1. Write a C# Sharp program that takes three letters as input and display them in reverse order.

namespace Assignments

{

internal class Program

{

static void Main(string[] args)

{

char[] arr = new char[3];

try

{

Console.Write("Enter 1st letter : ");

arr[0] = Convert.ToChar(Console.ReadLine());

Console.Write("Enter 2nd letter : ");

arr[1] = Convert.ToChar(Console.ReadLine());

Console.Write("Enter 3rd letter : ");

arr[2] = Convert.ToChar(Console.ReadLine());

for (int i = (arr.Length)-1 ; i >= 0 ; i--)

Console.Write(arr[i]);

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

}

}

}

}

Text

Description automatically generated

1. Write a C# Sharp program that takes a number and a width also a number, as input and then displays a triangle of that width, using that number.

namespace Assignments

{

internal class Program

{

static void Main(string[] args)

{

try

{

Console.Write("Enter a number to print on console : ");

int num = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter a number for width of triangle : ");

int width = Convert.ToInt32(Console.ReadLine());

for(int i = width; i > 0; i--)

{

for(int j = i; j > 0; j--)

{

Console.Write(num + " ");

}

Console.Write("\n");

}

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

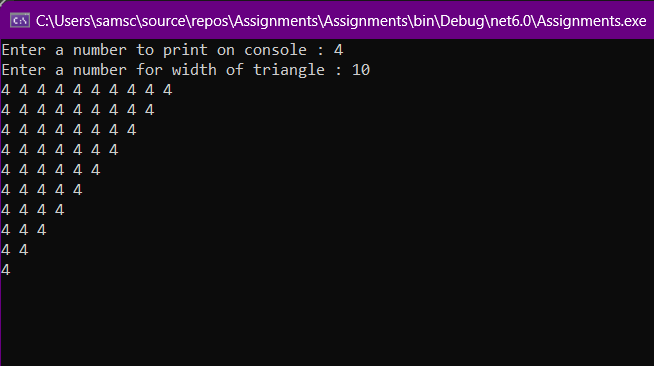
}

Console.ReadKey();

}

}

}



1. Write a C# Sharp program that takes userid and password as input (type string). After 3 wrong attempts, user will be rejected.(Store user data in hashtable/Dictionary).

namespace Assignments

{

internal class Program

{

static void Main(string[] args)

{

int counter = 3;

var cities = new Hashtable(){

{"Username", "abhijeet002"},

{"Password", "abhi@1223"}

};

do

{

Console.Write("Enter Username : ");

string username = Console.ReadLine();

Console.Write("Enter Password : ");

string password = Console.ReadLine();

if (username != (string)cities["Username"] || password != (string)cities["Password"])

{

Console.WriteLine("wrong credentials");

counter--;

Console.WriteLine($"You have only {counter} attempts available..");

}

else

{

Console.WriteLine("Welcome to C#");

break;

}

} while (counter > 0);

if (counter == 0)

{

Console.WriteLine("Your account has been locked contact admin");

}

Console.ReadKey();

}

}

}

Text

Description automatically generated

1. Write a C# Sharp program that takes two numbers as input and perform an operation ( + , - , \*, x , / ) on them and displays the result of that operation.

namespace Assignments

{

class Program

{

static double getCalculation(double a, double b, int choice)

{

var result = choice switch

{

1 => a + b,

2 => a - b,

3 => a / b,

4 => a \* b,

\_ => 0,

};

return result;

}

static void Main(string[] args)

{

char input = 'n';

do

{

try

{

Console.WriteLine("\n\n===============================");

Console.WriteLine("\t\tMenu");

Console.WriteLine("Press 1 for Addition: ");

Console.WriteLine("Press 2 for Subtraction: ");

Console.WriteLine("Press 3 for Division: ");

Console.WriteLine("Press 4 for Multiplication: ");

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

int choice = Convert.ToInt32(Console.ReadLine());

if (choice < 1 || choice > 4)

{

Console.WriteLine("Wrong Input!!!! Retry again...");

continue;

}

else

{

Console.Write("Enter 1st number for operation: ");

double a = Convert.ToDouble(Console.ReadLine());

Console.Write("Enter 2nd number for operation: ");

double b = Convert.ToDouble(Console.ReadLine());

double result = getCalculation(a, b, choice);

Console.WriteLine("\n\nCalculated result : " + result);

}

Console.WriteLine("Do you wish to continue? (Y/N)");

input = Convert.ToChar(Console.ReadLine());

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

Console.WriteLine("Do you wish to continue? (Y/N)");

input = Convert.ToChar(Console.ReadLine());

continue;

}

}

while (input == 'y' || input == 'Y');

}

}

}

Text

Description automatically generated

1. Write a C# Sharp program that takes the radius of a circle as input and calculate the perimeter and area of the circle.

namespace Assignments

{

class Program

{

static void Circle(double r)

{

double pi = 3.14;

Console.WriteLine($"\nArea of the circle {pi\*r\*r} unit sq.");

Console.WriteLine($"Perimeter of the circle {2\*pi\*r} unit sq.");

}

static void Main(string[] args)

{

try

{

Console.Write("Enter the radius of the circle : ");

double radius = Convert.ToDouble(Console.ReadLine());

Circle(radius);

}

catch(Exception ex)

{

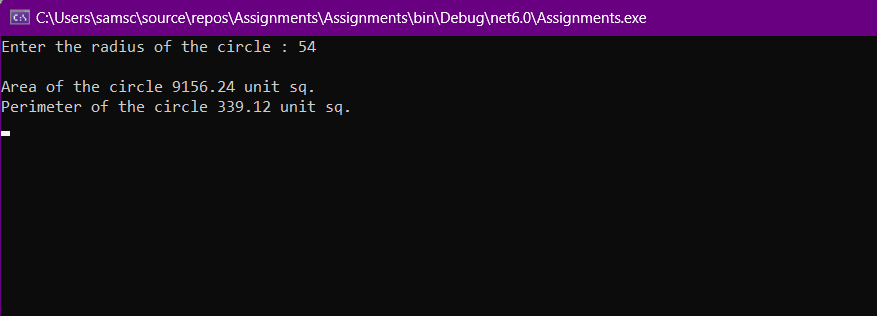
Console.WriteLine(ex.Message);

}

}

}

}



1. Write a C# Sharp program that takes distance and time as input and displays the speed in kilometres per hour and miles per hour.

namespace Assignments

{

class Program

{

static void GetSpeed(double d, double t)

{

double speed = d / t;

Console.WriteLine($"\nSpeed is - {speed}kmph");

Console.WriteLine($"Speed is {speed/1.6}mph");

Console.WriteLine($"Speed is {speed/3.6}mps");

}

static void Main(string[] args)

{

try

{

Console.Write("Enter the distance (in km) : ");

double distance = Convert.ToDouble(Console.ReadLine());

Console.Write("Enter the time (in hours) : ");

double time = Convert.ToDouble(Console.ReadLine());

GetSpeed(distance, time);

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

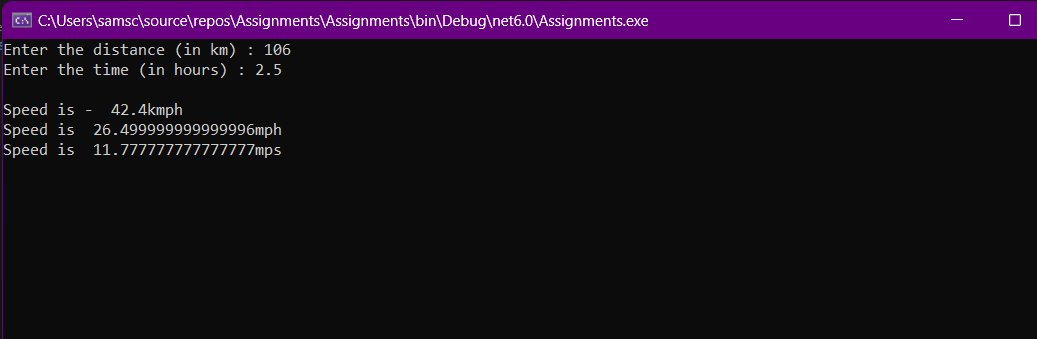
}

Console.ReadKey();

}

}

}



1. Write a C# Sharp program that takes a character as input and check the input (lowercase) is a vowel, a digit, or any other symbol.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

try

{

Console.Write("Enter a character : ");

char input = Convert.ToChar(Console.ReadLine().ToLower());

if ((int)input >= 48 && (int)input <= 57)

Console.WriteLine("Entered character is a digit");

else if((int)input >= 97 && (int)input <= 122)

{

if (input == 'a' || input == 'e' || input == 'i' || input == 'o' || input == 'u')

Console.WriteLine("Entered character is a vowel");

else

Console.WriteLine("Entered character is a alphabet");

}

else

Console.WriteLine("Entered character is a symbol");

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

}

Console.ReadKey();

}

}

}

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

1. Write a C# Sharp program that takes two numbers as input and returns true or false when both numbers are even or odd.

namespace Assignments

{

class Program

{

static bool getOddEven(int a, int b)

{

if(a%2 == 0 && b%2 == 0)

return true;

else

return false;

}

static void Main(string[] args)

{

try

{

Console.Write("Enter a number : ");

int num1 = Convert.ToInt32(Console.ReadLine);

Console.Write("Enter another number : ");

int num2 = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Result is : " + getOddEven(num1, num2));

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

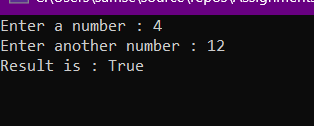
}

Console.ReadKey();

}

}

}



Text

Description automatically generated

1. Write a C# Sharp program that takes a decimal number as input and displays its equivalent in binary form.

namespace Assignments

{

class Program

{

static string getBinary(int num)

{

int rem;

string binary = "";

while(num > 0)

{

rem = num % 2;

binary += rem;

num /= 2;

}

return binary;

}

static void Main(string[] args)

{

try

{

Console.Write("Enter a deciaml number : ");

int num1 = Convert.ToInt32(Console.ReadLine());

string binaryNum = getBinary(num1);

for(int i = binaryNum.Length - 1; i >= 0; i--)

Console.Write(binaryNum[i]);

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

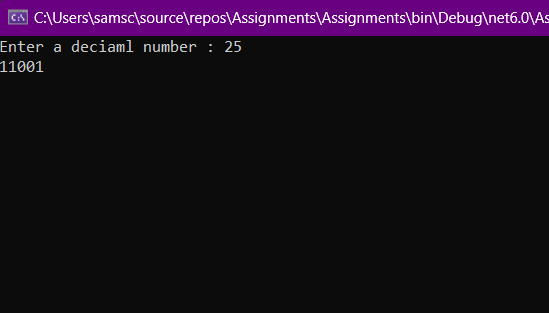
}

Console.ReadKey();

}

}

}



1. Write a C# Sharp program to get the absolute difference between n and 51. If n is greater than 51 return triple the absolute difference.

namespace Assignments

{

class Program

{

static int getAbsolute(int num)

{

if (num > 51)

return (num \* 3);

else

return (51 - num);

}

static void Main(string[] args)

{

try

{

int[] numArr = { 36, 54, 65, 51, 11, 13 };

Console.WriteLine("Absolute Differene for number and 51 is: ");

foreach (int num in numArr)

Console.WriteLine(getAbsolute(num));

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

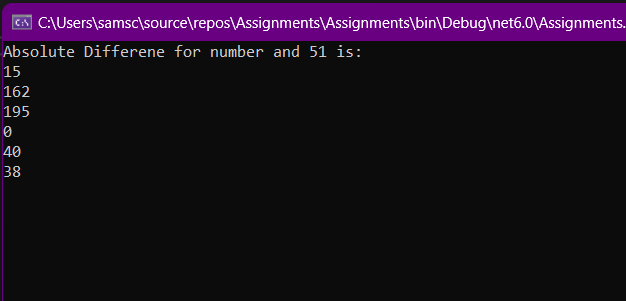
}

Console.ReadKey();

}

}

}



1. Write a C# Sharp program to remove the character in a given position of a given string. The given position will be in the range 0.. string length -1 inclusive.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

try

{

string tempStr = "Australia";

string slicedStr = "";

Console.WriteLine(tempStr);

Console.Write("\nEnter index to remove element : ");

int index = Convert.ToInt32(Console.ReadLine());

if (index < 0 || index > tempStr.Length - 1)

Console.WriteLine("Out of Range...");

else

{

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

{

if(i != index)

{

slicedStr += tempStr[i];

}

}

Console.WriteLine(slicedStr);

}

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

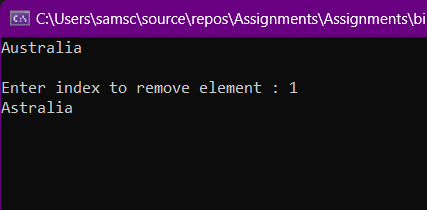
}

Console.ReadKey();

}

}

}



Text

Description automatically generated

1. Write a C# Sharp program to exchange the first and last characters in each string and return the new string.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

try

{

string Str = "Abhishek";

char[] CharArray = Str.ToCharArray();

char temp;

if (Str.Length > 1)

{

int Length = CharArray.Length - 1;

temp = CharArray[0];

CharArray[0] = CharArray[Length];

CharArray[Length] = temp;

Console.WriteLine(CharArray);

}

else

{

Console.WriteLine(CharArray);

}

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

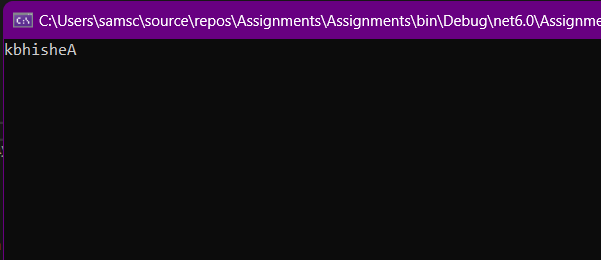
}

Console.ReadKey();

}

}

}



1. Write a C# Sharp program to create a new array from two given array of integers, each length 3.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

try

{

int[] IntArr1 = { 110, 232, 130 };

int[] IntArr2 = { 224, 536, 618, 124, 658 };

int[] ResultArr = IntArr1.Concat(IntArr2).ToArray();

Console.Write("Array 1 : ");

foreach(int i in IntArr1)

Console.Write(i +", ");

Console.Write("\n\nArray 2 : ");

foreach(int i in IntArr2)

Console.Write(i +", ");

Console.WriteLine("\n\n\nResult Array : ");

foreach(int i in ResultArr)

Console.Write(i +", ");

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

}

Console.ReadKey();

}

}

}

Text

Description automatically generated

1. Write a C# Sharp program to count the number of strings with given length in given array of strings

namespace Assignments

{

class Program

{

static int getStrings(string[] arr, int n)

{

int Count = 0;

foreach (string str in arr)

if (str.Length == n)

Count++;

return Count;

}

static void Main(string[] args)

{

try

{

string[] StringArr = { "a", "b", "c", "aa", "bbb" };

int StringLen = 1;

Console.WriteLine($"Number of strings with length {StringLen} is : {getStrings(StringArr, StringLen)}");

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

}

Console.ReadKey();

}

}

}

Text

Description automatically generated

1. Write a C# Sharp program to calculate the value that results from raising 3 to a power ranging from 0 to 32.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

for(int i = 0; i < 33; i++)

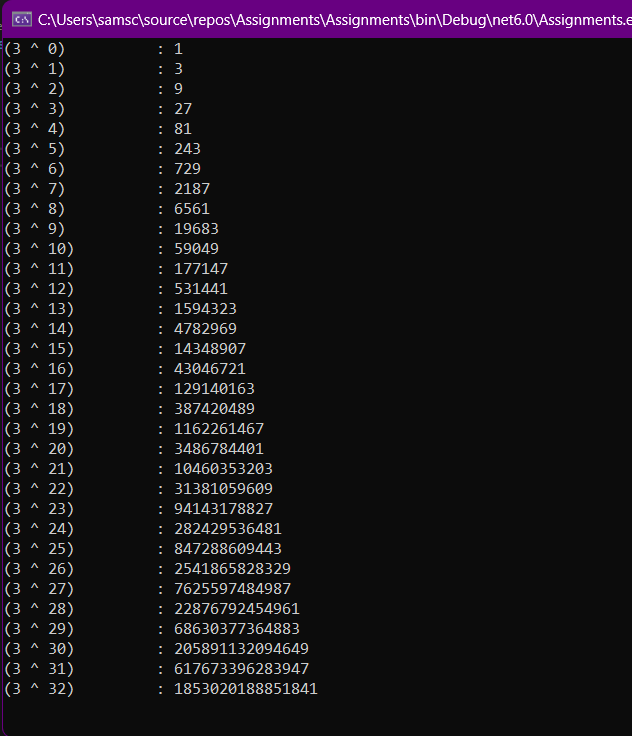
Console.WriteLine($"(3 ^ {i}) \t : {Math.Pow(3, i)}");

Console.ReadKey();

}

}

}



1. Write a C# Sharp program to convert a given integer value to Roman numerals.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

int[] values = { 1000, 900, 500, 400, 100, 90, 50, 40, 10, 9, 5, 4, 1 };

String[] romanLetters = { "M", "CM", "D", "CD", "C", "XC", "L", "XL", "X", "IX", "V", "IV", "I" };

Console.Write("Enter number to coonvert it into roman number : ");

int num = Convert.ToInt32(Console.ReadLine());

Console.Write(num);

string roman = "";

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

{

while (num >= values[i])

{

num -= values[i];

roman += romanLetters[i].ToString();

}

}

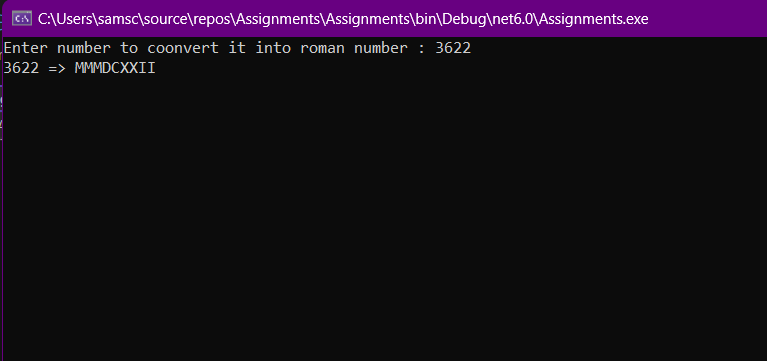
Console.Write(" => " + roman);

Console.ReadKey();

}

}

}



1. Write a program in C# Sharp to find the sum of first n natural numbers using recursion.

namespace Assignments

{

class Program

{

public static int Sum(int n)

{

if (n > 0)

return n + Sum(n - 1);

return n;

}

static void Main(string[] args)

{

Console.WriteLine("Enter the limit for sum : ");

int num = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Sum is : " + Sum(num));

Console.ReadKey();

}

}

}

Text

Description automatically generated

1. Write a program in C# Sharp to display the individual digits of a given number using recursion.

namespace Assignments

{

class Program

{

public static void Digit(int n)

{

if (n < 10)

{

Console.Write(n + " ");

return;

}

Digit(n / 10);

Console.Write(n % 10 + " ");

}

static void Main(string[] args)

{

Console.WriteLine("Enter the limit for sum : ");

int num = Convert.ToInt32(Console.ReadLine());

Digit(num);

Console.ReadKey();

}

}

}

Text

Description automatically generated

1. Write a program in C# Sharp to find the number of an array and the square of each number using LINQ.

namespace Assignments

{

public class NumSquare

{

public int Num { get; set; }

public int Square { get; set; }

}

class Program

{

static void Main(string[] args)

{

List<NumSquare> numList = new()

{

new NumSquare {Num = 1, Square = 1},

new NumSquare {Num = 3, Square = 9},

new NumSquare {Num = 9, Square = 81},

new NumSquare {Num = 5, Square = 25},

new NumSquare {Num = 7, Square = 49},

new NumSquare {Num = 2, Square = 4},

new NumSquare {Num = 4, Square = 16},

new NumSquare {Num = 8, Square = 64},

new NumSquare {Num = 10, Square = 100},

};

Console.Write("Enter the minimum range of squares (max - 100): ");

int input = Convert.ToInt32(Console.ReadLine());

IEnumerable<NumSquare> sq = from num in numList where num.Square > input select num;

foreach (NumSquare a in sq)

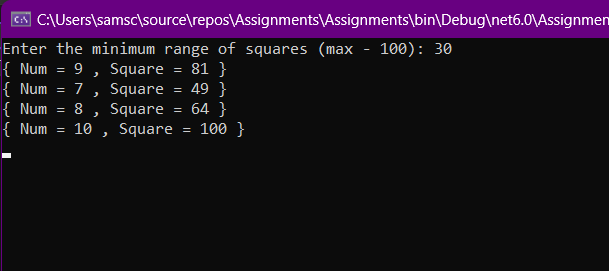
Console.WriteLine($"{{ Num = {a.Num} , Square = {a.Square} }}");

Console.ReadKey();

}

}

}



1. Write a program in C# Sharp to display the characters and frequency of character from giving string using LINQ.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

Console.Write("Enter the string to get frequency : ");

string? input = Console.ReadLine();

var Frequency = from letter in input

group letter by letter into freq

select freq;

Console.Write("\n\nFrequency of the characters in the given string is :\n");

foreach (var character in Frequency)

Console.WriteLine($"{character.Key} - {character.Count()}");

Console.ReadKey();

}

}

}

Text

Description automatically generated

1. Write a program in C# Sharp to find the string which starts and ends with a specific character using LINQ.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

List<string> CityList = new()

{

"Jabalpur", "Chandigarh", "Jalandhar", "Gwalior", "Meerut", "Rishikesh", "Delhi", Nagpur", "Ajmer", "Rajkot", "Vijayawada", "Jagdalpur", "Bengaluru", "Pune", "Mumbai”

};

Console.WriteLine("Enter starting character : ");

char start = Convert.ToChar(Console.ReadLine().ToUpper());

Console.WriteLine("Enter ending character : ");

char end = Convert.ToChar(Console.ReadLine().ToLower());

var cities = from cityName in CityList where cityName.StartsWith(start) where cityName.EndsWith(end) select cityName;

Console.WriteLine("\nFound Cities are - ");

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

foreach (var city in cities)

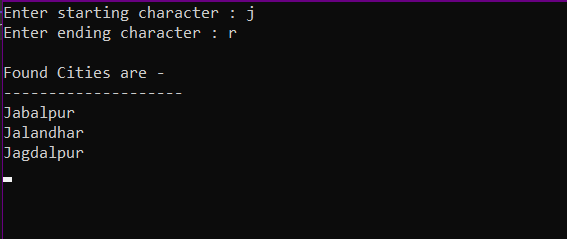
Console.WriteLine(city);

Console.ReadKey();

}

}

}



1. Write a program in C# Sharp to display the top n-th records using LINQ.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

List<int> StudentMarks = new()

{

75, 64, 94, 52, 68, 71, 55, 12, 36, 79, 92, 83, 91, 70, 43, 21

};

try

{

Console.Write("Enter number of top records you want : ");

int record = Convert.ToInt32(Console.ReadLine());

StudentMarks.Sort();

StudentMarks.Reverse();

Console.WriteLine("\nTop records for marks are - ");

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

foreach (int i in StudentMarks.Take(record))

Console.WriteLine(i);

Console.ReadKey();

}

catch (Exception ex)

{

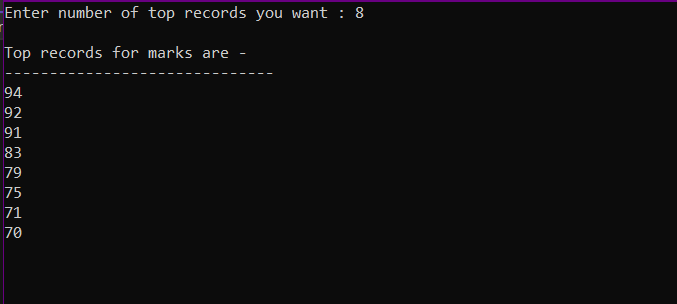
Console.WriteLine(ex.Message);

}

}

}

}



1. Write a program in C# Sharp to count file extensions and group it using LINQ.

namespace Assignments

{

class Program

{

static void Main(string[] args)

{

List<string> FileList = new()

{

"php.txt", "contact.html", "c#.cs", "calculator.cs", "c++.cpp", "index.html", "home.html", "numbers.cs", "laptop.cpp", "hello.py"

};

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

FileList[i] = FileList[i].Split(".")[1];

var ExtGroup = from file in FileList group file by file into extensions select extensions;

Console.WriteLine("Extension count for the files are ");

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

foreach (var ext in ExtGroup)

Console.WriteLine($"{ext.Key} \t- {ext.Count()}");

Console.ReadKey();

}

}

}

Text

Description automatically generated