using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

namespace ConsoleApp2

{

//class Test

//{

// //const int year = 10;

// readonly int year = 10;

// public Test()

// {

// year = 20;

// }

// public void Show()

// {

// year = 200;

// Console.WriteLine(year);

// }

//}

// 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)

// {

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

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

// //{

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

// // {

// // //matrix[i, k] = (i + 1) \* 10 + k + 1;

// // matrix[i, k] = new Random().Next(1, 100);

// // Thread.Sleep(1);

// // }

// //}

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

// //{

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

// // {

// // Console.Write(matrix[i,k]+" ");

// // }

// // Console.WriteLine();

// //}

// //Car car1 = new Car("M5", "BMW", 4.8);

// //Car car2 = new Car("La Ferrari", "Ferrari", 5.5);

// //Car car3 = new Car("Huracan", "Lamborghini", 3.7);

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

// //gallery[0] = car1;

// //gallery[1] = car2;

// //gallery[2] = car3;

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

// //{

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

// //}

// //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);

// }

// }

//class Human

//{

// public int Id { get; set; }

// public string Name { get; set; }

// public string Surname { get; set; }

// public void Show()

// {

// Console.WriteLine(Id);

// Console.WriteLine(Name);

// Console.WriteLine(Surname);

// }

//}

//class Teacher : Human

//{

// public decimal Salary { get; set; }

// public void Show()

// {

// base.Show();

// Console.WriteLine(Salary);

// }

//}

//class Student:Human

//{

// public double Score { get; set; }

// public void Show()

// {

// base.Show();

// Console.WriteLine(Score);

// }

//}

//class Program

//{

// static void Main(string[] args)

// {

// Human human = new Student

// {

// Id = 1,

// Name = "Leyla",

// Surname = "Leylali",

// Score = 100

// };

// }

//}

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}");

}

}

sealed 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($"AVG : {\_average}");

Console.WriteLine();

}

}

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;

}

public new void Show()

{

base.Show();

Console.WriteLine($"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 new void Show()

{

base.Show();

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

}

}

class Programmer : Employee

{

string \_language;

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

: base(name, surname, date, salary)

{

\_language = lang;

}

public new 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 new 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("Rafiq", "Eliyev", DateTime.Now, 1120, "Babies");

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

//Scientist scientist2 = new Scientist("Elgun", "Elgunlu", DateTime.Now.AddYears(-10), 350, "Literature");

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

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

//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();

//}

//Academy =>Groups

//Group->Teacher,Students

//Teacher=>name,surname,email,salary

//Student=>name,surname,email,Exams,getAvgScore()

//Exam=>lessonname,score,examDate

}

}

//class SuperStudent:Student

//{

//}

//class Program

//{

// static void Main(string[] args)

// {

// //Human human = new Human();

// Student student = new Student("Elvin", "Camalzade", 81);

// student.Show();

// }

//}

}