**LINQ Assignments**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace LINQAssignment

{

class Employee

{

public int EmployeeID { get; set; }

public string FirstName { get; set; }

public string LastName { get; set; }

public string Title { get; set; }

public DateTime DOB { get; set; }

public DateTime DOJ { get; set; }

public string City { get; set; }

}

class Program

{

public static List<Employee> empList = new List<Employee>

{

new Employee() {EmployeeID = 1001,FirstName = "Malcolm",LastName = "Daruwalla",Title = "Manager",DOB = DateTime.Parse("1984-01-02"),DOJ = DateTime.Parse("2011-08-09"),City = "Mumbai"},

new Employee() {EmployeeID = 1002,FirstName = "Asdin",LastName = "Dhalla",Title = "AsstManager",DOB = DateTime.Parse("1984-08-20"),DOJ = DateTime.Parse("2012-7-7"),City = "Mumbai"},

new Employee() {EmployeeID = 1003,FirstName = "Madhavi",LastName = "Oza",Title = "Consultant",DOB = DateTime.Parse("1987-11-14"),DOJ = DateTime.Parse("2015-4-12"),City = "Pune"},

new Employee() {EmployeeID = 1004,FirstName = "Saba",LastName = "Shaikh",Title = "SE",DOB = DateTime.Parse("6/3/1990"),DOJ = DateTime.Parse("2/2/2016"),City = "Pune"},

new Employee() {EmployeeID = 1005,FirstName = "Nazia",LastName = "Shaikh",Title = "SE",DOB = DateTime.Parse("3/8/1991"),DOJ = DateTime.Parse("2/2/2016"),City = "Mumbai"},

new Employee() {EmployeeID = 1006,FirstName = "Suresh",LastName = "Pathak",Title = "Consultant",DOB = DateTime.Parse("11/7/1989"),DOJ = DateTime.Parse("8/8/2014"),City = "Chennai"},

new Employee() {EmployeeID = 1007,FirstName = "Vijay",LastName = "Natrajan",Title = "Consultant",DOB = DateTime.Parse("12/2/1989"),DOJ = DateTime.Parse("6/1/2015"),City = "Mumbai"},

new Employee() {EmployeeID = 1008,FirstName = "Rahul",LastName = "Dubey",Title = "Associate",DOB = DateTime.Parse("11/11/1993"),DOJ = DateTime.Parse("11/6/2014"),City = "Chennai"},

new Employee() {EmployeeID = 1009,FirstName = "Amit",LastName = "Mistry",Title = "Associate",DOB = DateTime.Parse("8/12/1992"),DOJ = DateTime.Parse("12/3/2014"),City = "Chennai"},

new Employee() {EmployeeID = 1010,FirstName = "Sumit",LastName = "Shah",Title = "Manager",DOB = DateTime.Parse("4/12/1991"),DOJ = DateTime.Parse("1/2/2016"),City = "Pune"},

};

public static void DisplayAll()

{

var displayAllEmployees = from Emp in empList select Emp;

Console.WriteLine("----------Display All Employees----------------\n");

foreach (var e in displayAllEmployees)

{

Console.WriteLine(e.EmployeeID + " " + e.FirstName+" "+e.LastName+" "+e.Title+" "+e.DOB+" "+e.DOJ+" "+e.City);

}

}

public static void NotMumbai()

{

var NotMumbaiList = (from e in empList

where e.City != "Mumbai"

select e);

Console.WriteLine("\n");

Console.WriteLine("-----------List of Employees whose location is not Mumbai-------------\n");

foreach (var e in NotMumbaiList)

{

Console.WriteLine(e.EmployeeID + " " + e.FirstName + " " + e.LastName + " " + e.Title + " " + e.DOB + " " + e.DOJ + " " + e.City);

}

}

public static void TitleName()

{

var TitleList= (from e in empList

where e.Title == "AsstManager"

select e);

Console.WriteLine("\n");

Console.WriteLine("-----------List of Employees whose title is AsstManager-----------\n");

foreach (var e in TitleList)

{

Console.WriteLine(e.EmployeeID + " " + e.FirstName + " " + e.LastName + " " + e.Title + " " + e.DOB + " " + e.DOJ + " " + e.City);

}

}

public static void DisplayLastName()

{

var LastNameList = from e in empList where e.LastName.StartsWith("S") select e;

Console.WriteLine("\n");

Console.WriteLine("-----------List of Employees whose last name starts with 'S'-----------\n");

foreach (var e in LastNameList)

{

Console.WriteLine(e.EmployeeID + " " + e.FirstName + " " + e.LastName + " " + e.Title + " " + e.DOB + " " + e.DOJ + " " + e.City);

}

}

public static void DisplayDOJ()

{

var JoiningDateList = from e in empList where e.DOJ < DateTime.Parse("1/1/2015") select e;

Console.WriteLine("\n");

Console.WriteLine("-----------List of Employees who have joined before 1/1/2015------------\n");

foreach (var e in JoiningDateList)

{

Console.WriteLine(e.EmployeeID + " " + e.FirstName + " " + e.LastName + " " + e.Title + " " + e.DOB + " " + e.DOJ + " " + e.City);

}

}

public static void DisplayDOB()

{

var BirthDate = from e in empList where e.DOB > DateTime.Parse("1/1/1990") select e;

Console.WriteLine("\n");

Console.WriteLine("-----------List of Employees who have Date of birth is after 1/1/1990----------\n");

foreach (var e in BirthDate)

{

Console.WriteLine(e.EmployeeID + " " + e.FirstName + " " + e.LastName + " " + e.Title + " " + e.DOB + " " + e.DOJ + " " + e.City);

}

}

public static void DisplayDesignation()

{

var DesignationShow = (from e in empList where( e.Title== "Consultant" && e.Title=="Associate") select e);

Console.WriteLine("\n");

Console.WriteLine("-----------List of Employees whose designation is Consultant & Associate----------");

foreach (var e in DesignationShow)

{

Console.WriteLine(e.EmployeeID + " " + e.FirstName + " " + e.LastName + " " + e.Title + " " + e.DOB + " " + e.DOJ + " " + e.City);

}

}

public static void DisplayCount()

{

var empCount = (from e in empList select e).Count();

Console.WriteLine("\n");

Console.WriteLine("-----------Total Number of Employees----------\n");

Console.WriteLine(empCount);

}

public static void BelongChennai()

{

var belong = empList.Count(e => e.City == "Chennai");

Console.WriteLine("\n");

Console.WriteLine("-----------Total number of employee belonging to chennai----------\n");

Console.WriteLine(belong);

}

public static void HighestID()

{

var DisplayHighest = empList.Last();

Console.WriteLine("\n");

Console.WriteLine("-----------------Display Highest Employee ID---------------");

Console.WriteLine( DisplayHighest.EmployeeID);

}

public static void DisplayJoinedAfter()

{

var cityList = (from e in empList where e.DOJ < DateTime.Parse("1/1/2015") select e).Count();

Console.WriteLine("\n");

Console.WriteLine("-----------Total Number of employees who have joined after 1/1/2015----------");

Console.WriteLine(cityList);

}

public static void NotAssociate()

{

var DesignationList = (from e in empList

where e.Title!= "Associate"

select e).Count();

Console.WriteLine("\n");

Console.WriteLine("-----------Total Number of employee whose designation is not Associate----------");

Console.WriteLine(DesignationList);

}

public static void BasedOnCity()

{

var displayCity = (from e in empList

group e by new

{

e.City

} into e

select new

{

City = e.Key.City,

Count = e.Count()

}).ToList();

Console.WriteLine("-----------Total Number of employee based on City--------------");

Console.WriteLine( displayCity.Count);

}

public static void BasedOnCityAndTitle()

{

var displayCityandTitle = (from e in empList

group e by new

{

e.City,

e.Title

} into e

select new

{

City = e.Key.City,

Title = e.Key.Title,

Count = e.Count()

}).ToList();

Console.WriteLine("-----------Total Number of employee based on City & Title--------------");

Console.WriteLine( displayCityandTitle.Count);

}

public static void TotalYoungestEmp()

{

var youngest = empList.Min();

Console.WriteLine("The yongest employee" + youngest);

}

static void Main(String[] args)

{

DisplayAll();

NotMumbai();

TitleName();

DisplayLastName();

DisplayDOJ();

DisplayDOB();

DisplayDesignation();

DisplayCount();

BelongChennai();

HighestID();

DisplayJoinedAfter();

NotAssociate();

BasedOnCity();

BasedOnCityAndTitle();

TotalYoungestEmp();

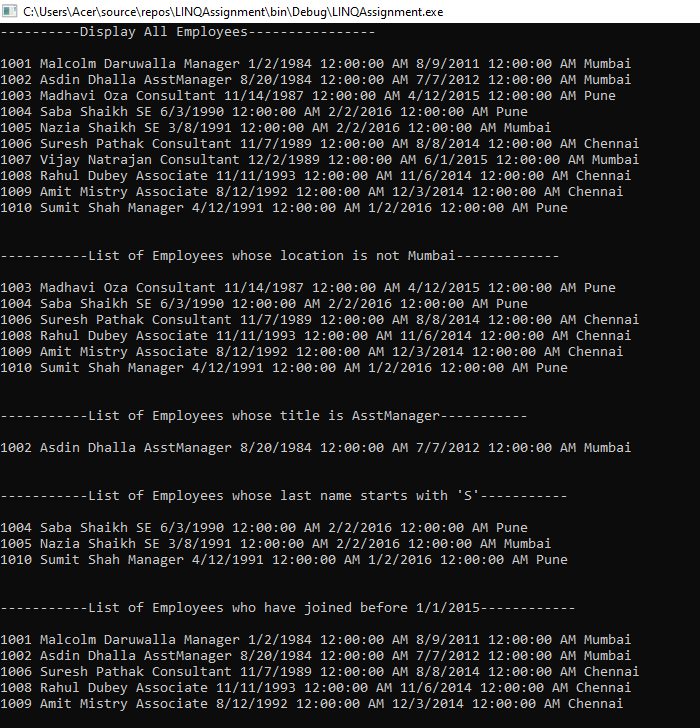
Console.ReadKey();

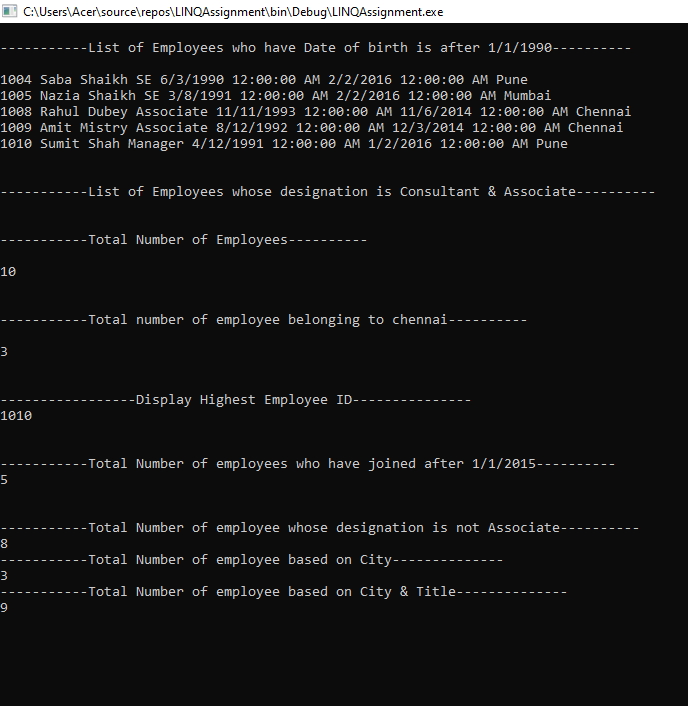
}

}

}

**Output:**

****

****