

Day 13 Assignment

By

Vinay Kudali

09-02-22

nations benefits

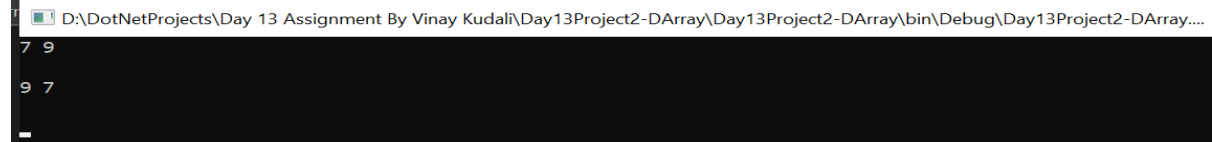
1. Declare a 2 dimensional array of size (2,2) and initialize using indexes and print the values using nested for loop

Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day13Project2_DArray
{
    internal class Program
    {
        //Author: Vinay Kudali
        //Purpose: Creating 2-D Array
        static void Main(string[] args)
        {
            int[,] data = new int[2,2];
            data[0, 0] = 7;
            data[0, 1] = 9;
            data[1, 0] = 9;
            data[1, 1] = 7;

            for(int i=0;i<2;i++)
            {
                for (int j = 0; j < 2; j++)
                {
                    Console.Write(data[i,j] + " ");
                }
                Console.WriteLine("\n");
            }
            Console.ReadLine();
        }
    }
}
```

Output:

```
D:\DotNetProjects\Day 13 Assignment By Vinay Kudali\Day13Project2-DArray\Day13Project2-DArray\bin\Debug\Day13Project2-DArray....
7 9
9 7
```

2. Declare a 2-D array of size (3,2) and initialize in the same line while declaring and print the values using nested for loop**Code:**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day13Project2_DArray
{
    internal class Program
    {
        //Author: Vinay Kudali
        //Purpose: Creating 2-D Array and initializing in same Line
        static void Main(string[] args)
        {
            int[,] data = new int[,] { { 7, 9 }, { 9, 7 }, { 8, 2 } };

            for (int i = 0; i < 3; i++)
            {
                for (int j = 0; j < 2; j++)
                {
                    Console.Write(data[i, j] + " ");
                }
                Console.WriteLine("\n");
            }
            Console.ReadLine();
        }
    }
}
```

Output:

```
D:\DotNetProjects\Day 13 Assignment By Vinay Kudali\Day13Project2,2-D ArrayWithSamelineInitialisation\D
7 9
9 7
8 2
```

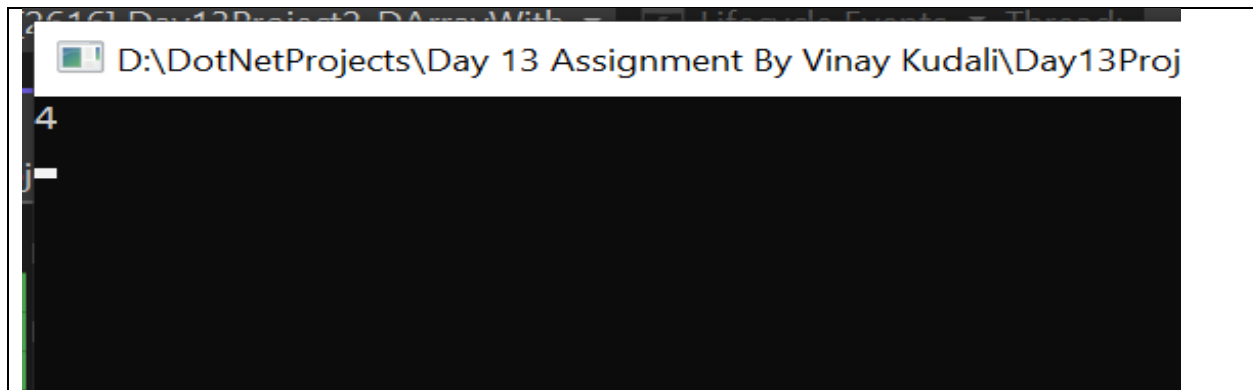
3. Declare a 2-D array of size (3,3) and print trace of the array

Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day13Project2_DArrayWithTrace
{
    internal class Program
    {
        //Author: Vinay Kudali
        //Purpose Creating 2-D Array With sum of trace (Diagnoal)
        static void Main(string[] args)
        {
            int sum = 0;
            int[,] data = new int[,] { { 1, 2, 3 }, { 3, 2, 1 }, { 2, 3, 1 } };
            for (int i = 0; i < 3; i++)
            {
                for (int j = 0; j < 3; j++)
                {
                    if (i == j)
                        sum = sum + data[i, j];
                }
            }
            Console.WriteLine(sum);
            Console.ReadLine();
        }
    }
}
```

Output:



4. Declare a 2-D array of size (2,2) and read values from user and print the array values.

Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day13Project42_DArrayWithreadDataFromUser
{
    internal class Program
    {
        //Author: Vinay Kudali
        //Purpose: 2-D Array using Read Data And Print Data
        static void Main(string[] args)
        {
            int[,] data = new int[2,2];
            //Read Data From User
            for (int i = 0; i < 2; i++)
            {
                for (int j = 0; j < 2; j++)
                {
                    Console.WriteLine("Enter Array Value at: " + data [i,j]);
                    data[i, j] = Convert.ToInt32(Console.ReadLine());
                }
            }
            //Print data From User

            for (int i = 0; i < 2; i++)
            {
                for (int j = 0; j < 2; j++)
                {
```

```

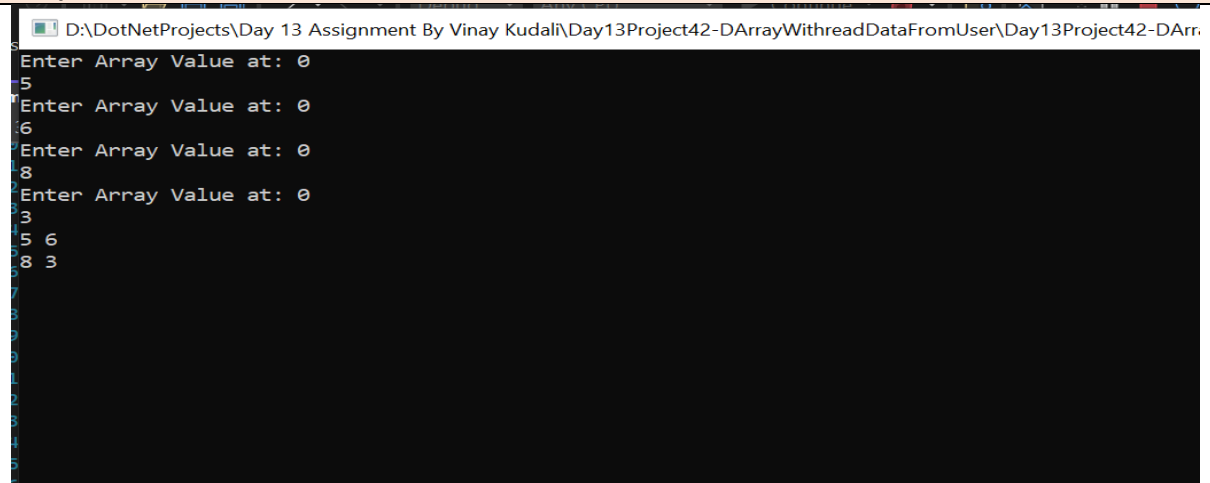
        Console.Write(data[i, j] + " ");
    }
    Console.WriteLine();

}
Console.ReadLine();

}
}
}

```

Output:



The screenshot shows a Windows command prompt window with the following text:

```

D:\DotNetProjects\Day 13 Assignment By Vinay Kudali\Day13Project42-DArrayWithreadDataFromUser\Day13Project42-DArr
Enter Array Value at: 0
5
Enter Array Value at: 0
6
Enter Array Value at: 0
8
Enter Array Value at: 0
3
5 6
8 3

```

5. Declare TWO 2-D arrays of size (2,2) and read values from user and print the sum of the two matrices.

Code:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day13Project5SumOfMatrices
{
    internal class Program
    {
        //Author: Vinay Kudali
        //Purpose Sum of two Matrices
        static void Main(string[] args)

```

```

{
    int[,] x = new int[2, 2];
    int[,] y = new int[2, 2];
    int[,] sum = new int[2,2];

    //Read Data From User for x matrices
    for (int i = 0; i < 2; i++)
    {
        for (int j = 0; j < 2; j++)
        {
            Console.WriteLine("Enter An Array Value 'a' :");
            x[i, j] = Convert.ToInt32(Console.ReadLine());
        }
    }
    //read Data from User for y matrices
    for (int i = 0; i < 2; i++)
    {
        for (int j = 0; j < 2; j++)
        {
            Console.WriteLine("Enter An Array Value 'b' :");
            y[i, j] = Convert.ToInt32(Console.ReadLine());
        }
    }

    //Additon Of matrices
    for (int i = 0; i < 2; i++)
    {
        for (int j = 0; j < 2; j++)
        {
            sum[i, j] = x[i, j] + y[i, j];

            Console.Write(sum[i, j] + " ");
        }
        Console.WriteLine();
    }
    Console.ReadLine();
}
}
}

```

Output:

```
D:\DotNetProjects\Day 13 Assignment By Vinay Kudali\Day13Project5SumOfMatrices\Day
Enter An Array Value 'a' :
1
Enter An Array Value 'a' :
2
Enter An Array Value 'a' :
3
Enter An Array Value 'a' :
4
Enter An Array Value 'b' :
4
Enter An Array Value 'b' :
3
Enter An Array Value 'b' :
2
Enter An Array Value 'b' :
1
5 5
5 5
```

7. What is a jagged array. What is the benefit of jagged array

Jagged Array:

Jagged Array is the Array in C# which will have different sizes for Different rows and columns.

Benefits of jagged Array:

- ❖ Array Memory will not be wasted
- ❖ We can increase size dynamically.

8. WACP to declare a jagged array and print values

Code:

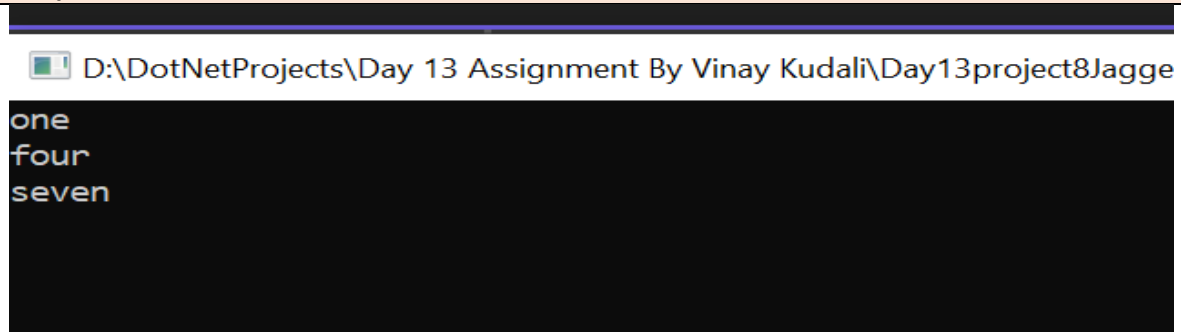
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day13project8JaggedArray
{
    internal class Program
    {
```

```
//Author : Vinay Kudali
//Purpose: jagged array with printing values
static void Main(string[] args)
{
    char[][] names = new char[3][];
    names[0] = new char[] { 'o', 'n', 'e' };
    names[1] = new char[] { 'f', 'o', 'u', 'r' };
    names[2] = new char[] { 's', 'e', 'v', 'e', 'n' };

    for(int i=0; i<3; i++)
    {
        for(int j = 0; j<names[i].Length; j++)
        {
            Console.Write(names[i][j]);
        }
        Console.WriteLine();
    }
    Console.ReadLine();
}
}
```

Output:



```
D:\DotNetProjects\Day 13 Assignment By Vinay Kudali\Day13project8Jagge
one
four
seven
```

9. What is Recursion

Recursion is a function calling itself repeatedly until specific condition satisfies.

10. WACP to illustrate usage of Recursion. What are the benefits of recursion

Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
```



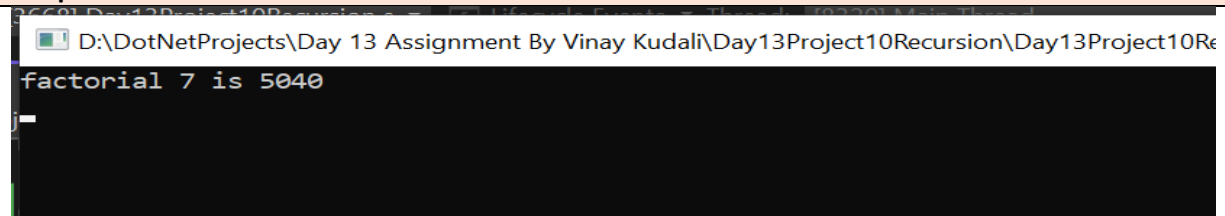
```

using System.Text;
using System.Threading.Tasks;

namespace Day13Project10Recursion
{
    internal class Program
    {
        //Author: Vinay Kudali
        //Purpose: Factorial using Recursion
        public static int Factorial(int n)
        {
            if (n == 0)
                return 1;
            else
                return n * Factorial(n - 1);
        }
        public static void Print(int n)
        {
            Console.WriteLine("factorial {0} is {1}", n, Factorial(n));
        }
        static void Main(string[] args)
        {
            int n = 7;
            Print(n);
            Console.ReadLine();
        }
    }
}

```

Output:



D:\DotNetProjects\Day 13 Assignment By Vinay Kudali\Day13Project10Recursion\Day13Project10Re

factorial 7 is 5040

11. WACP to illustrate usage of Stack<> Write couple of points about Stack

Code:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

```

namespace Day13project7
{
    class Program
    {
        static void Main(string[] args)
        {

            //Author: Vinay Kudali
            //Purpose:performing operation by using stack

            Stack<int> data = new Stack<int>();
            data.Push(74);
            data.Push(91);
            data.Push(32);
            data.Push(53);
            Console.WriteLine(data.Count);
            Console.WriteLine(data.Peek());
            Console.WriteLine(data.Pop());
            Console.WriteLine(data.Count());
            Console.ReadLine();
        }
    }
}

```

Output:



The screenshot shows a console window with the following output:

```

4
53
53
3

```

The output corresponds to the operations performed in the code: Count (4), Peek (53), Pop (53), and Count (3).

12. WACP to illustrate usage of Queue<> Write couple of points about Stack

Code:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day13project8

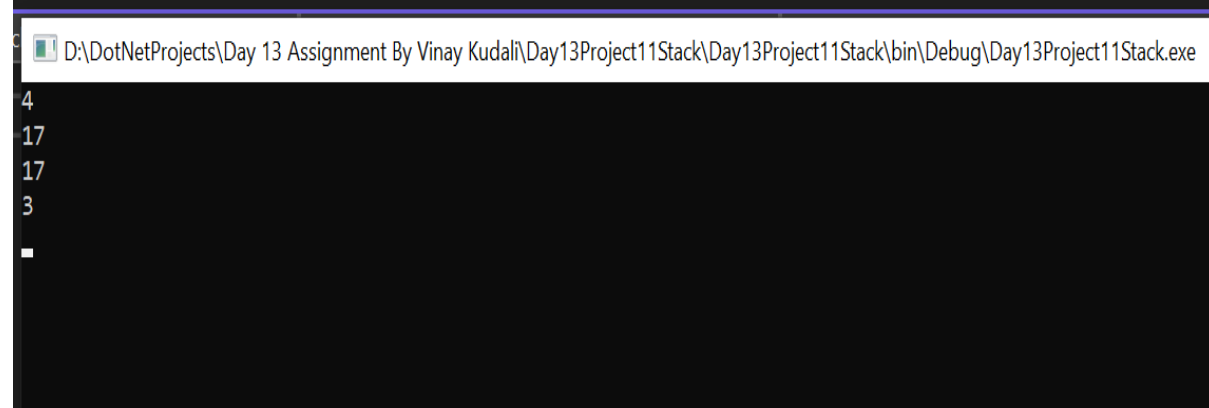
```

```
{
    class Program
    {
        static void Main(string[] args)
        {

            //Author:Vinay Kudali
            //Purpose:performing operation by using queue

            Queue<int> data = new Queue<int>();
            data.Enqueue(17);
            data.Enqueue(19);
            data.Enqueue(21);
            data.Enqueue(97);
            Console.WriteLine(data.Count);
            Console.WriteLine(data.Peek());
            Console.WriteLine(data.Dequeue());
            Console.WriteLine(data.Count());
            Console.ReadLine();
        }
    }
}
```

Output:



The screenshot shows a console window titled "D:\DotNetProjects\Day 13 Assignment By Vinay Kudali\Day13Project11Stack\Day13Project11Stack\bin\Debug\Day13Project11Stack.exe". The output consists of five lines: "4", "17", "17", "3", and a blank line. The first line "4" represents the count of the queue after enqueuing four elements. The second line "17" is the value of the first element (17) as returned by the Peek() method. The third line "17" is the value of the first element (17) as returned by the Dequeue() method. The fourth line "3" represents the count of the queue after dequeuing the first element. The fifth line is a blank line, likely the result of the ReadLine() call.