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)

Зайченко Ярослав Ігорович

(прізвище й ініціали студента)

Прийняв: доц. Вдовітченко О.В.

(посада, науковий ступінь, прізвище й ініціали)

Кількість балів:

Харків – 2023

# ПОСТАНОВКА ЗАДАЧІ

1. Клонувати репозиторій: github.com/IlonaShevchenko/Practice\_Linq.git
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. ОформитиReadme.md файл.

Зміст звіту

* Титульний аркуш
* Постановка завдання
* Реалізація запитів (навести формулювання запиту і його реалізацію за допомогою LINQ)
* Код програми
* Результат роботи програми
* Адреса віддаленого репозиторію

ХІД РОБОТИ

Реалізація запитів

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

var selectedGames = games.Where(UKgameIN2012 => UKgameIN2012.Country == "Ukraine" && UKgameIN2012.Date.Year == 2012);

1. Вивести Friendly матчі збірної Італії, які вона провела з 2020 року.

var selectedGames = games.Where(FrgameIT2020 => FrgameIT2020.Tournament == "Friendly" && FrgameIT2020.Date.Year >= 2020 && (FrgameIT2020.Home\_team == "Italy" || FrgameIT2020.Away\_team == "Italy"));

1. Вивести всі домашні матчі збірної Франції за 2021 рік, де вона зіграла у нічию.

var selectedGames = games.Where(DrawFRgame2021 => DrawFRgame2021.Date.Year == 2021 && DrawFRgame2021.Country == "France" && (DrawFRgame2021.Home\_team == "France") && DrawFRgame2021.Home\_score == DrawFRgame2021.Away\_score);

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

var selectedGames = games.Where(DEgame18\_20LOSER => DEgame18\_20LOSER.Date.Year >= 2018 && DEgame18\_20LOSER.Date.Year <= 2020 && DEgame18\_20LOSER.Away\_team == "Germany" && DEgame18\_20LOSER.Home\_score > DEgame18\_20LOSER.Away\_score);

1. 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(UK\_Win\_Cvalification\_UEFA => UK\_Win\_Cvalification\_UEFA.Tournament == "UEFA Euro qualification" && (UK\_Win\_Cvalification\_UEFA.City == "Kyiv" || UK\_Win\_Cvalification\_UEFA.City == "Kharkiv") && UK\_Win\_Cvalification\_UEFA.Home\_team == "Ukraine" && UK\_Win\_Cvalification\_UEFA.Home\_score > UK\_Win\_Cvalification\_UEFA.Away\_score);

1. Вивести всі матчі останнього чемпіоната світу з футболу (FIFA World Cup), починаючи з чвертьфіналів.

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

1. Вивести перший матч у 2023 році, в якому збірна України виграла.

FootballGame g = games.FirstOrDefault(FSTgameUKwinIN2023 => FSTgameUKwinIN2023.Date.Year == 2023 && ((FSTgameUKwinIN2023.Away\_team == "Ukraine" && FSTgameUKwinIN2023.Away\_score > FSTgameUKwinIN2023.Home\_score) || (FSTgameUKwinIN2023.Home\_team == "Ukraine" && FSTgameUKwinIN2023.Home\_score > FSTgameUKwinIN2023.Away\_score)));

1. Перетворити всі матчі Євро-2012 (UEFA Euro), які відбулися в Україні, на матчі з наступними властивостями.

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

{

MatchYear = x.Date.Year,

Team1 = x.Home\_team,

Team2 = x.Away\_team,

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

});

1. Перетворити всі матчі UEFA Nations League у 2023 році на матчі з наступними властивостями.

var selectedGames = games.Where(convertUEFANL2023 => convertUEFANL2023.Tournament == "UEFA Nations League" && convertUEFANL2023.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")

});

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

var selectedGames = games.Where(GoldCUPgameJULY2023 => GoldCUPgameJULY2023.Tournament == "Gold Cup" && GoldCUPgameJULY2023.Date.Year == 2023 && GoldCUPgameJULY2023.Date.Month == 7).Skip(4).Take(6);

Код програми

using Newtonsoft.Json;

using System.Linq;

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(UKgameIN2012 => UKgameIN2012.Country == "Ukraine" && UKgameIN2012.Date.Year == 2012);

// Перевірка

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

foreach (var gameIN2012UK in selectedGames)

Console.WriteLine($"{gameIN2012UK.Date.ToString("d")} {gameIN2012UK.Home\_team} - {gameIN2012UK.Away\_team}, Score: {gameIN2012UK.Home\_score} - {gameIN2012UK.Away\_score}, Country: {gameIN2012UK.Country}");

}

// 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(FrgameIT2020 => FrgameIT2020.Tournament == "Friendly" && FrgameIT2020.Date.Year >= 2020 && (FrgameIT2020.Home\_team == "Italy" || FrgameIT2020.Away\_team == "Italy"));

// 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}");

}

// 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(DrawFRgame2021 => DrawFRgame2021.Date.Year == 2021 && DrawFRgame2021.Country == "France" && (DrawFRgame2021.Home\_team == "France") && DrawFRgame2021.Home\_score == DrawFRgame2021.Away\_score);

// Перевірка

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

foreach (var DrawFRgame2021 in selectedGames)

Console.WriteLine($"{DrawFRgame2021.Date.ToString("d")}, {DrawFRgame2021.Home\_team} - {DrawFRgame2021.Away\_team}, Score: {DrawFRgame2021.Home\_score} - {DrawFRgame2021.Away\_score}, Country: {DrawFRgame2021.Country}");

}

// 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(DEgame18\_20LOSER => DEgame18\_20LOSER.Date.Year >= 2018 && DEgame18\_20LOSER.Date.Year <= 2020 && DEgame18\_20LOSER.Away\_team == "Germany" && DEgame18\_20LOSER.Home\_score > DEgame18\_20LOSER.Away\_score);

// Перевірка

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

foreach (var DEgame18\_20LOSER in selectedGames)

Console.WriteLine($"{DEgame18\_20LOSER.Date.ToString("d")}, {DEgame18\_20LOSER.Home\_team} - {DEgame18\_20LOSER.Away\_team}, Score: {DEgame18\_20LOSER.Home\_score} - {DEgame18\_20LOSER.Away\_score}, Country: {DEgame18\_20LOSER.Country}");

}

// 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(UK\_Win\_Cvalification\_UEFA => UK\_Win\_Cvalification\_UEFA.Tournament == "UEFA Euro qualification" && (UK\_Win\_Cvalification\_UEFA.City == "Kyiv" || UK\_Win\_Cvalification\_UEFA.City == "Kharkiv") && UK\_Win\_Cvalification\_UEFA.Home\_team == "Ukraine" && UK\_Win\_Cvalification\_UEFA.Home\_score > UK\_Win\_Cvalification\_UEFA.Away\_score);

// Перевірка

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

foreach (var UK\_Win\_Cvalification\_UEFA in selectedGames)

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

}

// 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(LSTWrldCUP => LSTWrldCUP.Tournament == "FIFA World Cup").OrderByDescending(u => u.Date).Take(8);

// Перевірка

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

foreach(var LSTWrldCUP in selectedGames)

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

}

// 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(FSTgameUKwinIN2023 => FSTgameUKwinIN2023.Date.Year == 2023 &&((FSTgameUKwinIN2023.Away\_team == "Ukraine" && FSTgameUKwinIN2023.Away\_score > FSTgameUKwinIN2023.Home\_score) || (FSTgameUKwinIN2023.Home\_team == "Ukraine" && FSTgameUKwinIN2023.Home\_score > FSTgameUKwinIN2023.Away\_score)));

// Перевірка

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

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

}

// 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(Convertation => Convertation.Tournament == "UEFA Euro" && Convertation.Date.Year == 2012 && Convertation.Country == "Ukraine").Select(x => new

{

MatchYear = x.Date.Year,

Team1 = x.Home\_team,

Team2 = x.Away\_team,

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

});

// Перевірка

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

foreach(var Convertation in selectedGames)

Console.WriteLine($"{Convertation.MatchYear} {Convertation.Team1} - {Convertation.Team2}, Goals: {Convertation.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(convertUEFANL2023 => convertUEFANL2023.Tournament == "UEFA Nations League" && convertUEFANL2023.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")

});

// Перевірка

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

foreach (var convertUEFANL2023 in selectedGames)

Console.WriteLine($"{convertUEFANL2023.MatchYear} {convertUEFANL2023.Game}, Result for team1: {convertUEFANL2023.Result}");

}

// 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(GoldCUPgameJULY2023 => GoldCUPgameJULY2023.Tournament == "Gold Cup" && GoldCUPgameJULY2023.Date.Year == 2023 && GoldCUPgameJULY2023.Date.Month == 7).Skip(4).Take(6);

// Перевірка

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

foreach (var GoldCUPgameJULY2023 in selectedGames)

Console.WriteLine($"{GoldCUPgameJULY2023.Date.ToString("d")} {GoldCUPgameJULY2023.Home\_team} - {GoldCUPgameJULY2023.Away\_team}, Score: {GoldCUPgameJULY2023.Home\_score} - {GoldCUPgameJULY2023.Away\_score}, Country: {GoldCUPgameJULY2023.Country}");

}

}

}

Результат роботи програми

Після написання повного коду програми можемо запустити його, та впевнитись, що виведення результатів запитів (рис. 1, 2) співпадає з прикладом у файлі output\_example.

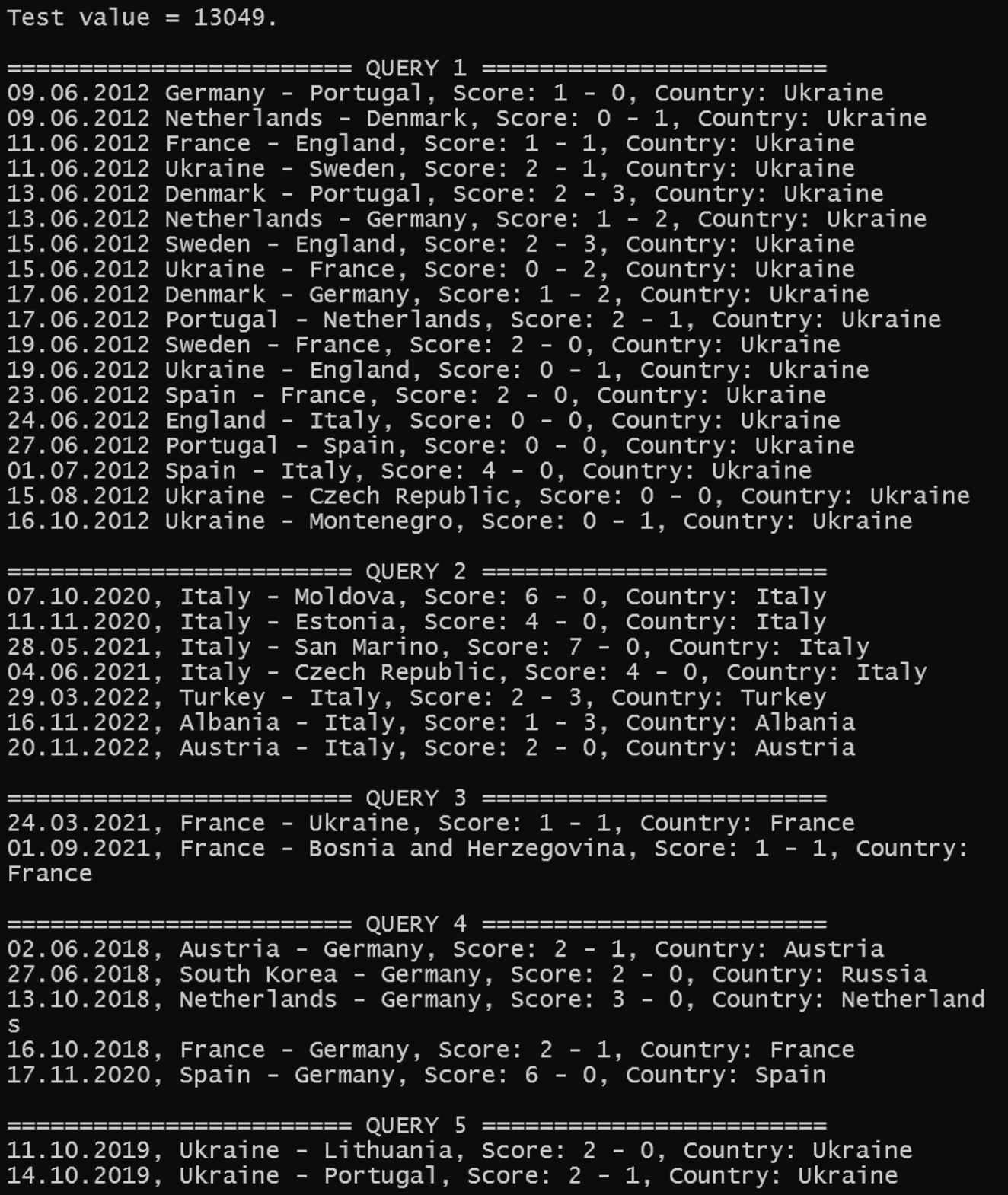


Рисунок 1 – Виведення на екран запитів 1-5

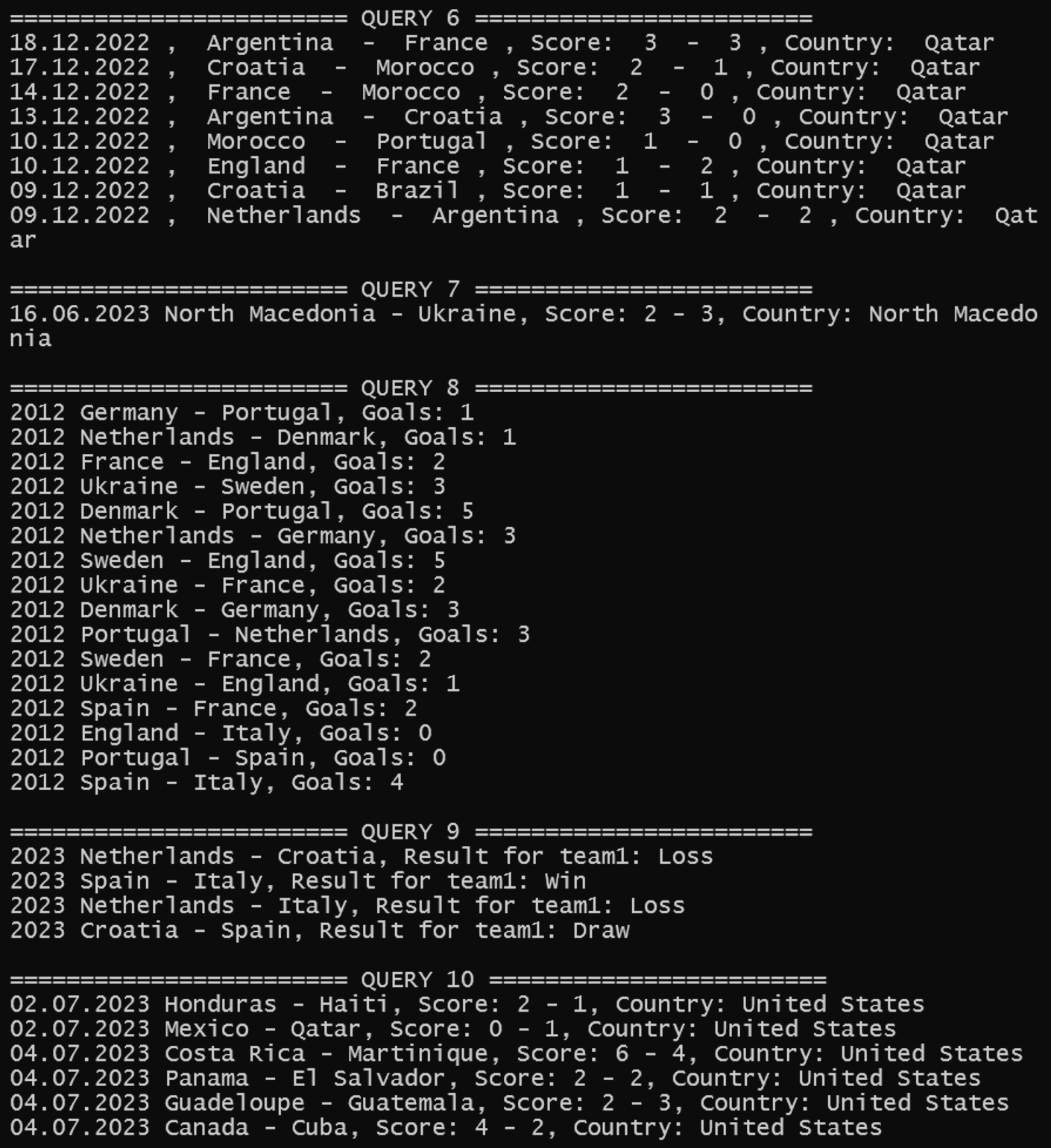


Рисунок 2 – Виведення на екран запитів 6-10

Адреса віддаленого репозиторію

<https://github.com/Yarik-Za/OOP_Linq_Practice_2>