

Ex. No. 08	MULTITHREADING AND SYNCHRONIZATION
10.10.2023	

Aim

To develop C# console application using Multithreading and Synchronization concepts.

Description

Multithreading:

- Allows a process to manage two or more concurrent threads
- Each can handle a task independently
- Implemented under System.Threading and Thread class.
- Executing a function passing it to thread class object created
- Explicitly call Start() function in order to run the thread
- Join() makes the other threads to wait for the running thread.

Synchronization:

Locks the shared object which makes to access by only one thread at a time until it gets completed.

Syntax:

```
Thread th= new Thread(<function_name>);
```

```
th.Start();
```

```
lock(object){ //lock block of statements }
```

Source Code

A 1.

```
using System;

using System.Threading;

namespace Ex8{

    internal class Question1 {

        public static string para;

        public static void word_printing(){

            string[] li=para.Split(' ');

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

                Console.Write(li[i]+" ");

                Thread.Sleep(2000);}

        public static void vowels_printing(){

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

                if ("aeiouAEIOU".Contains(para[i])){

                    Console.Write(para[i]+" ");}}}

        static void Main(string[] args){

            Console.WriteLine("Enter any Paragraph");

            para=Console.ReadLine();

            Thread t1=new Thread(word_printing);

            Thread t2=new Thread(vowels_printing);

            t1.Start();

            t1.Join();
```

```
t2.Start();  
  
t2.Join();  
  
Console.ReadKey();}}}
```

A 2.

```
using System;  
  
using System.Threading;  
  
namespace Ex8{  
  
    internal class Question2{  
  
        public static int[] arr;  
  
        public static void even_printing(){  
  
            Console.WriteLine("Displaying Even Numbers:");{  
  
                if (arr[i]%2 == 0){  
  
                    Console.Write(arr[i]+", ");}}  
  
            Thread.Sleep(2000);}  
  
        public static void odd_printing(){  
  
            Console.WriteLine("\nDisplaying Odd Numbers:");  
  
            for (int i = 0; i < arr.Length; i++){  
  
                if (arr[i] % 2 != 0){  
  
                    Console.Write(arr[i] + ", ");}}}  
  
        static void Main(string[] args){  
  
            arr=new int[10];  
  
            Console.WriteLine("Enter Array Elements: ");  
  
            for(int i = 0; i < arr.Length; i++){  
  
                arr[i]=Convert.ToInt32(Console.ReadLine());}
```

```
Thread t1=new Thread(even_printing);

Thread t2=new Thread(odd_printing);

t1.Start();

t1.Join();

t2.Start();

t2.Join();

Console.ReadKey();}}}
```

B.

```
using System;

using System.Threading;

namespace Ex8{

    internal class Question3{

        public static int num1, num2,num3;

        public void table_printing(){

            lock (this){

                Console.WriteLine("\nPrinting Table:");

                for (int i = 1; i <= 10; i++){

                    if (Thread.CurrentThread.Name == "num1_t") { Console.WriteLine(num1 + "x"

+ i + "=" + num1 * i); }

                    if (Thread.CurrentThread.Name == "num2_t") { Console.WriteLine(num2 + "x"

+ i + "=" + num2 * i); }

                    if (Thread.CurrentThread.Name == "num3_t") { Console.WriteLine(num3 + "x"

+ i + "=" + num3 * i); }}

                Thread.Sleep(2000);}}

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

```
Console.Write("Enter Num1: ");  
  
num1 = Convert.ToInt32(Console.ReadLine());  
  
Console.Write("Enter Num2: ");  
  
num2 = Convert.ToInt32(Console.ReadLine());  
  
Console.Write("Enter Num3: ");  
  
num3 = Convert.ToInt32(Console.ReadLine());  
  
Question3 c=new Question3();  
  
Thread num1_t=new Thread(c.table_printing);  
  
Thread num2_t = new Thread(c.table_printing);  
  
Thread num3_t = new Thread(c.table_printing);  
  
num1_t.Name = "num1_t";  
  
num2_t.Name = "num2_t";  
  
num3_t.Name = "num3_t";  
  
num1_t.Start();  
  
num2_t.Start();  
  
num3_t.Start();  
  
Console.ReadKey();}}}
```

Output

A 1.

```
Enter any Paragraph  
Hello World Welcome to C# Programming  
Hello, World, Welcome, to, C#, Programming, e, o, o, e, o, e, o, o, a, i,
```

A 2.

```
Enter Array Elements:
1
2
3
4
5
6
7
8
9
10
Displaying Even Numbers:
2, 4, 6, 8, 10,
Displaying Odd Numbers:
1, 3, 5, 7, 9,
```

B.

```
Enter Num1: 5
Enter Num2: 10
Enter Num3: 3

Printing Table:
5x1=5
5x2=10
5x3=15
5x4=20
5x5=25
5x6=30
5x7=35
5x8=40
5x9=45
5x10=50

Printing Table:
3x1=3
3x2=6
3x3=9
3x4=12
3x5=15
3x6=18
3x7=21
3x8=24
3x9=27
3x10=30

Printing Table:
10x1=10
10x2=20
10x3=30
10x4=40
10x5=50
10x6=60
10x7=70
10x8=80
10x9=90
10x10=100
```

Result

The C# console application using Multithreading and Synchronization concepts has been executed successfully and the desired output is displayed on the screen.