МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП’ЮТЕРНИХ НАУК

Лабораторна робота № 7

на тему «Наслідування класів»

З дисципліни:

«Програмування під платформу. Net Framework»

Виконала: Чорномаз В. Р., група ІТ-33

Перевірив: Авраменко В.В.

СУМИ – 2025

**Постановка задачі**

**Створити програму мовою C#.**

**Завдання №1**

**Про любий корабель потрібно знати: ім’я, призначення, водотоннажність, потужність двигуна, вид палива.**

**Про авіаносець додатково треба знати, які літаки і в якій кількості він несе. А про ракетоносець треба, крім загальних даних, знати тип ракет та їхню кількість.**

**Описати базовий та похідні від нього класи. Продемонструвати роботу механізму наслідування.**

***Завдання №2***

***Для варіанту завдання створити програму, яка реалізує задану ієрархію класів. Елементами даних і методами кожного класу задатися самостійно.***

***Продемонструвати як похідні класи використовують елементи даних базових класів.***

**До №15 брати варіант згідно із списком в групі або підгрупі. В подальшому переходити до варіантів 1, 2 і т.д.**

**Варіант 11.**

**Автомобіль, поїзд, транспортний засіб, експрес**

**Вирішення завдання 1**

**Вхідні дані**: користувач вводить: ім’я, призначення, водотоннажність, потужність двигуна, вид палива, які літаки і в якій кількості він несе. А про ракетоносець треба, крім загальних даних, знати тип ракет та їхню кількість.

**Лістинг коду:**

using System;

namespace ShipInheritance

{

// Base class: Ship

class Ship

{

protected string name;

protected string purpose;

protected double displacement;

protected double enginePower;

protected string fuelType;

public void Input()

{

Console.WriteLine("Enter ship name:");

name = Console.ReadLine();

Console.WriteLine("Enter ship purpose:");

purpose = Console.ReadLine();

Console.WriteLine("Enter displacement (in tons):");

displacement = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter engine power (in kW):");

enginePower = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter fuel type:");

fuelType = Console.ReadLine();

}

public void Output()

{

Console.WriteLine("\nShip Info:");

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

Console.WriteLine($"Purpose: {purpose}");

Console.WriteLine($"Displacement: {displacement} tons");

Console.WriteLine($"Engine Power: {enginePower} kW");

Console.WriteLine($"Fuel Type: {fuelType}");

}

}

// Derived class: Aircraft Carrier

class AircraftCarrier : Ship

{

protected string aircraftType;

protected int aircraftCount;

public void InputAircraftCarrier()

{

Input(); // Call base class method

Console.WriteLine("Enter aircraft type:");

aircraftType = Console.ReadLine();

Console.WriteLine("Enter number of aircrafts:");

aircraftCount = Convert.ToInt32(Console.ReadLine());

}

public void OutputAircraftCarrier()

{

Output(); // Call base class method

Console.WriteLine($"Aircraft Type: {aircraftType}");

Console.WriteLine($"Aircraft Count: {aircraftCount}");

}

}

// Derived class: Missile Carrier

class MissileCarrier : Ship

{

protected string missileType;

protected int missileCount;

public void InputMissileCarrier()

{

Input(); // Call base class method

Console.WriteLine("Enter missile type:");

missileType = Console.ReadLine();

Console.WriteLine("Enter number of missiles:");

missileCount = Convert.ToInt32(Console.ReadLine());

}

public void OutputMissileCarrier()

{

Output(); // Call base class method

Console.WriteLine($"Missile Type: {missileType}");

Console.WriteLine($"Missile Count: {missileCount}");

}

}

class Program

{

static void Main(string[] args)

{

Console.WriteLine("=== Aircraft Carrier Data ===");

AircraftCarrier ac = new AircraftCarrier();

ac.InputAircraftCarrier();

ac.OutputAircraftCarrier();

Console.WriteLine("\n=== Missile Carrier Data ===");

MissileCarrier mc = new MissileCarrier();

mc.InputMissileCarrier();

mc.OutputMissileCarrier();

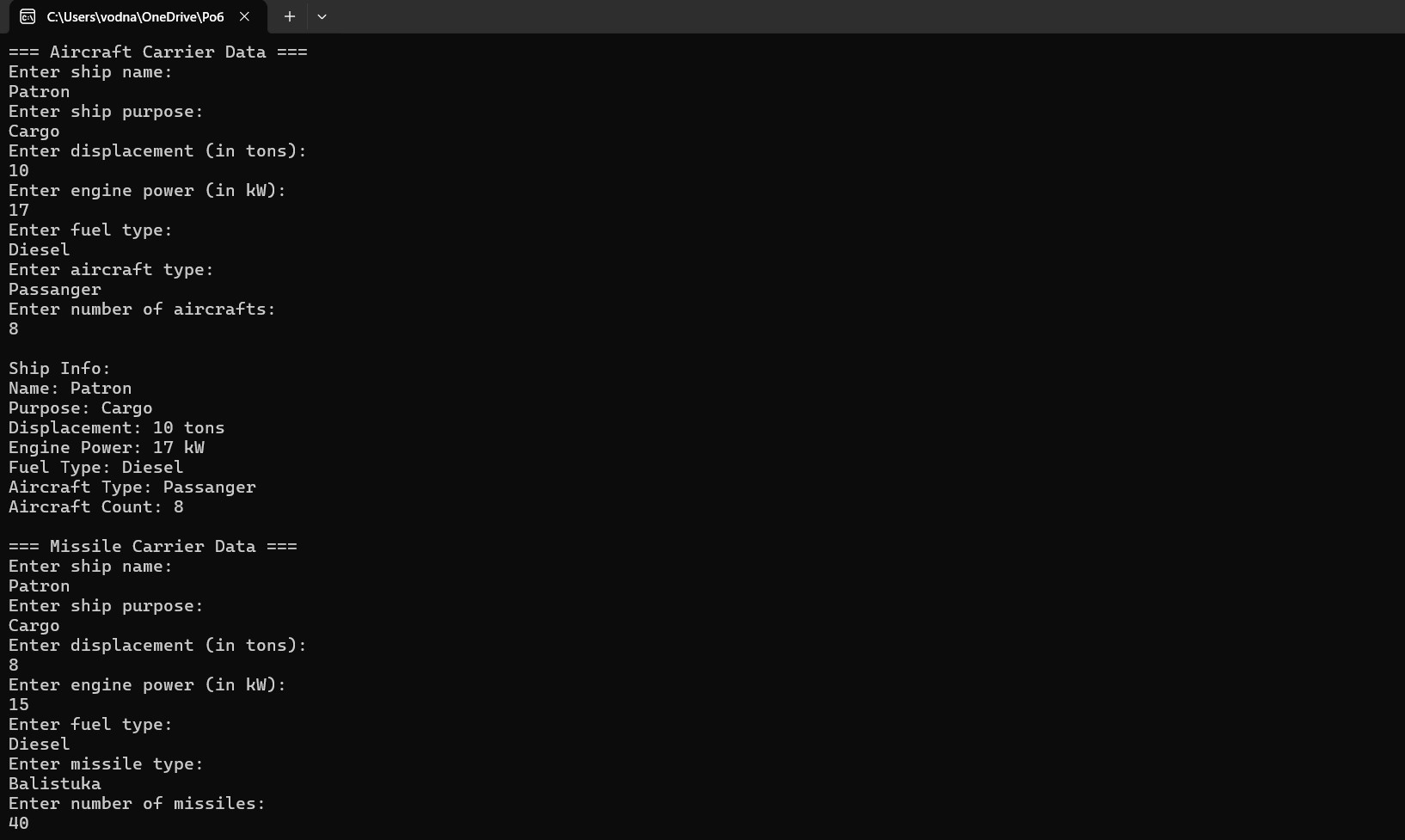
Console.WriteLine("\nPress any key to exit...");

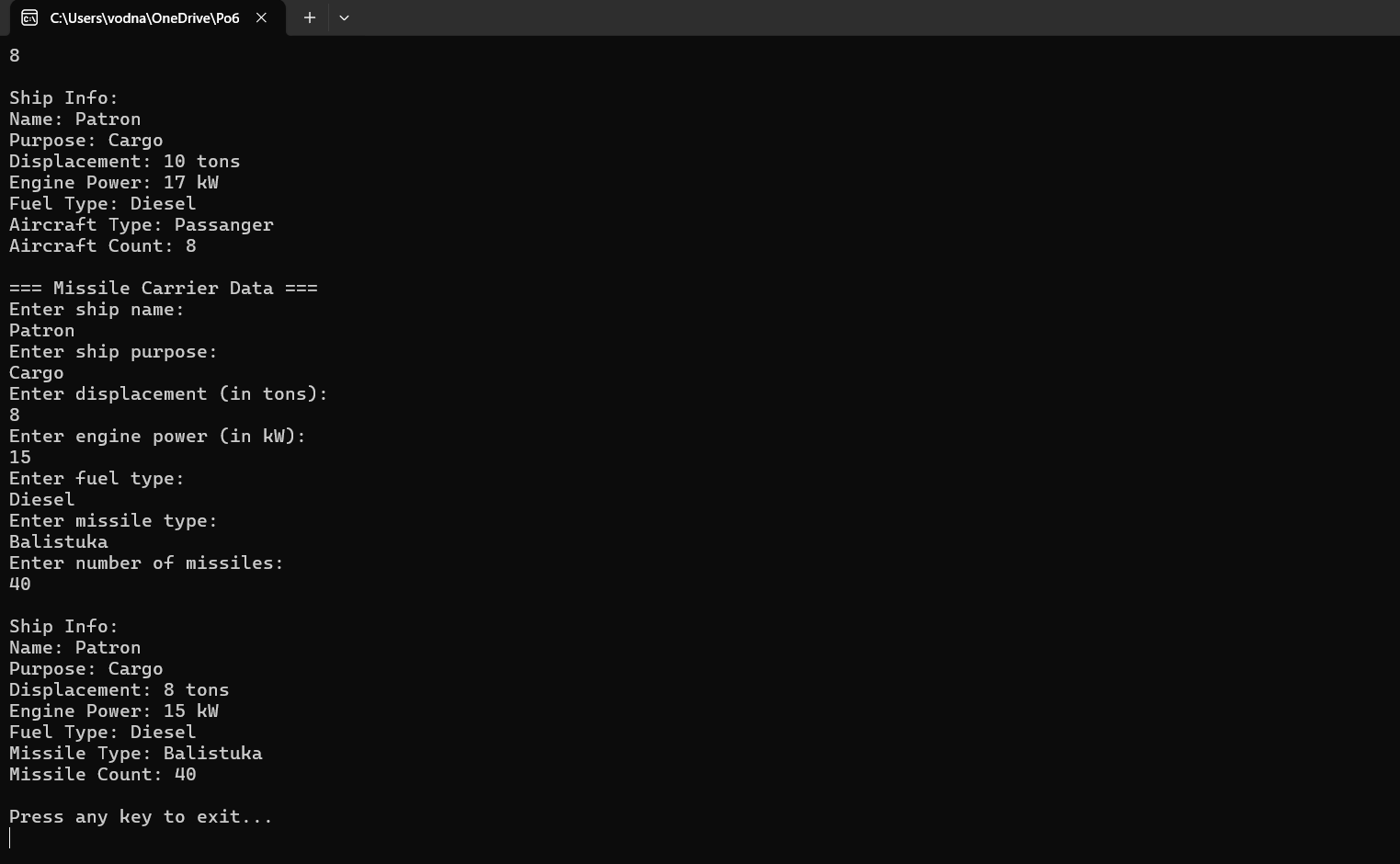
Console.ReadKey();

}

}

}





**Рисунок 1 - Результат виконання програми №1**

**Вирішення завдання 2**

**Лістинг коду:**

using System;

namespace TransportHierarchy

{

class Transport

{

protected string name;

protected double speed;

protected double weight;

public void Input()

{

Console.WriteLine("Enter transport name:");

name = Console.ReadLine();

Console.WriteLine("Enter speed (km/h):");

speed = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter weight (tons):");

weight = Convert.ToDouble(Console.ReadLine());

}

public void Output()

{

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

Console.WriteLine($"Speed: {speed} km/h");

Console.WriteLine($"Weight: {weight} tons");

}

}

class Car : Transport

{

protected string fuelType;

protected int seats;

public void InputCar()

{

Input();

Console.WriteLine("Enter fuel type:");

fuelType = Console.ReadLine();

Console.WriteLine("Enter number of seats:");

seats = Convert.ToInt32(Console.ReadLine());

}

public void OutputCar()

{

Output();

Console.WriteLine($"Fuel Type: {fuelType}");

Console.WriteLine($"Seats: {seats}");

}

}

class Train : Transport

{

protected int carriageCount;

protected string trackType;

public void InputTrain()

{

Input();

Console.WriteLine("Enter number of carriages:");

carriageCount = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Enter track type (e.g. electric/diesel):");

trackType = Console.ReadLine();

}

public void OutputTrain()

{

Output();

Console.WriteLine($"Carriages: {carriageCount}");

Console.WriteLine($"Track Type: {trackType}");

}

}

class Express : Train

{

private string route;

private bool hasWiFi;

public void InputExpress()

{

InputTrain();

Console.WriteLine("Enter route:");

route = Console.ReadLine();

Console.WriteLine("Has WiFi? (yes/no):");

string answer = Console.ReadLine().ToLower();

hasWiFi = (answer == "yes");

}

public void OutputExpress()

{

OutputTrain();

Console.WriteLine($"Route: {route}");

Console.WriteLine($"WiFi Available: {(hasWiFi ? "Yes" : "No")}");

}

}

class Program

{

static void Main(string[] args)

{

Console.WriteLine("=== Car ===");

Car car = new Car();

car.InputCar();

car.OutputCar();

Console.WriteLine("\n=== Express Train ===");

Express express = new Express();

express.InputExpress();

express.OutputExpress();

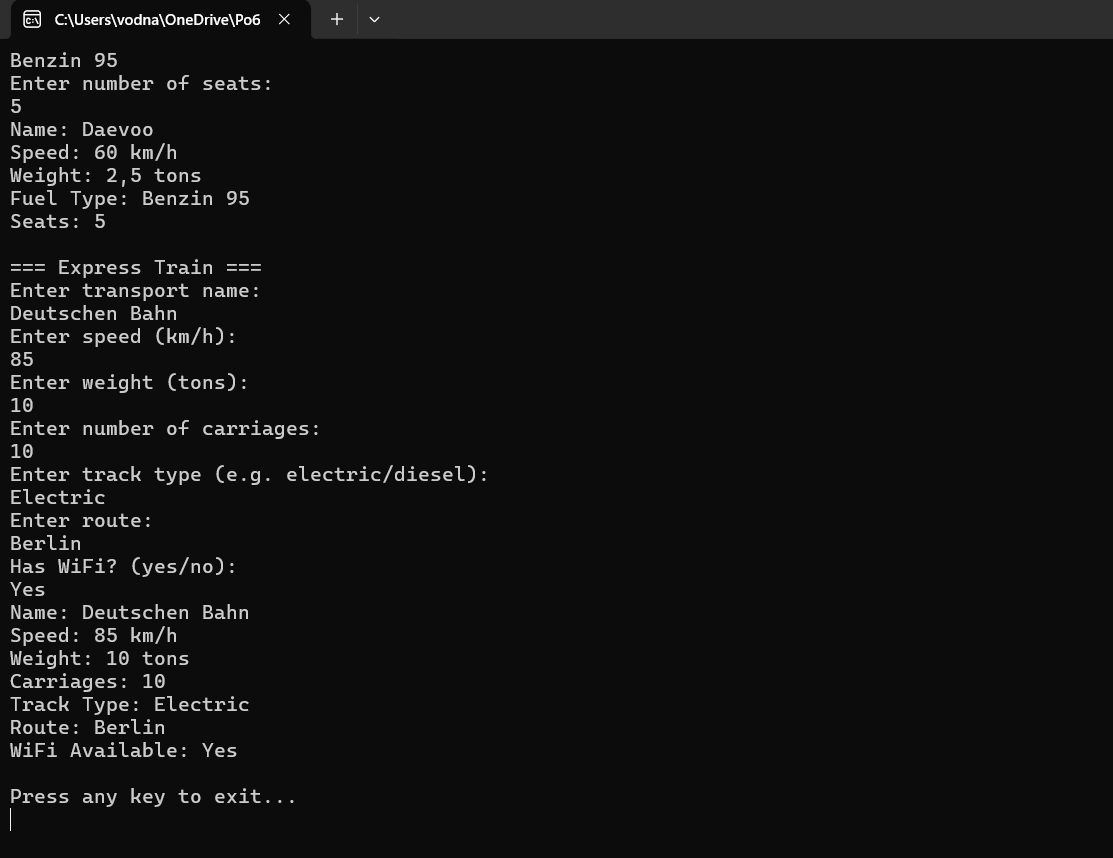
Console.WriteLine("\nPress any key to exit...");

Console.ReadKey();

}

}

}

****