МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

факультет програмної інженерії та бізнесу

кафедра інженерії програмного забезпечення

**Практична робота № 2**

з дисципліни « Об’єктно орієнтоване програмування »

*назва дисципліни*

на тему: « LINQ »

Виконав: студент 2 курсу групи № 622п

освітньої програми

121 інженерія програмного забезпечення

(шифр і назва ОП)

Петренко Є.О.

(прізвище й ініціали студента)

Прийняв: доц. Вдовітченко О.В.

(посада, науковий ступінь, прізвище й ініціали)

Кількість балів:

Харків – 2023

# ПОСТАНОВКА ЗАДАЧІ

**Завдання**

**Завдання:**

1. Клонувати репозиторій:

https://github.com/IlonaShevchenko/Practice\_Linq.git

У проєкті є частковий код:

* опис класу FootballGame, який описує футбольний матч

(FootballGame.сs);

* у файлі Program.cs вже реалізований метод ReadFromFileJson(), який

десеріалізаціє json-файл (data/results\_2010.json) у колекцію

List<FootballGame>;

* також у файлі Program.cs повністю реалізований метод Main().

Склонований проєкт має запускатися і виводити на екран тестове

значення 13049 (це кількість всіх матчів збірних, які були проведені з

2010 року включно).

1. Необхідно реалізувати методи Query1(), Query2(), …, Query10(),

шаблони цих методів наявні в проєкті. Завдання (запити для реалізації)

див. у вигляді коментарів до кожного методу.

Крім запиту у кожному методі має бути реалізований вивід на екран

результатів запиту (приклад оформлення виводу див. файл output.txt).

1. Для проєкту має бути створений власний локальний і віддалений

репозиторії. Після реалізації кожного методу QueryN() необхідно

робити коміт, вказуючи номер N реалізованого запиту.

1. Оформити Readme.md файл.

**Зміст звіту**

* Титульний аркуш
* Завдання
* Реалізація запитів
* Код програми
* Результат роботи програми
* Адреса віддаленого репозиторію

# ХІД РОБОТИ

**Опис програми**

Мова програмування: С#

ОС: Windows 10

Процесор: AMD Ryzen 5 4600H

Компілятор: MS Visual Studio 2022

Репозиторій: <https://github.com/eugegene/Practice-2/tree/master>

Реалізація запитів(див. рисунки 1-10):



Рисунок 1 – Запит 1

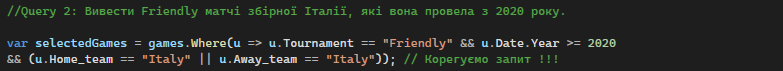


Рисунок 2 – Запит 2

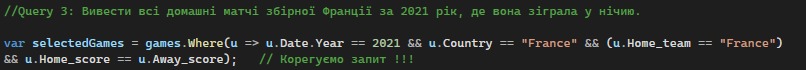


Рисунок 3 – Запит 3

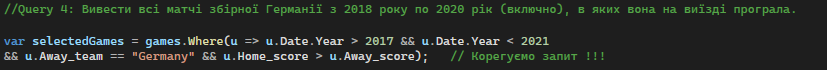


Рисунок 4 – Запит 4

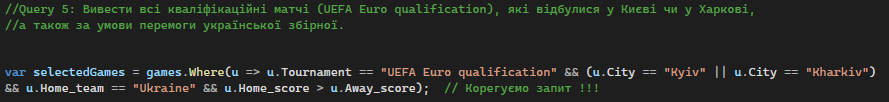


Рисунок 5 – Запит 5

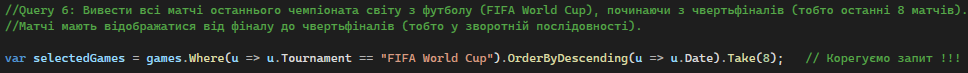


Рисунок 6 – Запит 6

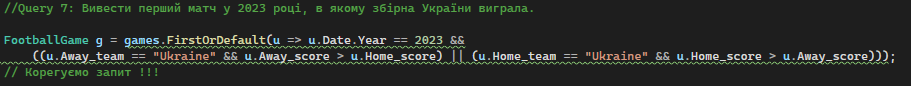


Рисунок 7 – Запит 7

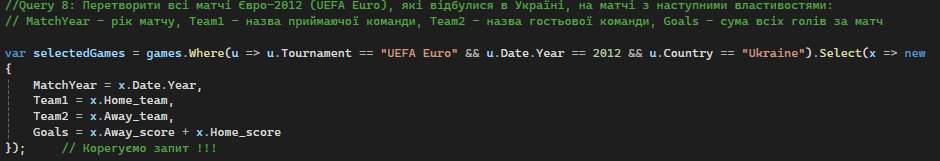


Рисунок 8 – Запит 8

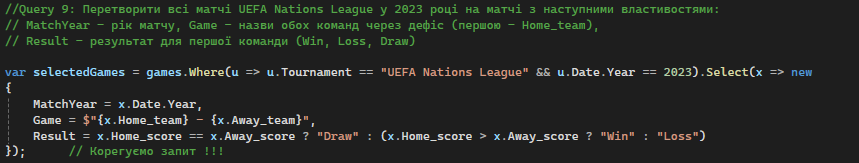


Рисунок 9 – Запит 9



Рисунок 10 – Запит 10

Результат роботи програми наведено нижче (див. рисунки 11-12):

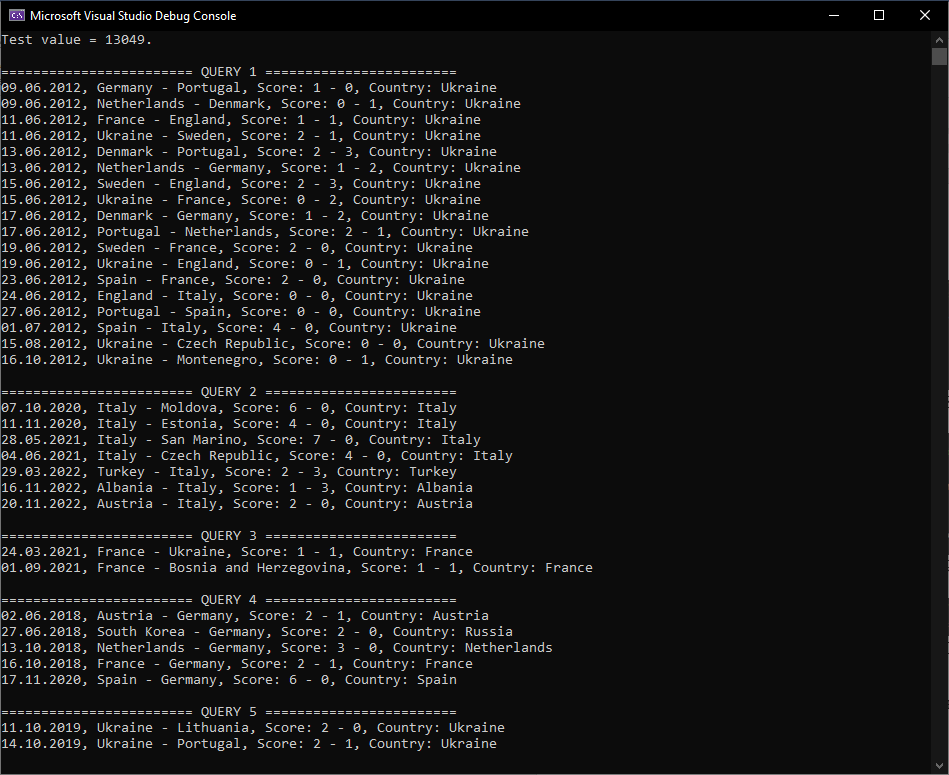


Рисунок 11 – Результати запитів 1-5

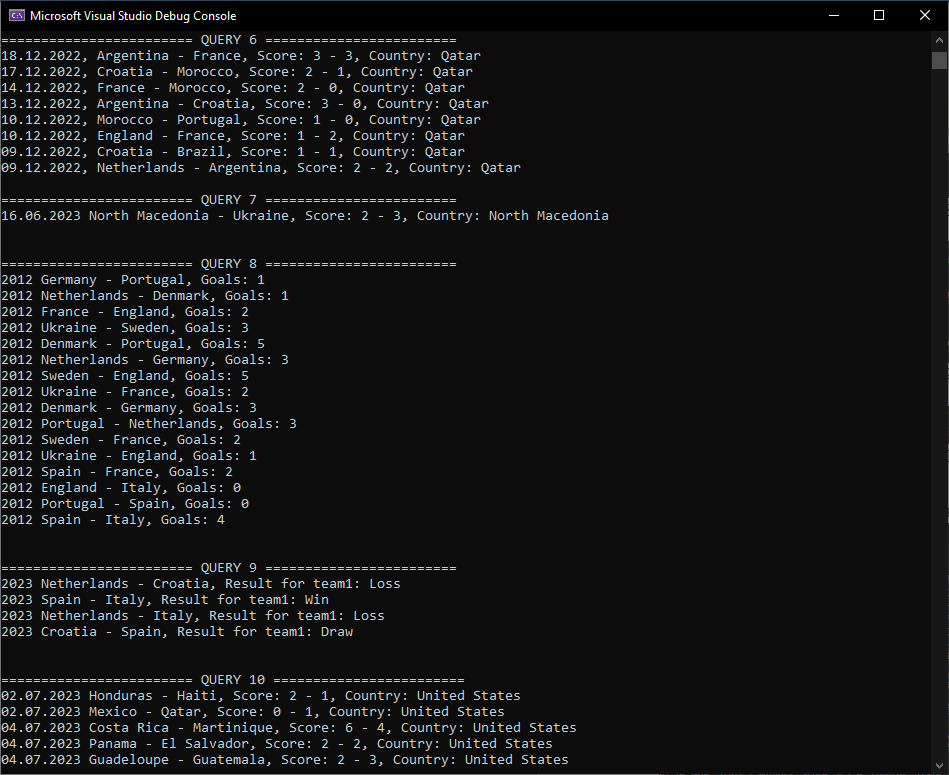


Рисунок 11 – Результати запитів 5-10

Код програми наведено у Додатку А (див. сторінку 7)

# *ДОДАТОК А*

Машинний лістинг програми

**Program.cs:**

using Newtonsoft.Json;

namespace Practice\_Linq

{

public class Program

{

static void Main(string[] args)

{

string path = @"../../../data/results\_2010.json";

List<FootballGame> games = ReadFromFileJson(path);

int test\_count = games.Count();

Console.WriteLine($"Test value = {test\_count}."); // 13049

Query1(games);

Query2(games);

Query3(games);

Query4(games);

Query5(games);

Query6(games);

Query7(games);

Query8(games);

Query9(games);

Query10(games);

}

// Десеріалізація json-файлу у колекцію List<FootballGame>

static List<FootballGame> ReadFromFileJson(string path)

{

var reader = new StreamReader(path);

string jsondata = reader.ReadToEnd();

List<FootballGame> games = JsonConvert.DeserializeObject<List<FootballGame>>(jsondata);

return games;

}

// Запит 1

static void Query1(List<FootballGame> games)

{

//Query 1: Вивести всі матчі, які відбулися в Україні у 2012 році.

var selectedGames = games.Where(u => u.Country == "Ukraine" && u.Date.Year == 2012); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 1 ========================");

foreach(var game in selectedGames) {

Console.WriteLine($"{game.Date.ToString("d")}, {game.Home\_team} - {game.Away\_team}, Score: {game.Home\_score} - {game.Away\_score}, Country: {game.Country}");

}

// див. приклад як має бути виведено:

}

// Запит 2

static void Query2(List<FootballGame> games)

{

//Query 2: Вивести Friendly матчі збірної Італії, які вона провела з 2020 року.

var selectedGames = games.Where(u => u.Tournament == "Friendly" && u.Date.Year >= 2020

&& (u.Home\_team == "Italy" || u.Away\_team == "Italy")); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 2 ========================");

foreach (var game in selectedGames)

{

Console.WriteLine($"{game.Date.ToString("d")}, {game.Home\_team} - {game.Away\_team}, Score: {game.Home\_score} - {game.Away\_score}, Country: {game.Country}");

}

// див. приклад як має бути виведено:

}

// Запит 3

static void Query3(List<FootballGame> games)

{

//Query 3: Вивести всі домашні матчі збірної Франції за 2021 рік, де вона зіграла у нічию.

var selectedGames = games.Where(u => u.Date.Year == 2021 && u.Country == "France" && (u.Home\_team == "France")

&& u.Home\_score == u.Away\_score); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 3 ========================");

foreach (var game in selectedGames)

{

Console.WriteLine($"{game.Date.ToString("d")}, {game.Home\_team} - {game.Away\_team}, Score: {game.Home\_score} - {game.Away\_score}, Country: {game.Country}");

}

// див. приклад як має бути виведено:

}

// Запит 4

static void Query4(List<FootballGame> games)

{

//Query 4: Вивести всі матчі збірної Германії з 2018 року по 2020 рік (включно), в яких вона на виїзді програла.

var selectedGames = games.Where(u => u.Date.Year > 2017 && u.Date.Year < 2021

&& u.Away\_team == "Germany" && u.Home\_score > u.Away\_score); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 4 ========================");

foreach (var game in selectedGames)

{

Console.WriteLine($"{game.Date.ToString("d")}, {game.Home\_team} - {game.Away\_team}, Score: {game.Home\_score} - {game.Away\_score}, Country: {game.Country}");

}

// див. приклад як має бути виведено:

}

// Запит 5

static void Query5(List<FootballGame> games)

{

//Query 5: Вивести всі кваліфікаційні матчі (UEFA Euro qualification), які відбулися у Києві чи у Харкові,

//а також за умови перемоги української збірної.

var selectedGames = games.Where(u => u.Tournament == "UEFA Euro qualification" && (u.City == "Kyiv" || u.City == "Kharkiv")

&& u.Home\_team == "Ukraine" && u.Home\_score > u.Away\_score); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 5 ========================");

foreach (var game in selectedGames)

{

Console.WriteLine($"{game.Date.ToString("d")}, {game.Home\_team} - {game.Away\_team}, Score: {game.Home\_score} - {game.Away\_score}, Country: {game.Country}");

}

// див. приклад як має бути виведено:

}

// Запит 6

static void Query6(List<FootballGame> games)

{

//Query 6: Вивести всі матчі останнього чемпіоната світу з футболу (FIFA World Cup), починаючи з чвертьфіналів (тобто останні 8 матчів).

//Матчі мають відображатися від фіналу до чвертьфіналів (тобто у зворотній послідовності).

var selectedGames = games.Where(u => u.Tournament == "FIFA World Cup").OrderByDescending(u => u.Date).Take(8); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 6 ========================");

foreach (var game in selectedGames)

{

Console.WriteLine($"{game.Date.ToString("d")}, {game.Home\_team} - {game.Away\_team}, Score: {game.Home\_score} - {game.Away\_score}, Country: {game.Country}");

}

// див. приклад як має бути виведено:

}

// Запит 7

static void Query7(List<FootballGame> games)

{

//Query 7: Вивести перший матч у 2023 році, в якому збірна України виграла.

FootballGame g = games.FirstOrDefault(u => u.Date.Year == 2023 &&

((u.Away\_team == "Ukraine" && u.Away\_score > u.Home\_score) || (u.Home\_team == "Ukraine" && u.Home\_score > u.Away\_score)));

// Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 7 ========================");

string res = $"{g.Date.ToString("d")} {g.Home\_team} - {g.Away\_team}, Score: {g.Home\_score} - {g.Away\_score}, Country: {g.Country}\n";

Console.WriteLine(res);

// див. приклад як має бути виведено:

}

// Запит 8

static void Query8(List<FootballGame> games)

{

//Query 8: Перетворити всі матчі Євро-2012 (UEFA Euro), які відбулися в Україні, на матчі з наступними властивостями:

// MatchYear - рік матчу, Team1 - назва приймаючої команди, Team2 - назва гостьової команди, Goals - сума всіх голів за матч

var selectedGames = games.Where(u => u.Tournament == "UEFA Euro" && u.Date.Year == 2012 && u.Country == "Ukraine").Select(x => new

{

MatchYear = x.Date.Year,

Team1 = x.Home\_team,

Team2 = x.Away\_team,

Goals = x.Away\_score + x.Home\_score

}); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 8 ========================");

string res = "";

foreach (var game in selectedGames)

{

res += $"{game.MatchYear} {game.Team1} - {game.Team2}, Goals: {game.Goals}\n";

}

Console.WriteLine(res);

// див. приклад як має бути виведено:

}

// Запит 9

static void Query9(List<FootballGame> games)

{

//Query 9: Перетворити всі матчі UEFA Nations League у 2023 році на матчі з наступними властивостями:

// MatchYear - рік матчу, Game - назви обох команд через дефіс (першою - Home\_team),

// Result - результат для першої команди (Win, Loss, Draw)

var selectedGames = games.Where(u => u.Tournament == "UEFA Nations League" && u.Date.Year == 2023).Select(x => new

{

MatchYear = x.Date.Year,

Game = $"{x.Home\_team} - {x.Away\_team}",

Result = x.Home\_score == x.Away\_score ? "Draw" : (x.Home\_score > x.Away\_score ? "Win" : "Loss")

}); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 9 ========================");

string res = "";

foreach (var game in selectedGames)

{

res += $"{game.MatchYear} {game.Game}, Result for team1: {game.Result}\n";

}

Console.WriteLine(res);

// див. приклад як має бути виведено:

}

// Запит 10

static void Query10(List<FootballGame> games)

{

//Query 10: Вивести з 5-го по 10-тий (включно) матчі Gold Cup, які відбулися у липні 2023 р.

var selectedGames = games.Where(u => u.Tournament == "Gold Cup" && u.Date.Year == 2023 && u.Date.Month == 7).Skip(4).Take(6); // Корегуємо запит !!!

// Перевірка

Console.WriteLine("\n======================== QUERY 10 ========================");

string res = "";

foreach (var game in selectedGames)

{

res += $"{game.Date.ToString("d")} {game.Home\_team} - {game.Away\_team}, Score: {game.Home\_score} - {game.Away\_score}, Country: {game.Country}\n";

}

Console.WriteLine(res);

// див. приклад як має бути виведено:

}

}

}