employee class as shown below:**  **class Employee**  **{**  **public int id;**  **public string name;**  **public int salary;**  **}now create employees array object and initialize with 5 employees write code using**  **a. for loop**  **b. foreach loop**  **c. lambda expression.** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project4  {  class Employee  {  public int id;  public string name;  public int salary;  }  internal class Program  {  static void Main(string[] args)  {  Employee[] employee = new Employee[5];    employee[0] = new Employee() {id =1,name="Vaishu",salary=5000};  employee[1] = new Employee() { id = 2, name = "Rathika",salary= 4000 };  employee[2] = new Employee() { id = 3, name = "Prashanth", salary = 1000 };  employee[3] = new Employee() { id = 4, name = "Aruna", salary = 2000 };  employee[4] = new Employee() { id = 5, name = "Pavan", salary = 8000 };  Console.WriteLine("=========================================================");  //forloop  for (int i=0; i<employee.Length; i++)  {  Console.WriteLine($"id={employee[i].id},name={employee[i].name}, salary={employee[i].salary}");  } Console.WriteLine("=========================================================");  //foreach loop  foreach(var e in employee)  {  Console.WriteLine($"id ={e.id}, name={e.name}, salary={e.salary}");  } Console.WriteLine("=========================================================");  //Lambda Expression  employee.ToList().ForEach(e => Console.WriteLine($"id{e.id}, name={e.name}, salary={e.salary}"));  Console.ReadLine();  }  };  } |
| **OUTPUT :** |
|  |

|  |
| --- |
| **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 Day7Project5  {  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day7Project4  {  class Employee  {  public int id;  public string name;  public int salary;  }  internal class Program  {  static void Main(string[] args)  {  Employee[] employee = new Employee[5];  employee[0] = new Employee() { id = 1, name = "Vaishu", salary = 50000 };  employee[1] = new Employee() { id = 2, name = "Rathika", salary = 40000};  employee[2] = new Employee() { id = 3, name = "Prashanth", salary = 10000 };  employee[3] = new Employee() { id = 4, name = "Aruna", salary = 20000 };  employee[4] = new Employee() { id = 5, name = "Pavan", salary = 80000};  Console.WriteLine("==========================================================================");  //forloop  for (int i = 0; i < employee.Length; i++)  {  if(employee[i].salary >= 30000)  Console.WriteLine($"id={employee[i].id},name={employee[i].name}, salary={employee[i].salary}");  }  Console.WriteLine("==========================================================================");  //foreach loop  foreach (var e in employee)  {  if(e.salary>=20000)  Console.WriteLine($"id ={e.id}, name={e.name}, salary={e.salary}");  }  Console.WriteLine("==========================================================================");  //Lambda Expression  employee.ToList().Where(e=>e.salary>=40000).ToList().ForEach(e => Console.WriteLine($"id{e.id}, name={e.name}, salary={e.salary}"));  Console.ReadLine();  }  };  }  } |
| **OUTPUT :** |
|  |

|  |
| --- |
| 8. Similar to 6 and 7 projects create list of Customer and Product Arrays and practice for, foreach and lambda expression |
| **Customer Code :** |
| class Customer  {  public int id;  public string name;  public string email;  }    internal class Program  {  static void Main(string[] args)  {  Customer[] customer = new Customer[6];  {  customer[0] = new Customer() { id = 1, name = "Vaishu", email = "vaish@gmail.com" };  customer[1] = new Customer() { id = 2, name = "Sindhu", email = "abcd@gmail.com" };  customer[2] = new Customer(){ id = 3, name = "Pavan", email = "pavan@gmail.com" };  customer[3] = new Customer() { id = 4, name = "Mouni", email = "Mouni25@gmail.com" };  customer[4] = new Customer() { id = 5, name = "swathi", email = "swathi\_6@gmail.com" };  }  Console.WriteLine("=======================================================================================");  // for loop  for(int i=0;i<customer.Length;i++)  {  Console.WriteLine($"id={customer[i].id}, name={customer[i].name}, email={customer[i].email}");  }  Console.WriteLine("=======================================================================================");  // foreach loop  foreach (var c in customer)  {  Console.WriteLine($"id={c.id}, name={c.name}, email={c.email}");  }  Console.WriteLine("=======================================================================================");  // Lambda Expression  customer.ToList().ToList().ForEach(c => Console.WriteLine($"id={c.id},name={c.name},salary{c.email}"));  Console.ReadLine();  }  }  } |
| **OUTPUT :** |
|  |
| **Product Code :** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Product  {  class Product  {  public int id;  public string name;  public int price;  }  internal class Program  {  static void Main(string[] args)  {  Product[] product = new Product[]  {  new Product() { id = 1, name ="Kavya", price =955},  new Product() { id = 2, name ="Aravind",price=255},  new Product() { id = 3, name ="Sushma",price=10},  new Product() { id = 4, name ="vaishnavi",price=64},  new Product() { id = 5, name ="Jaya",price=40}  };  //using for loop  for (int i = 0; i < product.Length; i++)  {  if (product[i].price >= 100)  Console.WriteLine($"id={product[i].id},name={product[i].name},Price={product[i].price}");  }  Console.WriteLine("========================================================");  //using foreach  foreach (var e in product)  {  if (e.price >= 100)  Console.WriteLine($"id={e.id},name={e.name},Price{e.price}");  }  Console.WriteLine("========================================================");  //using foreach  //using lamda expression  product.ToList().Where(e => e.price >= 100).ToList().ForEach(e => Console.WriteLine($"id={e.id},name={e.name},Price{e.price}"));  Console.ReadLine();  }  }  } |
|  |