|  |  |
| --- | --- |
| **Ex. No. 01** | **BASIC PROGRAMS IN C#** |
| **31.07.2023** |

**Aim**

To develop C# programs using control statements, arrays and methods.

**Description**

**if else**: If the condition is true the statements inside the if block gets executed else the statements inside the else block gets executed.

Syntax:

if{ //if block statements}

else{ //else block statements}

**switch:** condition matching case statement gets executed.

Syntax:

switch{

case <case\_matching>: //case block statements

default: //default block statements

}

**for loop:** To do a repetitive task; has initialization, condition and increment/decrement.

Syntax:

for(initialization; condition; increment/decrement){

//for loop body statements

}

**Array:** to store multiple values under a single name.

Syntax:

<data\_type>[] <variable>;

**params:** A type of parameters that denotes varying number of parameters.

Syntax:

<return\_type> <ftn\_name>(params <data\_type>[] <variable>)

**out:** A keyword that denotes variable need not to be initialized before passing it to the function

Syntax:

<return\_type> <ftn\_name>(ref <data\_type> <variable>)

**ref:** A keyword that gets the address of the variable that is passed to the function as parameter

Syntax:

<return\_type> <ftn\_name>(out <data\_type> <variable>)

**Source Code**

**1.**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Ex1{

internal class LeapYear{

static void Main(string[] args){

Console.Write("Enter any Year to check is Leap year or not: ");

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

if ((year%400==0)||(year%4==0 && year%100!=0))

{ Console.WriteLine("\n"+year+" is a Leap Year"); }

else { Console.WriteLine("\n"+year+" is not a Leap Year"); }

Console.ReadKey();

}

}

}

**2.**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Ex1{

internal class PrimeNumbers {

public static bool check(int num){

for(int i = 2; i < num/2; i++){

if (num%i == 0) { return false; }

}

if (num!=1) return true;

else return false;

}

static void Main(string[] args) {

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

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

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

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

Console.WriteLine("Prime Numbers Between the Range are as follows");

for (int i = start; i < stop; i++){

if (check(i)){ Console.Write(i+", "); }

}

Console.ReadKey();

}

}

}

**3.**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Ex1{

internal class SimpleCalculator{

static int simple\_cal(int num1, int num2, char ch) {

int res=0;

switch (ch){

case '+':

res=num1 + num2;

break;

case '-':

res=num1 - num2;

break;

case '\*':

res=num1 \* num2;

break;

case '/':

res=num1 / num2;

break;

}

return res;

}

static void Main(string[] args) {

Console.Write("Enter Num1: ");

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

Console.Write("Enter Num2: ");

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

Console.Write(" \n+ => Add \n- => Subtract \n\* => Multiply \n/ => Divide \nEnter Your Choice:");

char ch=(char) Console.Read();

Console.WriteLine("\nResult is: " +simple\_cal(num1,num2,ch));

Console.ReadKey();

}

}

}

**4.**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Ex1{

internal class SimpleCalculator2{

static int simple\_cal2(char ch, params int[] nums){

int res = nums[0];

switch (ch){

case '+':

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

res += nums[i];}

break;

case '-':

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

res -= nums[i];}

break;

case '\*':

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

res \*= nums[i]; }

break;

case '/':

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

res /= nums[i];}

break;

}

return res;

}

static void Main(string[] args) {

Console.Write("Enter #. Inputs: ");

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

int[] nums=new int[n];

Console.WriteLine("Enter #: ");

for (int i = 0; i < n; i++){

nums[i]=Convert.ToInt32(Console.ReadLine());}

Console.Write(" \n+ => Add \n- => Subtract \n\* => Multiply \n/ => Divide \nEnter Your Choice:");

char ch = (char)Console.Read();

Console.WriteLine("\nResult is: " + simple\_cal2(ch, nums));

Console.ReadKey();

}

}

}

**5.**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Ex1{

internal class Banking{

static void create(int amt, out int bal){

bal = amt; }

static void withdraw\_amt(int amt, ref int bal){

bal-=amt; }

static void Main(string[] args) {

int bal;

Console.Write("Enter Amount to create Account: ");

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

create(amt, out bal);

Console.WriteLine("\nAccount Created Successfully");

Console.Write("\nEnter Amount to withdraw from Account: ");

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

if (withdraw <= bal) {

withdraw\_amt(withdraw, ref bal);

Console.WriteLine("\nAmount Withdrawn Successfully");

Console.WriteLine("Balance: " + bal); }

else

Console.WriteLine("\nWithdrawl is not possible"); }

Console.ReadKey();

}

}

}

**6.**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Ex1{

internal class RegSearch{

static void Main(string[] args) {

Console.Write("Enter #. Register #: ");

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

int[] reg\_arr = new int[n];

Console.WriteLine("Enter Register #s: ");

for (int i = 0; i < n; i++){

reg\_arr[i] = Convert.ToInt32(Console.ReadLine());}

Console.Write("\n1. Display \n2. Search \nEnter your Choice: ");

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

if (ch == 1){

Console.WriteLine("Displaying Register #s");

for(int i = 0; i < n-1; i++) {

Console.Write(reg\_arr[i] + ", ");}

Console.Write(reg\_arr[n-1] + ", ");}

else if (ch == 2) {

Console.Write("Enter Register # to search: ");

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

bool flag = false;

for (int i = 0; i < n; i++){

if (reg\_arr[i] == num){

Console.WriteLine(num+" is there in Register #");

flag = true;

break;

}

}

if (!flag) Console.WriteLine(num + " is not there in Register #");

}

Console.ReadKey();

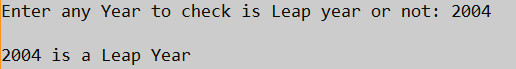
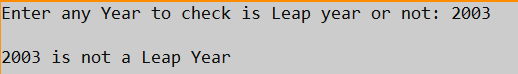
}

}

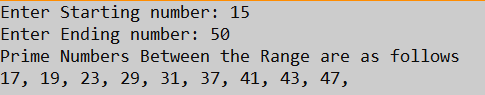
}

**Output**

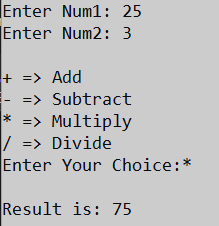
**1.**



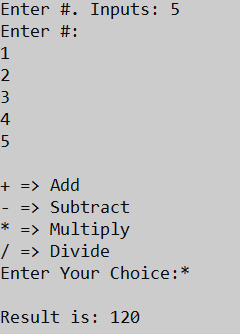
**2.**



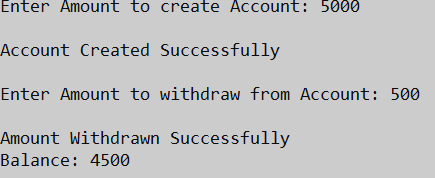
**3.**



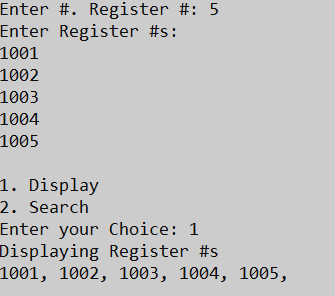
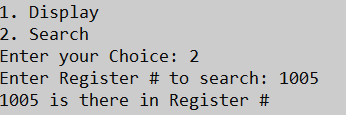
**4.**



**5.**



**6.**

**Result**

The C# programs using control statements, arrays and methods has been executed successfully and the desired output is displayed on the screen.