|  |  |
| --- | --- |
| 1.  Write a program in console application to input two number and display their sum. |  |
| 2.  Write a program to input a number and display it is prime number or not. |  |
| 3.  Write a program to input a number and display it is palindrome or not. |  |
| 4.  Write a program to input 10 numbers and display in ascending order by using Array function. And also display Min, Max value and sum of Array. |  |
| 5.  Write a program to demonstrate constructor, method overriding and method overloading. |  |
| 6.  Write a program to demonstrate Interface and Also show program of multiple inheritance by using interface. |  |
| 7.  Write a program to create Structure type which store Principal,rate , time and display simple interest. Initialized the variable to default value with constructor. |  |
| 8.  WAP to demonstrate multiple inheritance in .Net Framework |  |
| 9.  WAP to add, subtract of two number using delegate concept. |  |
| 10. WAP to find sum of digits and its reverse using delegate method. |  |
| 11. Create a seal class to demonstrate. |  |
| 12. Crate a stack data structure and demonstrate different type of stack operation. |  |
| 13. Create a Queue data structure and demonstrate different type of Queue operation. |  |
| 14. Write a program to collect id and name of student in dictionary and find information about id 1 student and display it. |  |
| 15. Write a program to demonstrate use of Hashset collection concept. |  |
| 16. Write a program to input a paragraph from a user and save in a file. |  |
| 17.Write a program to read content of file and display it. Before it check file exist or not. |  |

1.  Write a program in console application to input two number and display their sum.

using System;

namespace sumoftwo

{class Program

{

static void Main(string[] args)

{

int num1, num2, result;

Console.WriteLine("enter the number1");

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

Console.WriteLine("enter the number2");

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

console.writeline(“The sum is”);

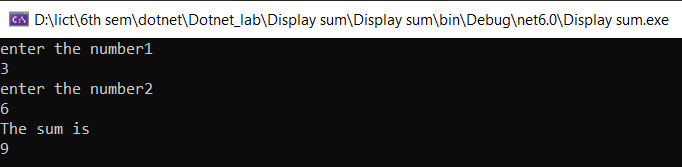
Console.WriteLine(num1 + num2);

Console.ReadLine();

}

}

}



2.  Write a program to input a number and display it is prime number or not.

using System;

public class PrimeNumberExample

{

public static void Main(string[] args)

{

int n, i, m = 0, flag = 0;

Console.Write("Enter the Number to check Prime: ");

n = int.Parse(Console.ReadLine());

m = n / 2;

for (i = 2; i <= m; i++)

{ if (n % i == 0)

{

Console.Write("Number is not Prime.");

flag = 1;

break;

}

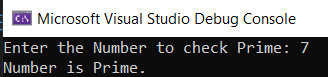
}

if (flag == 0)

Console.Write("Number is Prime.");

}

}



3.  Write a program to input a number and display it is palindrome or not.

using System;

using static System.Console;

namespace palindrome\_check

{

class Program

{

static void Main(string[] args)

{

WriteLine("Checking The number is Palindrome or Not");

Write("Enter a Number ");

int number = Convert.ToInt32(ReadLine());

int r, sum = 0;

int temp = number;

while (number > 0)

{

r = number % 10;

sum = (sum \* 10) + r;

number = number / 10;

}

if (temp == sum)

{

WriteLine("Palindrome");

}

else

{

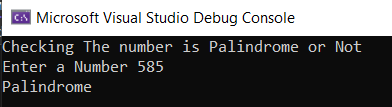
WriteLine("Not Palindrome");

}

}

}

}



4.  Write a program to input 10 numbers and display in ascending order by using Array function. And also display Min, Max value and sum of Array.

using System;

using static System.Console;

namespace array\_numbers

{ class Program

{static void Main(string[] args)

{

var number = new int[10];

int i;

int sum = 0;

WriteLine("Enter 10 numbers");

for (i = 0; i < 10; i++)

{ number[i] = Convert.ToInt32(ReadLine());

}

Array.Sort(number);

WriteLine("Arranging in ascending order");

foreach (int nums in number)

{

Write(nums + ", ");

}

int max = number[0];

int min = number[0];

foreach (int nums in number)

{

sum = sum + nums;

if (nums > max)

{max = nums;

}

else

{ min = nums;

}

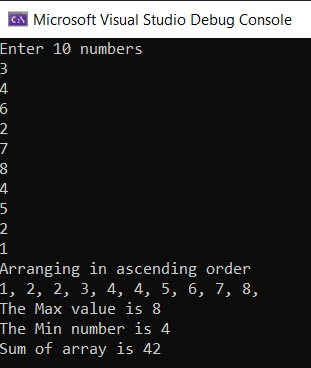
}

WriteLine("\nThe Max value is " + max);

WriteLine("The Min number is " + min);

WriteLine("Sum of array is " + sum);

}

}}  
  


5.  Write a program to demonstrate constructor, method  
overriding and method overloading.

using System;

namespace method\_Overriding

{

class Animal

{

public virtual void Eat()

{

Console.WriteLine("Animal Eats Foods");

}

}

class Dog : Animal

{

public override void Eat()

{

Console.WriteLine("Dog Eats Foods");

}

public static void Main(string[] args)

{

Animal myanimal = new Animal();

myanimal.Eat();

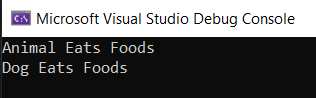
Animal mydog = new Dog();

mydog.Eat();

}

}

}



6.Write a program to demonstrate Interface and Also show program of multiple inheritance by using interface.

using System;

using static System.Console;

using System.Collections;

namespace Interface\_multi\_Inheritance

{

interface Iinterface1

{

void language();

}

class Program1 : Iinterface1

{

public void language()

{

ArrayList mylist = new ArrayList();

mylist.Add("C");

mylist.Add("C#");

mylist.Add("Python");

mylist.Add("Java");

WriteLine("\n\*\*\*\*\*\*\*Language You need to Know :\*\*\*\*\*\*\*\*");

foreach (var elements in mylist)

{

WriteLine(elements);

}

}

}

interface Iinterface2

{

void courses();

}

class Program2 : Iinterface2

{

public void courses()

{

ArrayList mylist = new ArrayList();

mylist.Add("C fundamentals");

mylist.Add("C# classes");

mylist.Add("Python for data science");

mylist.Add("Java for applications");

WriteLine("\n\*\*\*\*\*\*\*Courses provided by Programminglanguages\*\*\*\*\*\*");

foreach (var elements in mylist)

{

WriteLine(elements);

}

}

}

class Program : Iinterface1, Iinterface2

{

Program1 obj1 = new Program1();

Program2 obj2 = new Program2();

public void language()

{

obj1.language();

}

public void courses()

{

obj2.courses();

}

}

class main\_Class

{

static void Main(string[] args)

{

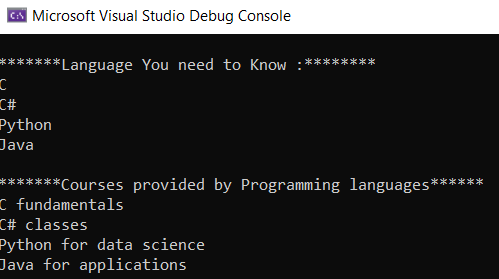
Program obj = new Program();

obj.language();

obj.courses();

}

}

}  
 

7.  Write a program to create Structure type which store Principal, rate , time and display simple interest. Initialized the variable to default  
value with constructor  
  
 using System;

using static System.Console;

namespace structure\_with\_constructor

{

struct SimpleInterest

{

public int principal;

public double rate;

public int time;

public void getSI(int p, int t, double r)

{

double si = (p \* t \* r) / 100;

WriteLine("Simple Interest is " + si);

}

}

class Program

{

static void Main(string[] args)

{

SimpleInterest obj = new SimpleInterest();

obj.principal = 100;

obj.time = 3;

obj.rate = 5.5;

obj.getSI(obj.principal, obj.time, obj.rate);

}

}

}



8.  WAP to demonstrate Interface in .Net Framework using System;

namespace InterfaceEg

{

interface Iinterface1

{

void method1();

void method2();

}

class Myclass1

{

public void M1()

{

Console.WriteLine("M1 from Class 1");

}

}

class Myclass2 : Myclass1, Iinterface1

{

public void method1()

{

Console.WriteLine("Method 1");

}

public void method2()

{

Console.WriteLine("Method 2");

}

}

class Program

{

static void Main(string[] args)

{

Myclass2 obj = new Myclass2();

obj.method1();

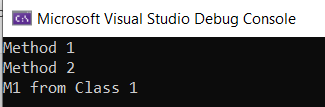
obj.method2();

obj.M1();

}

}

}



9.  WAP to add, subtract of two number using delegate  
concept.

using System;

namespace Programs

{

class class1

{

public delegate void addnum(int a, int b);

public delegate void subnum(int a, int b);

public void sum(int a, int b)

{

Console.WriteLine("(100 + 40) = {0}", a + b);

}

public void subtract(int a, int b)

{

Console.WriteLine("(100 - 60) = {0}", a - b);

}

public static void Main(String[] args)

{

class1 obj = new class1();

addnum del\_obj1 = new addnum(obj.sum);

subnum del\_obj2 = new subnum(obj.subtract);

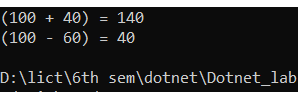
del\_obj1(100, 40);

del\_obj2(100, 60);

}

}

}

  
  
  
10. WAP to find sum of digits and its reverse using delegate  
method.

using System;

using static System.Console;

namespace digits\_delegates

{

public delegate void sumDelegates(int n);

class Program

{

public void reverseDigits(int digits\_number)

{

int rev = 0;

int sum = 0;

while (digits\_number > 0)

{

int i = digits\_number % 10;

rev = rev \* 10 + i;

sum = i + sum;

digits\_number = digits\_number / 10;

}

WriteLine("Reverse Digit is " + rev);

WriteLine("Sum is " + sum);

}

static void Main(string[] args)

{

Program obj = new Program();

sumDelegates digits = obj.reverseDigits;

WriteLine("Enter a number");

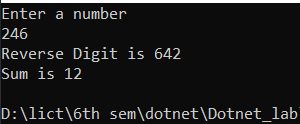
int num = Convert.ToInt32(ReadLine());

digits.Invoke(num);

}

}

}



11. Create a seal class to demonstrate.

using System;

using static System.Console;

namespace sealed\_demo

{

sealed class sealedDemo

{

public void add(int x, int y)

{

WriteLine("Sum is " + (x + y));

}

}

class Program

{

static void Main(string[] args)

{

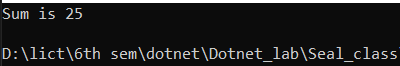
sealedDemo obj = new sealedDemo();

obj.add(10, 15);

}

}

}



12. Crate a stack data structure and demonstrate different  
type of stack operation.

using System;

using System.Collections;

namespace s\_push

{class program

{

static void Main(string[] args)

{

Stack st=new Stack();

st.Push(1);

st.Push(2);

st.Push(3);

foreach(object obj in st)

{

Console.WriteLine(obj);

}

Console.WriteLine();

Console.WriteLine("The number of elements in stack" + st.Count);

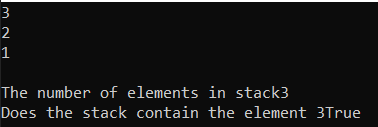
Console.WriteLine("Does the stack contain the element 3" + st.Contains(3));

Console.ReadLine();

}

}

}



13. Create a Queue data structure and demonstrate different  
type of Queue operation.

using System;

namespace Queue\_operation

{

internal class Program

{

static void Main(string[] args)

{

Queue<int> qt = new Queue<int>();

qt.Enqueue(1);

qt.Enqueue(2);

qt.Enqueue(3);

qt.Dequeue();

foreach (object obj in qt)

{

Console.WriteLine(obj);

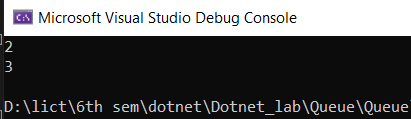
}

Console.ReadKey();

}

}

}

  
  
  
14. Write a program to collect id and name of student in  
dictionary and find information about id 1 student and display it.  
 using System;

using System.Collections.Generic;

using static System.Console;

namespace student\_dictionary

{

class Program

{

static void Main(string[] args)

{

Dictionary<int, string> dict\_obj = new Dictionary<int, string>();

dict\_obj.Add(2, "Ram");

dict\_obj.Add(3, "Hari");

dict\_obj.Add(1, "Rawan");

foreach (KeyValuePair<int, string> item in dict\_obj)

{

WriteLine($"Key:{item.Key} Value:{item.Value}");

}

WriteLine();

WriteLine("Display the student with id 1\n");

if (dict\_obj.ContainsKey(1) == true)

{

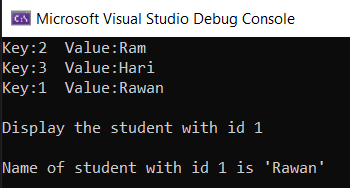
WriteLine($"Name of student with id 1 is '{dict\_obj[1]}'");

}

}

}

}



15. Write a program to demonstrate use of Hashset collection  
concept.

using System;

using System.Collections.Generic;

class GFG

{

static public void Main()

{

HashSet<string> myhash1 = new HashSet<string>();

myhash1.Add("C");

myhash1.Add("C++");

myhash1.Add("C#");

myhash1.Add("Java");

myhash1.Add("Ruby");

Console.WriteLine("Elements of myhash1:");

foreach (var val in myhash1)

{

Console.WriteLine(val);

}

HashSet<int> myhash2 = new HashSet<int>() {10,100,1000,10000,100000};

Console.WriteLine("Elements of myhash2:");

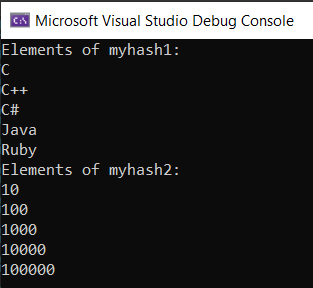
foreach (var value in myhash2)

{

Console.WriteLine(value);

}

}}}



**18 & 19. Write a program to input a paragraph from a user and save  
in a file.**

using System;

using System.IO;

using static System.Console;

using System.Threading;

namespace write\_text\_file

{

class Program

{

static void Main(string[] args)

{

string file = @"D:\lict\6th sem\dotnet\Dotnet\_lab\uttam"; //This is for my linux environment

WriteLine("Enter a sentence or a paragraph");

string message = ReadLine();

File.WriteAllText(file, message);

WriteLine();

WriteLine("Checking to read the file that exists or not");

if (File.Exists(file))

{

Thread.Sleep(2000);

WriteLine("File Exists");

Thread.Sleep(2000);

WriteLine();

WriteLine("The text of the file is");

Thread.Sleep(2000);

WriteLine(File.ReadAllText(file));

}

else

{

WriteLine("There is no such file");

}

}

}

}

