

Figure 1 All 6 into same output

Program.cs

//Create a application that

//take 2 numbers and find its

//1) sum,=

//2) product

//3) and divide the first by second

//4), also supract the second from the first.

//5) Include another method to find the remainder when the first number is divided by second

using Arithmetic;

class BasicArithMetic(int a, int b)

{

public int a = a;

public int b = b;

public void PerformTask()

{

int sum = this.a + this.b;

Console.WriteLine("Sum " + sum);

int product = this.a \* this.b;

Console.WriteLine("Product " + product);

int division = this.a / this.b;

Console.WriteLine("Division " + division);

this.a -= this.b;

Console.WriteLine("a after b is reduced from a");

Console.WriteLine($"Remainder {this.GetRemainder()}");

}

public int GetRemainder()

{

return this.a % this.b;

}

}

class Program

{

public int a;

public int b;

public static void Main(string[] args)

{

Console.WriteLine("Enter a and b");

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

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

BasicArithMetic arithmetic = new(a, b);

arithmetic.PerformTask();

Console.WriteLine("============= 2 ===========");

\_ = new MaxUntilNegate();

Console.WriteLine("============= 3 ===========");

\_ = new AverageUntillXBy7();

Console.WriteLine("============= 4 ===========");

\_ = new UserNameLength();

Console.WriteLine("============ 5 ===========");

\_ = new UserNamePasswordChecker();

Console.WriteLine("============ 6 ===========");

\_ = new LeastVowels();

}

}

//int sum = a + b;

//int prodcut = a \* b;

//int division = a / b;

//b -= a;

UsernameLength.cs

using System;

namespace Arithmetic

{

internal class UserNameLength

{

public string Name;

public UserNameLength()

{

Console.WriteLine("Enter your username");

this.Name = Console.ReadLine();

Console.WriteLine($"Length of the username is {this.Name.Length}");

}

}

}

UsernamePasswordCheck.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Runtime.InteropServices;

using System.Runtime.Intrinsics.X86;

using System.Text;

using System.Threading.Tasks;

using static System.Net.Mime.MediaTypeNames;

using static System.Runtime.InteropServices.JavaScript.JSType;

namespace Arithmetic

{

// Create a application

// that will take username and password from user.

// check if username is "ABC" and passwod is "123".

//

// Print error message if username or password is wrong

// Allow user 3 attemts.

// After 3rd attempt if user enters invalid credentials then print msg to intimate user

// taht he/she has exceeded the number of attempts and stop

internal class UserNamePasswordChecker

{

public string? username;

public string? password;

public UserNamePasswordChecker()

{

this.CheckPassword();

}

private void CheckPassword( int times = 3) {

int attempts = 0;

while (true)

{

Console.WriteLine("Enter your username");

this.username = Console.ReadLine();

Console.WriteLine("Enter your password");

this.password = Console.ReadLine();

if(this.username != "abc") {

Console.WriteLine($"No username found as {this.username}");

return;

}

while(attempts < times - 1) {

if(this.password == "123") {

Console.WriteLine("You're logged in!");

return;

}

Console.WriteLine("Wrong Password, please enter again");

this.password = Console.ReadLine();

attempts++;

}

return;

}

}

}

}

MaxUntilNegate

using System;

namespace Arithmetic

{

public class MaxUntilNegate

{

public int a;

public int b;

public MaxUntilNegate() {

// Create an application

// that will find the greatest of the given numbers.

// Take input until the user enters a negative number

Console.WriteLine("This program works until you give a negative number");

while (true) {

Console.WriteLine("Enter first value: ");

this.a = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Enter second value: ");

this.b = Convert.ToInt32(Console.ReadLine());

if(!(this.a >= 0) || !(this.b >= 0))

{

Console.WriteLine("Ending");

break;

}

Console.WriteLine($"Max - {Math.Max(this.a, this.b)}");

}

}

}

}

LeastVowels

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Arithmetic

{

internal class LeastVowels

{

//Take a string from user the words seperated by comma(",").

//seperate the words and

//

//find the words

// with the least number of

// repeating vowels.

//

//print the count and the word.

//If there is a tie then print all the words that tie for the least

public string InputString { get; set; }

public LeastVowels()

{

this.FindWordsWithLeastVowels();

}

private void FindWordsWithLeastVowels()

{

Console.WriteLine("Enter the string");

InputString = Console.ReadLine();

string[] words = InputString.Split(',');

int leastVowelCount = int.MaxValue;

foreach (string word in words)

{

leastVowelCount = Math.Min(leastVowelCount, CountRepeatedVowels(word));

}

foreach (string word in words) {

if(CountRepeatedVowels(word) == leastVowelCount)

{

Console.WriteLine(word);

}

}

}

private static int CountRepeatedVowels(string word)

{

int maxCount = int.MinValue;

char[] vowels = ['a', 'e', 'i', 'o', 'u'];

word = word.ToLower();

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

if (vowels.Contains(word[i])) {

int count = 0;

for (int j = i + 1; j < word.Length; j++) {

if (vowels.Contains(word[j])) count++;

}

maxCount = Math.Max(maxCount, count);

}

}

return maxCount;

}

}

}

AverageUntilXby7

using System;

namespace Arithmetic

{

internal class AverageUntillXBy7

{

//Find the avearage of all the numbers

//that are divisible by 7 until the user specified number.

//

//Take input until the user enters a negative number

public int num;

public AverageUntillXBy7() {

while (true)

{

Console.WriteLine("Enter your number");

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

if (this.num < 0)

{

Console.WriteLine("Ending");

break;

}

int sum = 0;

for (int i = 7; i < this.num; i += 7) {

sum += i;

}

Console.WriteLine($"Sum = {sum}");

}

}

}

}