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 «ООП»

на тему: «LINQ»

Виконав: студент 1 курсу групи № 612пст напряму підготовки (спеціальності)

121 інженерія програмного забезпечення

(шифр і назва напряму підготовки (спеціальності))

Підлісний.М.О.

(прізвище й ініціали студента)

Прийняв: доцент, к.т.н. Лучшев П.О.

(посада, науковий ступінь, прізвище й ініціали)

Національна шкала: Кількість балів: Оцінка ECTS:

Харків – 2023

*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;

− 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).

2. 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).

3. Для проєкту має бути створений власний локальний і віддалений репозиторії. Після реалізації кожного методу QueryN() необхідно робити коміт, вказуючи номер N реалізованого запиту.

4. Оформити Readme.md файл.

*Код програми:*

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;

}

static void PrintMatch(FootballGame game)

{

Console.WriteLine($"{game.Date.ToShortDateString()} {game.Home\_team} - {game.Away\_team}" +

$", Score: {game.Home\_score} - {game.Away\_score}" +

$", Country: {game.Country}");

}

// Запит 1

static void Query1(List<FootballGame> games)

{

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

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

// Перевірка

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

foreach (FootballGame game in selectedGames) { PrintMatch(game); }

}

// 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(game => game.Tournament == "Friendly" && (game.Home\_team == "Italy" || game.Away\_team == "Italy") && game.Date.Year >= 2020); // Корегуємо запит !!!

// Перевірка

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

foreach (FootballGame game in selectedGames) { PrintMatch(game); }

}

// 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(game => game.Country == "France" && game.Home\_team == "France" && game.Date.Year == 2021 && game.Home\_score == game.Away\_score); // Корегуємо запит !!!

// Перевірка

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

foreach (FootballGame game in selectedGames) { PrintMatch(game); }

}

// 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(game => game.Away\_team == "Germany" && game.Date.Year >= 2018 && game.Date.Year <= 2020 && game.Home\_score > game.Away\_score); // Корегуємо запит !!!

// Перевірка

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

foreach (FootballGame game in selectedGames) { PrintMatch(game); }

}

// 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(game => game.Tournament == "UEFA Euro qualification" &&

(game.City == "Kyiv" || game.City == "Kharkiv") &&

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

// Перевірка

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

foreach (FootballGame game in selectedGames) { PrintMatch(game); }

}

// 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(game => game.Tournament == "FIFA World Cup").OrderByDescending(game => game.Date).Take(8); // Корегуємо запит !!!

// Перевірка

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

foreach (FootballGame game in selectedGames) { PrintMatch(game); }

}

// Запит 7

static void Query7(List<FootballGame> games)

{

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

FootballGame game = games.Where(game => game.Date.Year == 2023 && ((game.Home\_team == "Ukraine" && game.Home\_score > game.Away\_score) || (game.Away\_team == "Ukraine" && game.Away\_score > game.Home\_score))).OrderBy(game => game.Date).First(); // Корегуємо запит !!!

// Перевірка

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

PrintMatch(game);

}

// 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(game => game.Tournament == "UEFA Euro" && game.Country == "Ukraine")

.Select(game => new

{

MatchYear = game.Date.Year,

Team1 = game.Home\_team,

Team2 = game.Away\_team,

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

}); // Корегуємо запит !!!

// Перевірка

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

foreach (var game in selectedGames)

{

Console.WriteLine($"{game.MatchYear} {game.Team1} - {game.Team2}, Goals: {game.Goals}");

}

}

// 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(game => game.Tournament == "UEFA Nations League" && game.Date.Year == 2023)

.Select(game => new

{

MatchYear = game.Date.Year,

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

Result = GetResultForTeam1(game)

}); // Корегуємо запит !!!

// Перевірка

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

foreach (var game in selectedGames)

{

Console.WriteLine($"{game.MatchYear} {game.Game}, Result for team1: {game.Result}");

}

}

static string GetResultForTeam1(FootballGame game)

{

if (game.Home\_score > game.Away\_score)

return "Win";

else if (game.Home\_score < game.Away\_score)

return "Loss";

else

return "Draw";

}

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

// Перевірка

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

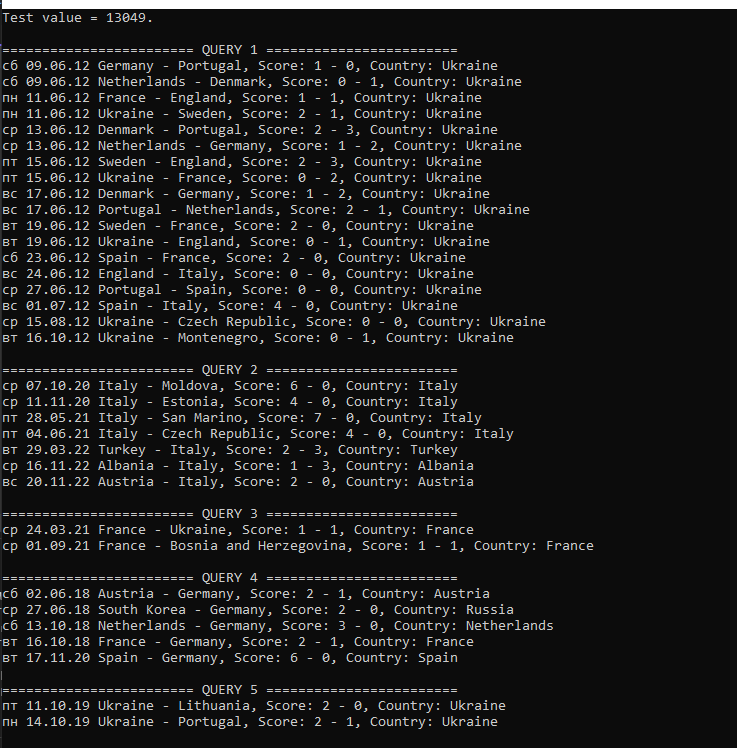
foreach (FootballGame game in selectedGames) { PrintMatch(game); }

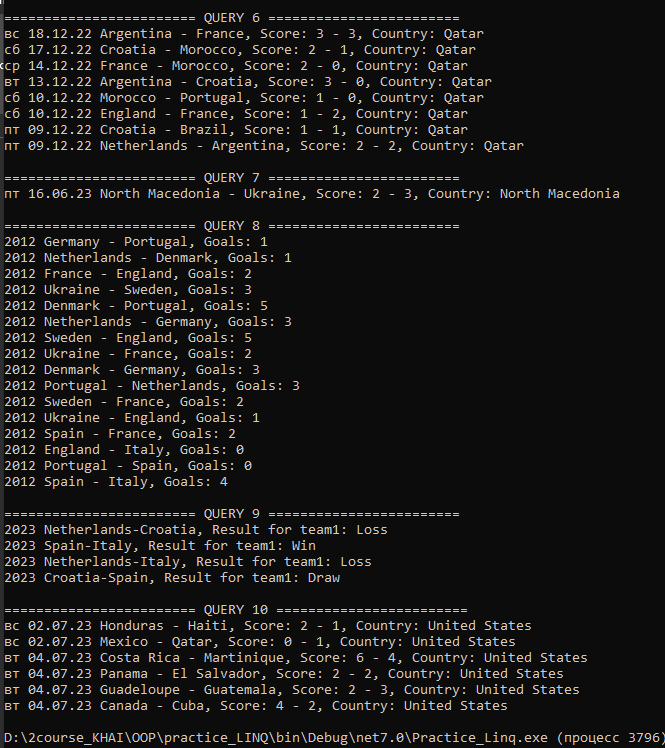
}

}

}

*Результат роботи програми:*





*Адреса віддаленого репозиторію:*

https://github.com/MykytaPD09/Practice\_Linq