МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

Національний технічний університет України

Факультет інформатики та обчислювальної техніки

«Київський політехнічний інститут імені Ігоря Сікорського»

Кафедра інформаційних систем та технологій

Звіт

з лабораторної роботи № 4

з дисципліни

«Програмування – 2»

Варіант № 30

|  |  |  |
| --- | --- | --- |
| Перевірив: |  | Виконала: Шелест Поліна |
|  |  | Студентка гр. ІС-11 , ФІОТ |
|  |  | 1 курс |

Київ 2021

**Лабораторна робота № 4**

**Тема:** Інтерфейси. Абстрактні класи

**Завдання:**

* Проаналізувати завдання, виділити інформаційні об’єкти та дії. Визначити правильну ієрархію об’єктів
* Створити базовий суперклас (абстрактний клас або інтерфейс) і визначити загальні методи для даного класу. Створити підкласи, в які додати специфічні властивості та методи. Частину методів перевизначити.
* Розробити програму з використанням абстрактних класів та інтерфейсів. Чітко розуміти, де доцільно використати суперклас, а де звичайний.
* При розробці використовувати наслідування та поліморфізм
* У всіх класах повинні бути реалізовані доцільні для класу методи, навіть якщо це не вказано у завданні
* Використовувати об’єкти підкласів для моделювання реальних ситуацій на об’єктів

30. Створити суперклас Корабель і підкласи Воєнний корабель, Авіаносець, Фрегат, Паром. За допомогою конструктора задати швидкість кожного засобу. Визначити кількість участі у бойових діях воєнних кораблів та кількість рейсів у пасажирських. Реалізувати метод присвоєння бойового знання воєнним кораблям.

**Код:**

**Program.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab4

{

class Program

{

static void Main(string[] args)

{

Harbour harbour = new Harbour();

MilitaryShip militaryShip = new MilitaryShip(50, "Військовий корабель", "Військовий");

militaryShip.TimesOfMilitaryAction = 10;

militaryShip.MilitaryTitle = "Корабель-герой порятунку Маріуполя";

Ferry ferry = new Ferry(20, "Паром", "Пасажирський");

ferry.TimesOfVoyages = 15;

harbour.AddShip(ferry);

AircraftCarrier aircraftCarrier = new AircraftCarrier(60, "Авiаносець", "Вiйськовий");

aircraftCarrier.TimesOfMilitaryAction = 15;

aircraftCarrier.MilitaryTitle = "Корабель-герой порятунку Одеси";

harbour.AddShip(aircraftCarrier);

Frigate frigate = new Frigate(30, "Фрегат", "Пасажиський");

frigate.TimesOfMilitaryAction = 25;

frigate.MilitaryTitle = "Корабель-герой порятунку Миколаєва";

harbour.AddShip(frigate);

Console.WriteLine(aircraftCarrier);

Console.WriteLine(frigate);

Console.WriteLine(ferry);

var countMilitary = harbour.GetActionsByTypeMilitary<MilitaryShip>();

var countPassenger = harbour.GetActionsByTypePassenger<PassengerShip>();

Console.WriteLine("Сумарна кiлькiсть воєнних дiй кораблiв: " + countMilitary);

Console.WriteLine("Сумарна кiлькiсть рейсiв кораблiв: " + countPassenger);

Console.ReadLine();

}

public class Ship : Harbour

{

public int Speed { get; set; }

public string Name { get; set; }

public string Type { get; set; }

}

public class MilitaryShip : Ship

{

public MilitaryShip(int speed, string name, string type)

{

Speed = speed;

Name = name;

Type = type;

}

public int TimesOfMilitaryAction { get; set; }

public string MilitaryTitle { get; set; }

}

public class AircraftCarrier : MilitaryShip

{

public AircraftCarrier(int speed, string name, string type) : base(speed, name, type)

{

}

public override string ToString()

{

return $"{Name}. Тип = {Type}, Швидкiсть = {Speed}, кiлькiсть боєвих дiй = {TimesOfMilitaryAction}, звання = {MilitaryTitle} ";

}

}

public class Frigate : MilitaryShip

{

public Frigate(int speed, string name, string type) : base(speed, name, type)

{

}

public override string ToString()

{

return $"{Name}. Тип = {Type}, Швидкiсть = {Speed}, кiлькiсть боєвих дiй = {TimesOfMilitaryAction}, звання = {MilitaryTitle} ";

}

}

public class PassengerShip : Ship

{

public PassengerShip(int speed, string name, string type)

{

Speed = speed;

Name = name;

Type = type;

}

public int TimesOfVoyages { get; set; }

}

public class Ferry : PassengerShip

{

public Ferry(int speed, string name, string type) : base(speed, name, type)

{

}

public override string ToString()

{

return $"{Name}. Тип = {Type}, Швидкiсть = {Speed}, кiлькiсть рейсiв = {TimesOfVoyages} ";

}

}

}

}

**Harbour.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using static Lab4.Program;

namespace Lab4

{

class Harbour

{

private List<Ship> \_listOfShips = new List<Ship>();

public void AddShip(Ship ship)

{

\_listOfShips.Add(ship);

}

public int GetActionsByTypeMilitary<T>() where T : MilitaryShip

{

return \_listOfShips.OfType<T>().Sum(x => x.TimesOfMilitaryAction);

}

public int GetActionsByTypePassenger<T>() where T : PassengerShip

{

return \_listOfShips.OfType<T>().Sum(x => x.TimesOfVoyages);

}

}

}

**Виконання програми:**

