(T19)討論DictionaryTKey、TValue、KeyValuePairTKey、TValue  
CourseGUID: 29f1196a-1950-41a4-b9c1-dd13a9e92d92  
=======================================================================  
(T19)討論DictionaryTKey、TValue、KeyValuePairTKey、TValue  
=======================================================================  
0. Summary

-----------

1. New Project

1.1. Create New Project : Sample

-----------

2. Sample : Program.cs  
=======================================================================

0. Summary

A Dictionary<TKey, TValue> is a collection of KeyValuePair<TKey, TValue>

and from System.Collections.Generic namespace.

TKey must be unique so using Tkey can get its pair TValue quickly.

1. New Project

1.1. Create New Project : Sample

File --> New --> Project... -->

Visual C# -->  **Console App** **(.Net Framework)** -->

Name: **Sample**







=============================================

2. Sample : Program.cs

using System;

using System.Collections.Generic;

using System.Linq;

using OnLineGame;

namespace Sample

{

    class Program

    {

        static void Main(string[] args)

        {

            // 1. ----------------------------------------------------

            Console.WriteLine("1. DictionarySample() =============================");

            DictionarySample();

            Console.ReadLine();

        }

        // 1. ---------------------------------------------

        static void DictionarySample()

        {

            // 1.0. ------------------------------------------------

            //Create Dictionary<TKey, TValue>

            //A Dictionary<TKey, TValue> is a collection of KeyValuePair<TKey, TValue>

            //and from System.Collections.Generic namespace.

            //TKey must be unique so using Tkey can get its pair TValue quickly.

            //In this case, TKey is string type, TValue is Gamer Type.

            Dictionary<string, Gamer> dictionaryGamers = new Dictionary<string, Gamer>

            {

                {"Key1", new Gamer {Id = 1, Name = "Name1", Email = "[1@1.com](mailto:1@1.com)"}},

                {"Key2", new Gamer {Id = 2, Name = "Name2", Email = "[2@2.com](mailto:2@2.com)"}}

            };

            dictionaryGamers.Add("Key3", new Gamer { Id = 3, Name = "Name3", Email = "[3@3.com](mailto:3@3.com)" });

            dictionaryGamers.Add("Key4", new Gamer { Id = 4, Name = "Name4", Email = "[4@4.com](mailto:4@4.com)" });

            // 1.1. ------------------------------------------------

            //print key and value

            Console.WriteLine("1.1. dictionaryGamers --------------------");

            KeyValuePair<string, Gamer> key1Value =

                dictionaryGamers.FirstOrDefault(g => g.Key == "Key1");

            Gamer gamer1 = dictionaryGamers.FirstOrDefault(g => g.Key == "Key1").Value;

            KeyValuePair<string, Gamer> key2Value =

                dictionaryGamers.FirstOrDefault(x => x.Key == "Key2");

            KeyValuePair<string, Gamer> key3Value =

                dictionaryGamers.FirstOrDefault(x => x.Key == "Key3");

            KeyValuePair<string, Gamer> key4Value =

                dictionaryGamers.FirstOrDefault(x => x.Key == "Key4");

            Console.WriteLine($"key1Value.Key : {key1Value.Key}  ;  " +

                              $"key1Value.Value.ToString() : {key1Value.Value.ToString()}");

            Console.WriteLine($"key2Value.Key : {key2Value.Key}  ;  " +

                              $"key2Value.Value.ToString() : {key2Value.Value.ToString()}");

            Console.WriteLine($"key3Value.Key : {key3Value.Key}  ;  " +

                              $"key3Value.Value : {key3Value.Value}");

            Console.WriteLine($"key4Value.Key : {key4Value.Key}  ;  " +

                              $"key4Value.Value : {key4Value.Value}");

            //1.1.dictionaryGamers--------------------

            //key1Value.Key : Key1; key1Value.Value.ToString() : Id == 1; Name == Name1; Email: [1@1.com](mailto:1@1.com)

            //key2Value.Key : Key2; key2Value.Value.ToString() : Id == 2; Name == Name2; Email: [2@2.com](mailto:2@2.com)

            //key3Value.Key : Key3; key3Value.Value : Id == 3; Name == Name3; Email: [3@3.com](mailto:3@3.com)

            //key4Value.Key : Key4; key4Value.Value : Id == 4; Name == Name4; Email: [4@4.com](mailto:4@4.com)

            // 1.2. ------------------------------------------------

            //ObjectDictionary[TKey] == TValue

            Console.WriteLine("1.2. dictionaryGamers input key output value ----------------");

            Gamer key1Gamer = dictionaryGamers["Key1"];

            Console.WriteLine("dictionaryGamers[\"Key1\"]  :  {0}", key1Gamer);

            //1.2. dictionaryGamers input key output value ----------------

            //dictionaryGamers["Key1"]  :  Id == 1; Name == Name1; Email: [1@1.com](mailto:1@1.com)

          // 1.3. ------------------------------------------------

            //TKey keyItem = ObjectDictionary.Keys.ElementAt(intIndex);

            //TValue valueItem = ObjectDictionary[ObjectDictionary.Keys.ElementAt(intIndex)];

            Console.WriteLine("1.3. get key and value ----------------");

            string lastItem = dictionaryGamers.Keys.ElementAt(dictionaryGamers.Count - 1);

            Console.WriteLine($"dictionaryGamers.Keys.ElementAt(dictionaryGamers.Count - 1)  ==  {lastItem}");  // Key, "Key4"

            Console.WriteLine($"dictionaryGamers[dictionaryGamers.Keys.ElementAt(dictionaryGamers.Count - 1)]  :  " +

                              $"{dictionaryGamers[lastItem]}");

            // Value, (Id == 4 ; Name == Name04 ; Email : [4@4.com](mailto:4@4.com))

            //1.3. get key and value ----------------

            //dictionaryGamers.Keys.ElementAt(dictionaryGamers.Count - 1)  ==  Key4

            //dictionaryGamers[dictionaryGamers.Keys.ElementAt(dictionaryGamers.Count - 1)]  :  Id == 4 ; Name == Name4 ; Email : [4@4.com](mailto:4@4.com)

            string fitstItem = dictionaryGamers.Keys.ElementAt(0);

            Console.WriteLine($"dictionaryGamers.Keys.ElementAt(0)  ==  {fitstItem}");  // Key, "Key1"

            Console.WriteLine($"dictionaryGamers[dictionaryGamers.Keys.ElementAt(0)]  :  " +

                              $"{dictionaryGamers[fitstItem]}");    // Value, (Id == 1 ; Name == Name01 ; Email : [1@1.com](mailto:1@1.com))

            //dictionaryGamers.Keys.ElementAt(0)  ==  Key1

            //dictionaryGamers[dictionaryGamers.Keys.ElementAt(0)]  :  Id == 1 ; Name == Name1 ; Email : [1@1.com](mailto:1@1.com)

            // 1.4. ------------------------------------------------

            //Print all elements from Dictionary

            Console.WriteLine("1.4. print all elements from Dictionary ----------------");

            foreach (KeyValuePair<string, Gamer> item in dictionaryGamers)

            {

                Console.WriteLine($"item.Key == {item.Key}   ;   item.Value : {item.Value}");

            }

            //1.4.print all elements from Dictionary ----------------

            //item.Key == Key1   ; item.Value : Id == 1; Name == Name1; Email: [1@1.com](mailto:1@1.com)

            //item.Key == Key2   ; item.Value : Id == 2; Name == Name2; Email: [2@2.com](mailto:2@2.com)

            //item.Key == Key3   ; item.Value : Id == 3; Name == Name3; Email: [3@3.com](mailto:3@3.com)

            //item.Key == Key4   ; item.Value : Id == 4; Name == Name4; Email: [4@4.com](mailto:4@4.com)

            // 1.5. ------------------------------------------------

            //Print all keys

            Console.WriteLine("1.5. Print all keys ----------------");

            foreach (string itemKey in dictionaryGamers.Keys)

            {

                Console.WriteLine($"itemKey == {itemKey}");

            }

            //1.5.Print all keys ----------------

            //itemKey == Key1

            //itemKey == Key2

            //itemKey == Key3

            //itemKey == Key4

           // 1.6. ------------------------------------------------

            //Print all Values

            Console.WriteLine("1.6. Print all values ----------------");

            foreach (Gamer itemValue in dictionaryGamers.Values)

            {

                Console.WriteLine($"itemValue : {itemValue}");

            }

            //1.6.Print all values ----------------

            //itemValue : Id == 1; Name == Name1; Email: [1@1.com](mailto:1@1.com)

            //itemValue : Id == 2; Name == Name2; Email: [2@2.com](mailto:2@2.com)

            //itemValue : Id == 3; Name == Name3; Email: [3@3.com](mailto:3@3.com)

            //itemValue : Id == 4; Name == Name4; Email: [4@4.com](mailto:4@4.com)

            // 1.7. ------------------------------------------------

            //DictionaryObject.ContainsKey(TKey)

            //TKey must be unique,

            //so better to check if the TKey is already existed before adding new item.

            Console.WriteLine("1.7. DictionaryObject.ContainsKey(TKey) ----------------");

            if (!dictionaryGamers.ContainsKey("Key5"))

            {

                dictionaryGamers.Add("Key5", new Gamer

                {

                    Id = 5,

                    Name = "Name5",

                    Email = "[5@5.com](mailto:5@5.com)"

                });

            }

            // if the key does not exist, then throw KeyNotFoundException.

            Console.WriteLine(

                dictionaryGamers.ContainsKey("Key5") ?

                $"dictionaryGamers[\"Key5\"]  :  {dictionaryGamers["Key5"]}" :

                "Key does not exist in the dictionary");

            //1.7.DictionaryObject.ContainsKey(TKey)----------------

            //dictionaryGamers["Key5"]  :  Id == 5; Name == Name5; Email: [5@5.com](mailto:5@5.com)

            // 1.8. ------------------------------------------------

            //DictionaryObject.TryGetValue(TKey, out itemValue)

            //TryGetValue() will try to get the value from the dictionary.

            //return false if it fails.

            ////E.g.

            ////TValue itemValue;

            ////bool getValueByKey = DictionaryObject.TryGetValue(TKey, out itemValue);

            Console.WriteLine("1.8. DictionaryObject.TryGetValue(TKey, out itemValue) ----------------");

            Gamer gamer;

            Console.WriteLine(

                dictionaryGamers.TryGetValue("Key5", out gamer) ?

                $"gamer : {gamer}" :

                "Value is not found by the key");

            //1.8.DictionaryObject.TryGetValue(TKey, out itemValue)----------------

            //gamer: Id == 5; Name == Name5; Email: [5@5.com](mailto:5@5.com)

           // 1.9. ------------------------------------------------

            //DictionaryObject.Count  return the number of item in the DictionaryObject.

            Console.WriteLine("1.9. DictionaryObject.Count ----------------");

            Console.WriteLine($"dictionaryGamers.Count = {dictionaryGamers.Count}");

            // using LINQ extension methods to find Id > 2

            Console.WriteLine($"dictionaryGamers.Count(g => g.Value.Id > 2)  :  {dictionaryGamers.Count(g => g.Value.Id > 2)}");

            //1.9.DictionaryObject.Count----------------

            //dictionaryGamers.Count = 5

            //dictionaryGamers.Count(g => g.Value.Id > 2)  :  3

            // 1.10. ------------------------------------------------

            //DictionaryObject.Remove(TKey) will remove the element with TKey

            //return false if it fails.

            Console.WriteLine("1.10. DictionaryObject.Remove(TKey) ----------------");

            Console.WriteLine($"dictionaryGamers.Count = {dictionaryGamers.Count}");

            Console.WriteLine(dictionaryGamers.Remove("Key") ?

                "The element has been removed." :

                "Key can not be found, so no element has been removed.");

            Console.WriteLine($"dictionaryGamers.Count = {dictionaryGamers.Count}");

            Console.WriteLine(dictionaryGamers.Remove("Key5") ?

                "The element has been removed." :

                "Key can not be found, so no element has been removed.");

            Console.WriteLine($"dictionaryGamers.Count = {dictionaryGamers.Count}");

            //1.10.DictionaryObject.Remove(TKey)----------------

            //dictionaryGamers.Count = 5

            //Key can not be found, so no element has been removed.

            //dictionaryGamers.Count = 5

            //The element has been removed.

            //dictionaryGamers.Count = 4

            // 1.11. ------------------------------------------------

            // DictionaryObject.Clear() remove all elements from the Dictionary.

            Console.WriteLine("1.11. DictionaryObject.Clear() ----------------");

            Console.WriteLine($"dictionaryGamers.Count = {dictionaryGamers.Count}");

            dictionaryGamers.Clear();

            Console.WriteLine($"dictionaryGamers.Count = {dictionaryGamers.Count}");

           // 1.12. ------------------------------------------------

            // ArrayObject.ToDictionary() will convert array to dictionary.

            Console.WriteLine("1.12. ArrayObject.ToDictionary() ----------------");

            Gamer[] customersArr = {

                new Gamer {Id = 1, Name = "Name1", Email = "[1@1.com](mailto:1@1.com)"},

                new Gamer {Id = 2, Name = "Name2", Email = "[2@2.com](mailto:2@2.com)"},

                new Gamer {Id = 3, Name = "Name3", Email = "[3@3.com](mailto:3@3.com)" },

                new Gamer {Id = 4, Name = "Name4", Email = "[4@4.com](mailto:4@4.com)" }

            };

            Console.WriteLine("1.12.1. dictionaryGamers1 ----------------");

            //ArrayObject.ToDictionary() will convert array to dictionary.

            //In this case, Key is Id, Value is Gamer Object.

            Dictionary<int, Gamer> dictionaryGamers1 =

                customersArr.ToDictionary(gamerItem => gamerItem.Id, gamerItem => gamerItem);

            foreach (KeyValuePair<int, Gamer> item in dictionaryGamers1)

            {

                Console.WriteLine($"item.Key == {item.Key}   ;   item.Value : {item.Value}");

            }

            //1.12.1.dictionaryGamers1----------------

            //item.Key == 1; item.Value : Id == 1; Name == Name1; Email: [1@1.com](mailto:1@1.com)

            //item.Key == 2; item.Value : Id == 2; Name == Name2; Email: [2@2.com](mailto:2@2.com)

            //item.Key == 3; item.Value : Id == 3; Name == Name3; Email: [3@3.com](mailto:3@3.com)

            //item.Key == 4; item.Value : Id == 4; Name == Name4; Email: [4@4.com](mailto:4@4.com)

            Console.WriteLine("1.12.2. dictionaryGamers2 ----------------");

            //ArrayObject.ToDictionary() will convert array to dictionary.

            //In this case, Key is Id, Value is Gamer Object.

            Dictionary<int, Gamer> dictionaryGamers2 =

                customersArr.ToDictionary(gamerItem => gamerItem.Id);

            foreach (KeyValuePair<int, Gamer> item in dictionaryGamers1)

            {

                Console.WriteLine($"item.Key == {item.Key}   ;   item.Value : {item.Value}");

            }

            //1.12.2.dictionaryGamers2----------------

            //item.Key == 1; item.Value : Id == 1; Name == Name1; Email: [1@1.com](mailto:1@1.com)

            //item.Key == 2; item.Value : Id == 2; Name == Name2; Email: [2@2.com](mailto:2@2.com)

            //item.Key == 3; item.Value : Id == 3; Name == Name3; Email: [3@3.com](mailto:3@3.com)

            //item.Key == 4; item.Value : Id == 4; Name == Name4; Email: [4@4.com](mailto:4@4.com)

            Console.WriteLine("1.12.3. dictionaryGamers3 ----------------");

            //In this case, Key is Id, Value is Gamer Object.

            Dictionary<int, Gamer> dictionaryGamers3 = new Dictionary<int, Gamer>();

            foreach (Gamer gamerItem in customersArr)

            {

                dictionaryGamers3.Add(gamerItem.Id, gamerItem);

            }

            foreach (KeyValuePair<int, Gamer> item in dictionaryGamers1)

            {

                Console.WriteLine($"item.Key == {item.Key}   ;   item.Value : {item.Value}");

            }

            //1.12.3.dictionaryGamers3----------------

            //item.Key == 1; item.Value : Id == 1; Name == Name1; Email: [1@1.com](mailto:1@1.com)

            //item.Key == 2; item.Value : Id == 2; Name == Name2; Email: [2@2.com](mailto:2@2.com)

            //item.Key == 3; item.Value : Id == 3; Name == Name3; Email: [3@3.com](mailto:3@3.com)

            //item.Key == 4; item.Value : Id == 4; Name == Name4; Email: [4@4.com](mailto:4@4.com)

            Console.WriteLine("1.12.4. dictionaryGamers4 ----------------");

            //In this case, Key is Id, Value is Gamer Object.

            Dictionary<int, Gamer> dictionaryGamers4 = new Dictionary<int, Gamer>();

            for (int index = 0; index < customersArr.Length; index++)

            {

                Gamer gamerItem = customersArr[index];

                dictionaryGamers4.Add(gamerItem.Id, gamerItem);

            }

            foreach (KeyValuePair<int, Gamer> item in dictionaryGamers1)

            {

                Console.WriteLine($"item.Key == {item.Key}   ;   item.Value : {item.Value}");

            }

            //1.12.4.dictionaryGamers4----------------

            //item.Key == 1; item.Value : Id == 1; Name == Name1; Email: [1@1.com](mailto:1@1.com)

            //item.Key == 2; item.Value : Id == 2; Name == Name2; Email: [2@2.com](mailto:2@2.com)

            //item.Key == 3; item.Value : Id == 3; Name == Name3; Email: [3@3.com](mailto:3@3.com)

            //item.Key == 4; item.Value : Id == 4; Name == Name4; Email: [4@4.com](mailto:4@4.com)

        }

    }

}

namespace OnLineGame

{

    // 1. ---------------------------------------------

    public class Gamer

    {

        public int Id { get; set; }

        public string Name { get; set; }

        public string Email { get; set; }

        public override string ToString()

        {

            return $"Id == {Id} ; Name == {Name} ; Email : {Email}";

        }

    }

}

Text

Description automatically generated

