using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

namespace Cs\_Lesson8

{

#region Part1

// class Car

// {

// public Guid ID { get; set; }

// public string Model { get; set; }

// public string Vendor { get; set; }

// public double Engine { get; set; }

// public Car()

// {

// ID = Guid.NewGuid();

// }

// public Car(string model, string vendor, double engine)

// {

// ID = Guid.NewGuid();

// Model = model;

// Vendor = vendor;

// Engine = engine;

// }

// public override string ToString()

// {

// return $@"

// ID : {ID}

// Model : {Model}

// Vendor : {Vendor}

// Engine : {Engine}

//";

// }

// }

// class CarGallery

// {

// public Car[] Cars { get; set; }

// public CarGallery(int size)

// {

// Cars = new Car[size];

// }

// public Car this[int index] { get => Cars[index]; set => Cars[index] = value; }

// }

// class Matrix

// {

// int[,] data = new int[5, 5];

// public int this[int row, int col]

// {

// get => data[row, col];

// set => data[row, col] = value;

// }

// }

// public class Program

// {

// static void Main(string[] args)

// {

// //var code = Guid.NewGuid();

// //Console.WriteLine(code);

// //code = Guid.NewGuid();

// //Console.WriteLine(code);

// //code = Guid.NewGuid();

// //Console.WriteLine(code);

// //code = Guid.NewGuid();

// //Console.WriteLine(code);

// //Car c1 = new Car("M5", "BMW", 4.8);

// //Car c2 = new Car("La Ferrari", "Ferrari", 5.5);

// //Car c3 = new Car("Huracan", "Lamborghini", 3.7);

// ////Console.WriteLine(c1);

// //CarGallery gallery = new CarGallery(3);

// //gallery[0] = c1;

// //gallery[1] = c2;

// //gallery[2] = c3;

// //for (int x = 0; x < gallery.Cars.Length; x++)

// //{

// // Console.WriteLine(gallery[x]);

// //}

// //Matrix matrix = new Matrix();

// //for (int x = 0; x < 5; x++)

// //{

// // for (int y = 0; y < 5; y++)

// // {

// // matrix[x, y] = (x + 1) \* 10 + y + 1;

// // //matrix[x, y] = new Random().Next(1, 100);

// // //Thread.Sleep(1);

// // }

// //}

// //for (int x = 0; x < 5; x++)

// //{

// // for (int y = 0; y < 5; y++)

// // {

// // Console.Write(matrix[x, y] + " ");

// // }

// // Console.WriteLine();

// //}

// }

// }

#endregion

#region Part2

//abstract class Human

//{

// public string Name { get; set; }

// public string Surname { get; set; }

// public DateTime BirthDate { get; set; }

// public Human()

// {

// }

// public Human(string name, string surname)

// {

// Name = name;

// Surname = surname;

// }

// public Human(string name, string surname, DateTime birthDate)

// : this(name, surname)

// {

// BirthDate = birthDate;

// }

// public void Show()

// {

// Console.WriteLine($"Name : {Name}");

// Console.WriteLine($"Surname : {Surname}");

// Console.WriteLine($"Birth Date : {BirthDate}");

// }

//}

//class Student : Human

//{

// int \_average;

// public Student()

// {

// }

// public Student(string name, string surname, int average)

// : base(name, surname)

// {

// \_average = average;

// }

// public Student(string name, string surname, DateTime birthDate, int average)

// : base(name, surname, birthDate)

// {

// \_average = average;

// }

// public new void Show()

// {

// base.Show();

// Console.WriteLine($"Average : {\_average}");

// }

//}

////sealed class SuperStudent : Student

////{

////}

//abstract class Employee : Human

//{

// double \_salary;

// public Employee(string name, string surname, double salary) : base(name, surname)

// {

// \_salary = salary;

// }

// public Employee(string name, string surname, DateTime birthdate, double salary) : base(name, surname, birthdate)

// {

// \_salary = salary;

// }

//}

//class Manager : Employee

//{

// string \_team;

// public Manager(string name, string surname, DateTime date, double salary, string team)

// : base(name, surname, date, salary)

// {

// \_team = team;

// }

// public void Show()

// {

// base.Show();

// Console.WriteLine($"Team : {\_team}");

// }

//}

//class Programmer : Employee

//{

// string \_language;

// public Programmer(string name, string surname, DateTime date, double salary, string language)

//: base(name, surname, date, salary)

// {

// \_language = language;

// }

// public void Show()

// {

// base.Show();

// Console.WriteLine($"Language : {\_language}");

// }

//}

//class Scientist : Employee

//{

// string \_direction;

// public Scientist(string name, string surname, DateTime date, double salary, string direction)

// : base(name, surname, date, salary)

// {

// \_direction = direction;

// }

// public void Show()

// {

// base.Show();

// Console.WriteLine($"Direction : {\_direction}");

// }

//}

//class Program

//{

// static void Main(string[] args)

// {

// Manager manager1 = new Manager("John","Eliyev",DateTime.Now,1000,"Babies");

// Manager manager2 = new Manager("John","Eliyev", DateTime.Now,2000, "Babies");

// Scientist scientist1 = new Scientist("Aysel", "Ayselli", DateTime.Now.AddYears(-10), 350, "Math");

// Scientist scientist2 = new Scientist("Elgun", "Elgunlu", DateTime.Now.AddYears(-10), 250, "Literature");

// Programmer programmer1 = new Programmer("Tofiq", "Tofiqli", new DateTime(1985, 10, 10), 3500, "C#");

// Employee[] employees = {scientist1,scientist2, manager1, manager2, programmer1};

// foreach (var e in employees)

// {

// if (e is Manager m)

// {

// m.Show();

// }

// else if (e is Scientist s)

// {

// s.Show();

// }

// else if (e is Programmer p)

// {

// p.Show();

// }

// Console.WriteLine();

// }

// }

//}

////class Program

////{

//// static void Main(string[] args)

//// {

//// //Student student = new Student("John", "Johnlu", DateTime.Now, 81);

//// //student.Show();

//// //Console.WriteLine();

//// //Human human= new Student("John", "Johnlu", DateTime.Now, 81);

//// //human.Show();

//// }

////}

#endregion

#region HomeTask

class Academy

{

public int myId { get; set; } = ++staticAcademyId;

public static int staticAcademyId { get; set; } = 0;

public Group[] groups { get; set; }

}

class Human

{

public string Name { get; set; }

public string Surname { get; set; }

public string Email { get; set; }

public Human() { }

public Human(string name, string surname, string email)

{

Name = name;

Surname = surname;

Email = email;

}

public void Show()

{

Console.WriteLine($"Name : {Name}");

Console.WriteLine($"Surname : {Surname}");

Console.WriteLine($"Email : {Email}");

}

}

class Teacher : Human

{

public int myId { get; set; } = ++staticTeacherId;

public static int staticTeacherId { get; set; } = 0;

public double Salary { get; set; }

public Teacher(string name, string surname, string email, double salary)

: base(name, surname, email)

{

Salary = salary;

}

public void Show()

{

Console.WriteLine($"ID : {myId}");

base.Show();

Console.WriteLine($"Salary : {Salary}");

}

}

class Student : Human

{

public int myId { get; set; }

public static int staticStudentId { get; set; } = 0;

Exam[] Exams { get; set; }

int ExamCount { get; set; } = 0;

public Student(string name, string surname, string email, Exam[] exams, int examCount)

: base(name, surname, email)

{

myId = ++staticStudentId;

Exams = exams;

ExamCount = examCount;

}

public void Show()

{

Console.WriteLine($"ID : {myId}");

base.Show();

if (Exams != null)

{

foreach (Exam exam in Exams)

{

exam.Show();

}

}

else

{

Console.WriteLine("Student does not have an exam.");

}

}

public double GetAverageScore()

{

double totalScore = 0;

foreach (Exam exam in Exams)

{

totalScore += exam.Score;

}

return totalScore / ExamCount;

}

}

class Group

{

public int myId { get; set; } = ++staticGroupId;

public static int staticGroupId { get; set; } = 0;

public string GroupName { get; set; }

public Teacher Teacher { get; set; }

public Student[] Students { get; set; }

public int StudentCount { get; set; } = 0;

public void Show()

{

Console.WriteLine($"Group Id : {myId}");

Console.WriteLine($"Group Name : {GroupName}");

Console.WriteLine($"Teacher : {Teacher}");

foreach (Student student in Students)

{

student.Show();

}

}

}

class Exam

{

public int myId = ++staticExamId;

public static int staticExamId = 0;

public string LessonName;

public double Score;

public DateTime Date;

public void Show()

{

Console.WriteLine($"Exam ID : {myId}");

Console.WriteLine($"Lesson Name : {LessonName}");

Console.WriteLine($"Score : {Score}");

Console.WriteLine($"Date : {Date}");

}

}

class Controller

{

public static void Start()

{

Student student1 = new Student("John", "Johnlu", "john@gmail.com", null, 0);

Student student2 = new Student("Leyla", "Leylali", "leyla@gmail.com", null, 0);

Student student3 = new Student("Tural", "Turalli", "john@gmail.com", null, 0);

student1.Show();

student2.Show();

student3.Show();

}

}

class Program

{

static void Main(string[] args)

{

Controller.Start();

}

}

#endregion

}

//bool IsValidEmail(string email)

//{

// var trimmedEmail = email.Trim();

// if (trimmedEmail.EndsWith("."))

// {

// return false; // suggested by @TK-421

// }

// try

// {

// var addr = new System.Net.Mail.MailAddress(email);

// return addr.Address == trimmedEmail;

// }

// catch

// {

// return false;

// }

//}