Multidimensional-Array

1. Hard code

using System;

namespace ConsoleApp1

{

internal class Program

{

static void Main(string[] args)

{

string[][] users = new string[4][];

users[0] = new string[4];

users[0][0] = "S.NO";

users[0][1] = "Names";

users[0][2] = "ID'S";

users[0][3] = "Contact";

users[1] = new string[4];

users[1][0] = "1";

users[1][1] = "Awais";

users[1][2] = "23003";

users[1][3] = "03247900621";

users[2] = new string[4];

users[2][0] = "2";

users[2][1] = "Aliza";

users[2][2] = "22429";

users[2][3] = "02347485931";

users[3] = new string[4];

users[3][0] = "3";

users[3][1] = "Hamza";

users[3][2] = "22429";

users[3][3] = "02462904177";

// Concatenate and print the values horizontally

Console.WriteLine(string.Join(", ", users[0]));

Console.WriteLine(string.Join(", ", users[1]));

Console.WriteLine(string.Join(", ", users[2]));

Console.WriteLine(string.Join(", ", users[3]));

Console.Read();

Console.Read();

}

}

}

1. Multidimensinal array

Using for loop

Without nested loop

Just using --> string.Join method

using System;

namespace ConsoleApp1

{

internal class Program

{

static void Main(string[] args)

{

string[][] users = new string[4][];

users[0] = new string[4];

users[0][0] = "S.NO";

users[0][1] = "Names";

users[0][2] = "ID'S";

users[0][3] = "Contact";

users[1] = new string[4];

users[1][0] = "1";

users[1][1] = "Awais";

users[1][2] = "23003";

users[1][3] = "03247900621";

users[2] = new string[4];

users[2][0] = "2";

users[2][1] = "Aliza";

users[2][2] = "22429";

users[2][3] = "02347485931";

users[3] = new string[4];

users[3][0] = "3";

users[3][1] = "Hamza";

users[3][2] = "22429";

users[3][3] = "02462904177";

// Print the values horizontally without nested loops

for (int i = 0; i < users.Length; i++)

{

// 2 argu dena must hy

Console.WriteLine(string.Join("\t", users[i]));

}

Console.Read();

}

}

}

1. Multidimensinal-array

Using for each loop

Without nested loop

using System;

internal class Program

{

private static void Main(string[] args)

{

string[][] users = new string[4][];

users[0] = new string[4];

users[0][0] = "S.NO";

users[0][1] = "Names";

users[0][2] = "ID'S";

users[0][3] = "Contact";

users[1] = new string[4];

users[1][0] = "1";

users[1][1] = "Awais";

users[1][2] = "23003";

users[1][3] = "03247900621";

users[2] = new string[4];

users[2][0] = "2";

users[2][1] = "Aliza";

users[2][2] = "22429";

users[2][3] = "02347485931";

users[3] = new string[4];

users[3][0] = "3";

users[3][1] = "Hamza";

users[3][2] = "22429";

users[3][3] = "02462904177";

// Print the values horizontally without nested loops

foreach (var item in users)

{

Console.WriteLine(string.Join("\t", item));

}

Console.Read();

}

}

1. Multidimensinal array

With nested loop

using System;

namespace ConsoleApp1

{

internal class Program

{

static void Main(string[] args)

{

string[][] users = new string[4][];

users[0] = new string[4];

users[0][0] = "S.NO";

users[0][1] = "Names";

users[0][2] = "ID'S";

users[0][3] = "Contact";

users[1] = new string[4];

users[1][0] = "1";

users[1][1] = "Awais";

users[1][2] = "23003";

users[1][3] = "03247900621";

users[2] = new string[4];

users[2][0] = "2";

users[2][1] = "Aliza";

users[2][2] = "22429";

users[2][3] = "02347485931";

users[3] = new string[4];

users[3][0] = "3";

users[3][1] = "Hamza";

users[3][2] = "22429";

users[3][3] = "02462904177";

// Print the values horizontally without using string.Join

for (int i = 0; i < users.Length; i++)

{

for (int j = 0; j < users[i].Length; j++)

{

Console.Write(users[i][j] + "\t");

}

Console.WriteLine();

}

Console.Read();

}

}

}

1. **Multidimensinal-array**

**Short method**

**Without using nested loop**

using System;

namespace ConsoleApp1

{

internal class Program

{

static void Main(string[] args)

{

string[][] users = new string[4][];

users[0] = new string[4] { "S.NO", "Names", "ID'S", "Contact" };

users[1] = new string[4] { "1", "Awais", "23003", "03247900621" };

users[2] = new string[4] { "2", "Aliza", "22429", "02347485931" };

users[3] = new string[4] { "3", "Hamza", "22429", "02462904177" };

foreach (var row in users)

{

Console.WriteLine(string.Join("\t", row));

}

Console.Read();

}

}

}

**Single dimensional array**

**Taking input**

using System;

class Program

{

static void Main(string[] args)

{

string[] names = new string[3];

Console.WriteLine("\*\*\*\*\*\*Input\*\*\*\*\*\*");

Console.WriteLine("Enter name at 0 index:");

names[0] = Console.ReadLine();

Console.WriteLine("Enter name at 1 index:");

names[1] = Console.ReadLine();

Console.WriteLine("Enter name at 2 index:");

names[2] = Console.ReadLine();

Console.WriteLine("\*\*\*\*\*\*Display\*\*\*\*\*\*");

Console.WriteLine("Name at 0 index: " + names[0]);

Console.WriteLine("Name at 1 index: " + names[1]);

Console.WriteLine("Name at 2 index: " + names[2]);

Console.Read();

}

}

**Unlimited data store in array**

using System;

using System.Collections.Generic;

class Program

{

static void Main(string[] args)

{

List<string> names = new List<string>();

Console.WriteLine("\*\*\*\*\*\*Input\*\*\*\*\*\*");

string input;

do

{

Console.Write("Enter a name (or 'exit' to stop): ");

input = Console.ReadLine();

if (input.ToLower() != "exit")

{

names.Add(input);

}

} while (input.ToLower() != "exit");

Console.WriteLine("\n\*\*\*\*\*\*Display\*\*\*\*\*\*");

for (int i = 0; i < names.Count; i++)

{

Console.WriteLine($"Name at index {i}: {names[i]}");

}

Console.Read();

}

}

1. Task # 2 ( method in c# )

using System;

namespace MyApplication

{

class Program

{

static int MyMethod(int x)

{

return 5 \* x;

}

static void Main(string[] args)

{

Console.WriteLine("one product price is 5 rupees:");

Console.WriteLine("Enter a product quantity:");

int inputNumber = int.Parse(Console.ReadLine());

int result = MyMethod(inputNumber);

Console.WriteLine("Total Product price is: " + result);

Console.Read();

}

}

}

1. **Push and pop**

using System;

using System.Collections;

namespace DemoApplication

{

class Program

{

static void Main(string[] args)

{

Stack stackVar = new Stack();

stackVar.Push(1);

stackVar.Push(2);

stackVar.Push(3);

stackVar.Pop();

stackVar.Pop();

foreach (var storeValue in stackVar)

{

Console.WriteLine(storeValue);

}

Console.Read();

}

}

}

1. **Push, pop, count and peek method**

using System.Collections;

namespace ConsoleApp1

{

internal class Program

{

static void Main(string[] args)

{

//push -> add method

Stack obj = new Stack();

obj.Push(1);

obj.Push(2);

obj.Push(3);

obj.Push(4);

obj.Push(5);

//top of the value in stack, using peek method

Console.WriteLine(" Using peek method");

Console.WriteLine("Top of the value in stack : " + obj.Peek() + "\n");

//before pop method , calculate total elements , using count method

Console.WriteLine("Using count method");

Console.WriteLine("Before, total elements are calculate : " + obj.Count + "\n");

//pop -> delete method

Console.WriteLine("Using pop method");

obj.Pop();

//after pop method , calculate total elements , using count method

Console.WriteLine("After, total elements are calculate : " + obj.Count + "\n");

// using loop

Console.WriteLine("calculate total value in stack ");

foreach (int store\_box in obj)

{

Console.WriteLine("push elements " +store\_box);

}

Console.Read();

}

}

}

using System;

namespace ConsoleApp1

{

internal class Program

{

static void Main(string[] args)

{

string[] chene = { "awais", "shaikh", "faiza", "talha" };

foreach (string daba in chene)

{

Console.WriteLine(daba);

}

Console.Read();

}

}

}