

# c# Internal Exam Answers

## Q3.

//3) Write a delegate which will take one integer argument. And write functions for Factorial, Fibonacci, Factors of number, Table of Number, //square and cube of given number. Call all the functions through delegate using menu driven approach.

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

```
namespace internalexam
{
```

```
    delegate void Cal(int n);
```

```
    internal class q3
```

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

```
        {
            choice:
            char choice;
```

```
            Console.WriteLine("1.Factorial");
            Console.WriteLine("2.Fibonacci");
            Console.WriteLine("3.Factors");
            Console.WriteLine("4.Tables");
```

```
            Console.Write("\nEnter your choice : ");
            choice = char.Parse(Console.ReadLine());
            int num;
```

```
            switch (choice)
            {
```

```
                case '1':
                    Console.Write("Enter the number : ");
                    num = int.Parse(Console.ReadLine());
                    Cal cal = new Cal(Factorial);
                    cal.Invoke(num);
                    Console.WriteLine("\nPress any key to continue.");
                    Console.ReadLine();
                    Console.Clear();
                    goto choice;
```

```
                case '2':
                    Console.Write("Enter the number : ");
                    num = int.Parse(Console.ReadLine());
                    Cal cal_1 = new Cal(Fibonacci);
                    cal_1.Invoke(num);
                    Console.WriteLine("\nPress any key to continue.");
                    Console.ReadLine();
                    Console.Clear();
                    goto choice;
```

```
                case '3':
```

```

        Console.Write("Enter the number : ");
        num = int.Parse(Console.ReadLine());
        Cal cal_2 = new Cal(Factors);
        cal_2.Invoke(num);
        Console.WriteLine("\nPress any key to continue.");
        Console.ReadLine();
        Console.Clear();
        goto choice;
    case '4':
        Console.Write("Enter the number : ");
        num = int.Parse(Console.ReadLine());
        Cal cal_3 = new Cal(Tables);
        cal_3.Invoke(num);
        //Console.WriteLine("\nPress any key to continue.");
        Console.ReadLine();
        Console.Clear();
        goto choice;
    default:
        break;
}
}

private static void Factorial(int n)
{
    int factorial = 1;

    Console.Write("Factorial of {0} : ", n);
    for (int i = 1; i <= n; i++)
    {
        factorial *= i;
        Console.Write("{0} ", factorial);
    }
}

private static void Fibonacci(int n)
{
    Console.WriteLine("Fibonacci series upto {0} : ", n);
    int a = 0, b = 1, c;
    Console.Write("{0} {1} ", a, b);
    for (int i = 2; i < n; i++)
    {
        c = a + b;

        a = b;
        b = c;

        Console.Write("{0} ", c);
    }
}

private static void Tables(int n)
{
    Console.WriteLine("Multiplication table for {0} ", n);
    for (int i = 1; i <= 10; i++)
    {
        int result = i * n;
        Console.WriteLine("{0} X {1} = {2}", i, n, result);
    }
}

```

```

private static void Factors(int n)
{
    Console.WriteLine("Factors of {0} : ", n);

    for (int i = 1; i <= n; i++)
    {
        if (n % i == 0)
        {
            Console.WriteLine("{0} ", i);
        }
    }
}
}
}

```

```

C:\Users\dattateja\source\repos\internalexam\internalexam\bin\Debug\net6.0\internalexam.exe
1. Factorial
2. Fibonacci
3. Factors
4. Tables

Enter your choice : 4
Enter the number : 7
Multiplication table for 7
1 X 7 = 7
2 X 7 = 14
3 X 7 = 21
4 X 7 = 28
5 X 7 = 35
6 X 7 = 42
7 X 7 = 49
8 X 7 = 56
9 X 7 = 63
10 X 7 = 70

```

## Q4

//4) Ask user to input account\_no, account\_holder\_name, branch, IFSC\_code and account\_type.

Create a XML file(bank.xml) to store the data entered

//by 3 users. Read the same file and show the data.

using System;

using System.Collections;

using System.Xml;

namespace internalExam

{

internal class q4

{

public static void Main()

{

```

start:
    string fileName;
    char choice;

    Console.WriteLine("1 creat xml file");
    Console.WriteLine("2 read xml file");

    Console.Write("Enter the choices : ");

    choice = char.Parse(Console.ReadLine());

    Console.Write("Enter the file Name : ");
    fileName = Console.ReadLine();

    string path = $"C:\\teja\\{fileName}.xml";

    switch (choice)
    {
        case '1':
            CreateXML(path);
            Console.WriteLine("Press any key to continue..");
            Console.ReadLine();
            Console.Clear();
            goto start;
        case '2':
            ReadXML(path);
            Console.WriteLine("Press any key to continue.");
            Console.ReadLine();
            Console.Clear();
            goto start;
        case '0':
            break;
        default:
            Console.WriteLine("Invalid input....");
            Console.WriteLine("Press any key to continue.");
            Console.ReadLine();
            Console.Clear();
            goto start;
    }

}

private static void ReadXML(string path)
{
    try
    {
        XmlTextReader xmlReader = new XmlTextReader(path);

        ArrayList acNoList = new ArrayList();
        ArrayList acHolderList = new ArrayList();
        ArrayList branchList = new ArrayList();
        ArrayList ifscList = new ArrayList();
        ArrayList acTypeList = new ArrayList();

        while (xmlReader.Read())
        {
            if (xmlReader.NodeType == XmlNodeType.Element && xmlReader.Name == "ac_no")

```

```

        {
            string accNo = xmlReader.ReadElementContentAsString();
            acNoList.Add(accNo);
        }
        if (xmlReader.NodeType == XmlNodeType.Element && xmlReader.Name ==
"ac_holder_name")
        {
            string accHolder = xmlReader.ReadElementContentAsString();
            acHolderList.Add(accHolder);
        }
        if (xmlReader.NodeType == XmlNodeType.Element && xmlReader.Name == "branch")
        {
            string branch = xmlReader.ReadElementContentAsString();
            branchList.Add(branch);
        }
        if (xmlReader.NodeType == XmlNodeType.Element && xmlReader.Name == "ifsc")
        {
            string ifsc = xmlReader.ReadElementContentAsString();
            ifscList.Add(ifsc);
        }
        if (xmlReader.NodeType == XmlNodeType.Element && xmlReader.Name ==
"ac_type")
        {
            string accType = xmlReader.ReadElementContentAsString();
            acTypeList.Add(accType);
        }
    }

    xmlReader.Close();
    Console.WriteLine("Read all the data.....");
    Console.WriteLine("Acc No    Acc Holder    Branch    IFSC    Acc Type");
    for (int i = 0; i < acNoList.Count; i++)
    {
        Console.WriteLine($"{acNoList[i]}\t\t{acHolderList[i]}\t\t{branchList[i]}\t\t{ifscList[i]}
\t\t{acTypeList[i]}");
    }
}
catch (Exception e)
{
    Console.WriteLine(e.Message);
}
}

private static void CreateXML(string path)
{
    Console.WriteLine("Create a xml file\n");
    int size;
    XmlTextWriter xmlWriter = new XmlTextWriter(path, System.Text.Encoding.UTF8);
    xmlWriter.Formatting = Formatting.Indented;

    xmlWriter.WriteStartDocument();

    xmlWriter.WriteStartElement("Bank");
    xmlWriter.WriteStartElement("Users");

    Console.Write("Enter the number of users : ");
    size = int.Parse(Console.ReadLine());

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

```

```

xmlWriter.WriteStartElement("user_" + i);

Console.WriteLine("\nEnter the user {0} details", i);
Console.Write("\nEnter the account number : ");
int accNo = int.Parse(Console.ReadLine());
xmlWriter.WriteElementString("acc_no", accNo.ToString());

Console.Write("Enter the account holder name : ");
string name = Console.ReadLine();
xmlWriter.WriteElementString("acc_holder_name", name);

Console.Write("Enter the branch name : ");
string branch = Console.ReadLine();
xmlWriter.WriteElementString("branch", branch);

Console.Write("Enter the IFSC code : ");
int ifsc = int.Parse(Console.ReadLine());
xmlWriter.WriteElementString("ifsc", ifsc.ToString());

Console.Write("Enter the account type : ");
string accType = Console.ReadLine();
xmlWriter.WriteElementString("acc_type", accType);

xmlWriter.WriteEndElement();
}

xmlWriter.WriteEndElement();
xmlWriter.WriteEndElement();

xmlWriter.WriteEndDocument();
xmlWriter.Close();

Console.WriteLine("Xml file successfully created....");
}
}
}

```

C:\Users\dattateja\source\repos\internalexam\internalexam\bin\Debug\net6.0\internalexam.exe

```

1 creat xml file
2 read xml file
Enter the choices : 1
Enter the file Name : teja
Create a xml file

Enter the number of users : 3

Enter the user 1 details

Enter the account number : 1234
Enter the account holder name : teja
Enter the branch name : hyd
Enter the IFSC code : 9089
Enter the account type : current

Enter the user 2 details

Enter the account number : 7668
Enter the account holder name : ram
Enter the branch name : pune
Enter the IFSC code : 6783
Enter the account type : savings

Enter the user 3 details

Enter the account number : 3746
Enter the account holder name : sai
Enter the branch name : hyd
Enter the IFSC code : 9848

```

```

▼<Bank>
  ▼<Users>
    ▼<user_1>
      <acc_no>1234</acc_no>
      <acc_holder_name>teja</acc_holder_name>
      <branch>hyd</branch>
      <ifsc>9089</ifsc>
      <acc_type>current</acc_type>
    </user_1>
    ▼<user_2>
      <acc_no>7668</acc_no>
      <acc_holder_name>ram</acc_holder_name>
      <branch>pune</branch>
      <ifsc>6783</ifsc>
      <acc_type>savings</acc_type>
    </user_2>
    ▼<user_3>
      <acc_no>3746</acc_no>
      <acc_holder_name>sai</acc_holder_name>
      <branch>hyd</branch>
      <ifsc>9848</ifsc>
      <acc_type>current</acc_type>
    </user_3>
  </Users>
</Bank>

```

## Q5

/\* Create a sealed class having two methods, one for swapping two numbers and another for checking given number is prime or not. Add three more methods to that class using extension method. first method will check given number is palidrome or not. second method will find sum of 1 to n number. and third will find the prime factor of given number.  
\*/

using System;

```

namespace internalexam
{
    public sealed class q5
    {
        public void Swap()
        {
            int a, b;
            Console.WriteLine("enter two numbers to swape : ");
            a = int.Parse(Console.ReadLine());
            b = int.Parse(Console.ReadLine());

            int temp = a;
            a = b;
            b = temp;

            Console.WriteLine("After swapping");
            Console.WriteLine("a:{0}", a);
            Console.WriteLine("b:{0}", b);
        }

        public void Prime(int n)

```

```

{
    bool isPrime = true;
    for (int i = 2; i <= n / 2; i++)
    {
        if (n % i == 0)
        {
            Console.WriteLine("{0} is not a prime number", n);
            isPrime = false;
            break;
        }

        isPrime = true;
    }

    if (isPrime)
    {
        Console.WriteLine("{0} is a prime number", n);
    }
}

public void IsPalindrome(int num)
{
    int r, sum = 0, temp;

    temp = num;

    while (num > 0)
    {
        r = num % 10;
        sum = (sum * 10) + r;
        num = num / 10;
    }

    if (temp == sum)
    {
        Console.WriteLine("{0} is palindrome", temp);
    }
    else
    {
        Console.WriteLine("{0} is not palindrome", temp);
    }
}

public void sum(int num)
{
    int sum = 0;
    for (int i = 1; i <= num; i++)
    {
        sum = sum + i;
    }
    Console.WriteLine("Sum of 1 to {0} numbers is {1}", num, sum);
}

public void primefactors(int num)
{
    bool prime = false;
    Console.WriteLine("prime factors of {0}: ", num);

    for (int i = 2; i <= num; i++)
    {
        if (num % i == 0)

```



```

        {
            prime = true;
            for (int j = 2; j <= i / 2; j++)
            {
                prime = false;
                break;
            }
        }
        if (prime)
        {
            Console.WriteLine("{0} ", i);
        }
    }
    Console.WriteLine();
}
}
}

```

```

internal class internalexam
{
    public static void Main()
    {
        q5 cm = new q5();
        int n;

        Console.WriteLine("\nenter the number for checking prime or not , palidrome ,prime
factor , sum : ");
        n = int.Parse(Console.ReadLine());

        cm.Swap();

        Console.WriteLine("\n");
        cm.Prime(n);
        Console.WriteLine("\n");
        cm.IsPalindrome(n);
        Console.WriteLine("\n");
        cm.Prime(n);
        Console.WriteLine("\n");
        cm.sum(n);
        Console.WriteLine("\n");
        cm.primefactors(n);
    }
}
}

```

```

Microsoft Visual Studio Debug Console

enter the number for checking prime or not , palidrome ,prime factor , sum :
7
enter two numbers to swape :
12
89
After swapping
a:89
b:12

7 is a prime number

7 is palidrome

7 is a prime number

Sum of 1 to 7 numbers is 28

prime factors of 7:

C:\Users\dattateja\source\repos\internalexam\internalexam\bin\Debug\net6.0\internalexam.exe (process 6380) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

```

## Q6

6) Write five functions, each function will print some text in specific colour. Create five different threads and run all threads simultaneously.

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

namespace internalexam
{
    class Q6
    {
        public static void text1()
        {
            for (int i = 0; i <= 5; i++)
            {
                Console.ForegroundColor = ConsoleColor.Yellow;
                Console.WriteLine("Hey there");
            }
        }

        public static void text2()
        {
            for (int i = 0; i <= 5; i++)
            {
                Console.ForegroundColor = ConsoleColor.Blue;
                Console.WriteLine("Hey , wassaup");
            }
        }

        public static void text3()
        {
            for (int i = 0; i <= 5; i++)
            {
                Console.ForegroundColor = ConsoleColor.Green;
                Console.WriteLine("nothing , what about u?");
            }
        }

        public static void text4()
        {
            for (int i = 0; i <= 5; i++)
            {
                Console.ForegroundColor = ConsoleColor.Red;
                Console.WriteLine("same here");
            }
        }

        public static void text5()
        {
            for (int i = 0; i <= 5; i++)
            {
                Console.ForegroundColor = ConsoleColor.Cyan;
                Console.WriteLine("ok..then Have a good day");
            }
        }

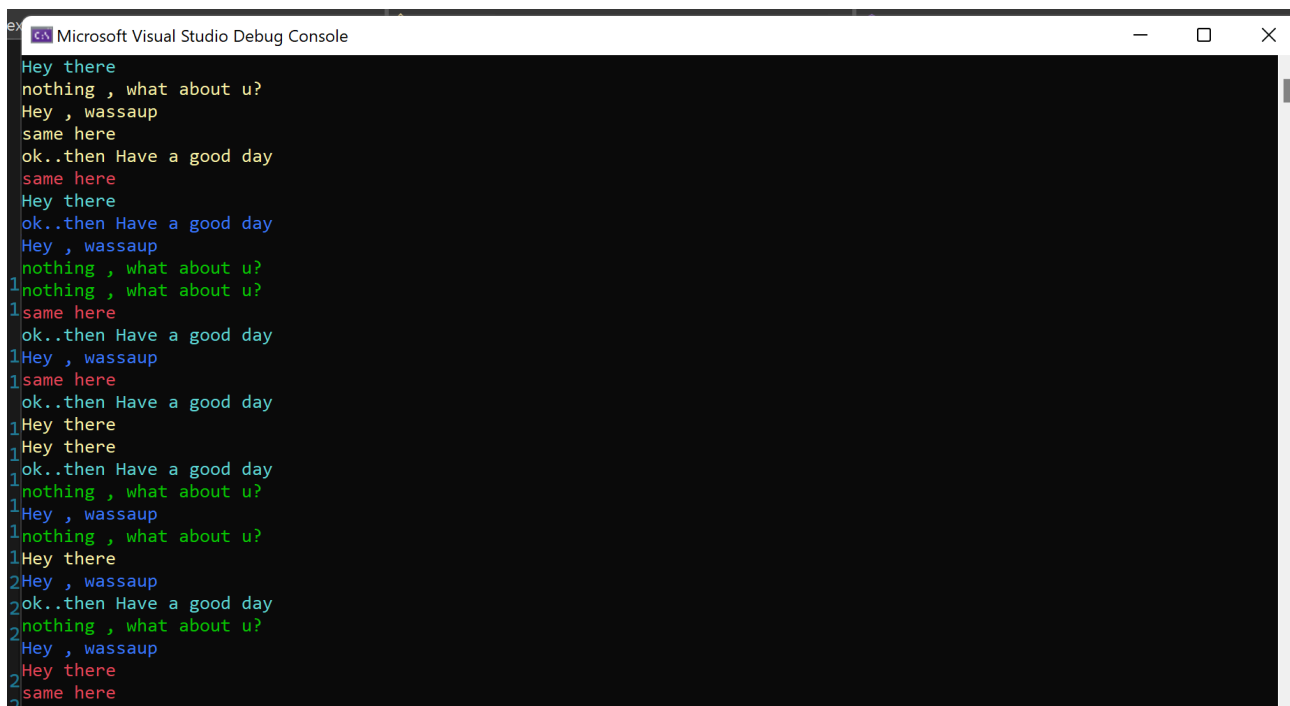
        static void Main()
    }
}
```

```

{
    Thread td1 = new Thread(text1);
    Thread td2 = new Thread(text2);
    Thread td3 = new Thread(text3);
    Thread td4 = new Thread(text4);
    Thread td5 = new Thread(text5);

    td1.Start();
    td2.Start();
    td3.Start();
    td4.Start();
    td5.Start();
}
}
}

```



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio Debug Console". The console output displays a multi-threaded chat application. The messages are interleaved, indicating concurrent execution. The messages are: "Hey there", "nothing , what about u?", "Hey , wassaup", "same here", and "ok..then Have a good day". The output is color-coded: "Hey there" is yellow, "nothing , what about u?" is green, "Hey , wassaup" is blue, "same here" is red, and "ok..then Have a good day" is blue. The messages are grouped into two sections, each preceded by a line number (1 and 2). The first section (lines 1-10) shows a sequence of messages from thread 1. The second section (lines 2-10) shows a sequence of messages from thread 2. The interleaving of messages from both threads demonstrates the concurrent execution of the threads.

```

Hey there
nothing , what about u?
Hey , wassaup
same here
ok..then Have a good day
same here
Hey there
ok..then Have a good day
Hey , wassaup
nothing , what about u?
1 nothing , what about u?
1 same here
ok..then Have a good day
1 Hey , wassaup
1 same here
ok..then Have a good day
1 Hey there
1 Hey there
ok..then Have a good day
1 nothing , what about u?
1 Hey , wassaup
1 nothing , what about u?
1 Hey there
2 Hey , wassaup
2 ok..then Have a good day
2 nothing , what about u?
2 Hey , wassaup
2 Hey there
2 same here
2

```