

Teacher:

# Department of Software Engineering

Name: Muhammad Huzaifa Bawany.

Father Name: Muhammad Hanif Bawany. Section: 2-B.

Student ID: BSE-22S-082

Subject: Object Oriented Programming.

Lab Mannual.

Co-Ordinator: Ayesha Ghayas.

Table of Contents

[**Lab# 01** 3](#_Toc126278414)

[**Lab# 02** 5](#_Toc126278415)

[**Lab# 03** 8](#_Toc126278423)

[**Lab# 04 is cancel by miss.** 15](#_Toc126278431)

[**Lab# 05** 15](#_Toc126278432)

[**Lab# 06** 23](#_Toc126278433)

[**Lab# 07** 28](#_Toc126278434)

[**Lab# 08** 37](#_Toc126278435)

[**Lab# 09** 41](#_Toc126278437)

[**Lab# 10** 48](#_Toc126278438)

[**Lab # 11** 56](#_Toc126278439)

[**Lab# 12** s63](#_Toc126278440)

[**Lab# 13** 69](#_Toc126278441)

# **Lab# 01**

***Tasks***

***Q # 01 (a) Write the output of the program***

using System;

class Addition

{

static void Main( string[] args )

{

string firstNumber, secondNumber;

int number1, number2, sum;

Console.Write( "Please enter the first integer: " );

firstNumber = Console.ReadLine();

Console.Write( "\nPlease enter the second integer: " );

secondNumber = Console.ReadLine();

number1 = Int32.Parse( firstNumber );

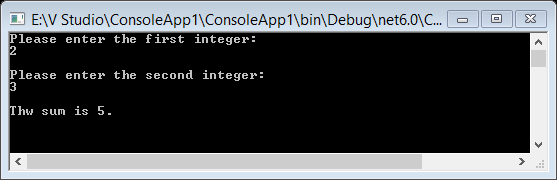
number2 = Int32.Parse( secondNumber );

sum = number1 + number2;

Console.WriteLine( "\nThe sum is {0}.", sum );

}

**Output:**

****

***(b)Output of Block A and Block B are same if it is true then write your findings***

Block B

Console.Write("\nPlease enter the second integer: " );

secondNumber = Console.ReadLine();

number1 = Int32.Parse( firstNumber );

Console.WriteLine( "\nThe sum is {0}.", number1);

Block A

int number1;

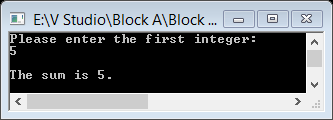
Console.Write( "Please enter the first integer: " );

number1 = Int32.Parse( Console.ReadLine() );

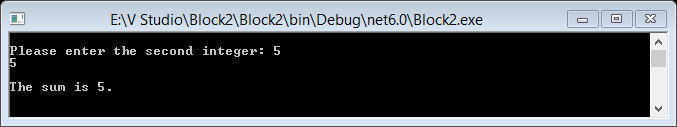
Console.WriteLine( "\nThe sum is {0}.", number1);

**Answer b:**

**Block A Output:**

****

**Block B Output:**

****

Yes Block A and B both are same. we know that in these program method and arrguments both are same and console also print same result output. But logic is different in these program.

***Q # 02 Write an application that asks the user to enter two numbers, obtains the two numbers from the user and prints the sum, product, difference and quotient of the two numbers.***

**Program:**

Source code:

using System;

class Calculator

{

static void Main(string[] args)

{

int num1, num2, sum, difference, product, quotient;

double divide;

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

num1 = Int32.Parse(Console.ReadLine());

Console.WriteLine("Enter Second intiger: ");

num2 = Int32.Parse(Console.ReadLine());

sum = num1 + num2;

difference = num1 - num2;

product = num1 \* num2;

quotient = num1 / num2;

Console.WriteLine("Sum is {0}.", sum);

Console.WriteLine("difference is {0}.", difference);

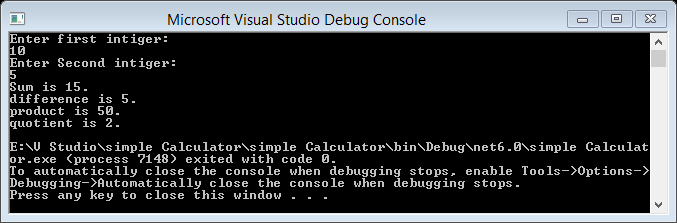
Console.WriteLine("product is {0}.", product);

Console.WriteLine("quotient is {0}.", quotient);

}

}

**Output:**



``

# **Lab# 02**

**Lab Task:**

1. **Write a program that determines that number entered by user is even or odd.**

**Program:**

Source Code :

using System;

namespace Even\_Odd

{

class Program

{

static void Main(string[] args)

{

int num;

Console.WriteLine("Enter the numbers");

num = Int32.Parse(Console.ReadLine());

if (num % 2 == 0)

{

Console.WriteLine("This number is Even Number");

}

else

{

Console.WriteLine("This number is Odd Number");

}

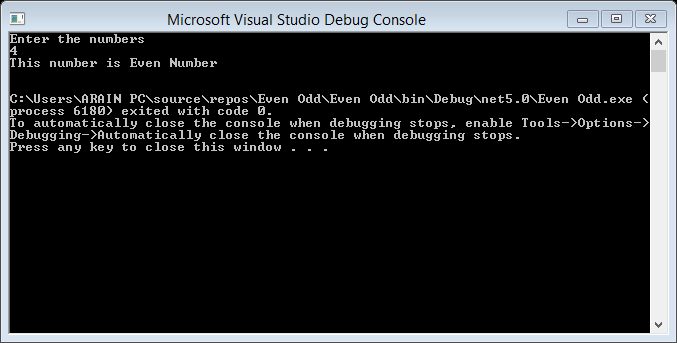
Console.ReadLine();

}

}

}

**Output :**

****

1. **Write a program which takes user’s input for age and on basis of that input gives the following output:**

**If Age is greater than 45 then prints the message “You are old stay at home and wait for call”**

**If Age is less than 30 then prints “Hey! Man enjoy your life with your kids”**

**Use Switch statement in b Task.**

**Program;**

**Source Code:**

using System;

namespace age

{

class Program

{

static void Main(string[] args)

{

int age;

Console.WriteLine("Enter the age :");

age = Int32.Parse(Console.ReadLine());

bool i = age >= 45;

switch (i)

{

case true:

{

Console.WriteLine("You are old stay at home and wait for call");

break;

}

case false:

{

Console.WriteLine("Hey! Man enjoy your life with your kids");

break;

}

}

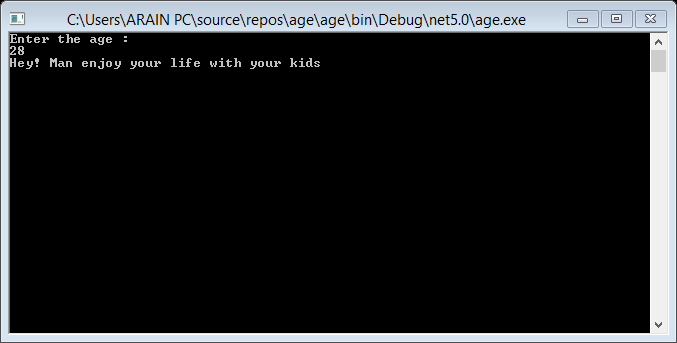
Console.ReadLine();

}

}

}

**Out put:**

****

1. **Write a C# program that takes an integer as amount of money entered to be drawn. If the amount is less than or equal to current balance, then print message “Amount is available.” And if the amount entered by the user is greater than current balance, then print the message “Amount is not available.”**

**Program:**

**Source code:**

using System;

class amount

{

static void Main(string[] args)

{

int balance = 20000;

int withdraw, tax, bal;

Console.WriteLine("Enter the Amount you want to withdraw ");

withdraw = Int32.Parse(Console.ReadLine());

tax = (2 \* withdraw) / 100;

bal = balance - (tax + withdraw);

if (bal >= withdraw)

{

Console.WriteLine("Withdraw sucessfull...");

}

else

{

tax = 0;

bal = balance;

Console.WriteLine("You are out of balance for this transaction ");

}

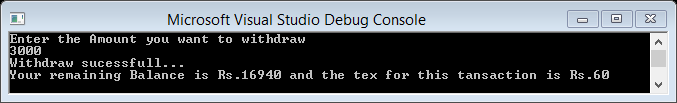
Console.WriteLine("Your remaining Balance is Rs.{0} and the tex for this tansaction is Rs.{1}", bal, tax);

Console.ReadKey();

}

}

**Output:**

****

# **Lab# 03**

**Lab Tasks:**

**Perform All tasks in both C# & Java.**

1. **Write a program to display table of a number, get input by user.**

**Hint: Use for loop.**

**Program:**

**Source code:**

using System;

namespace Table\_of\_a\_number

{

class Program

{

static void Main(string[] args)

{

int i, n;

Console.WriteLine("Enter the Number");

n = Int32.Parse(Console.ReadLine());

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

{

Console.WriteLine("{0} \* {1} = {2} \n",n,i,(n\*i));

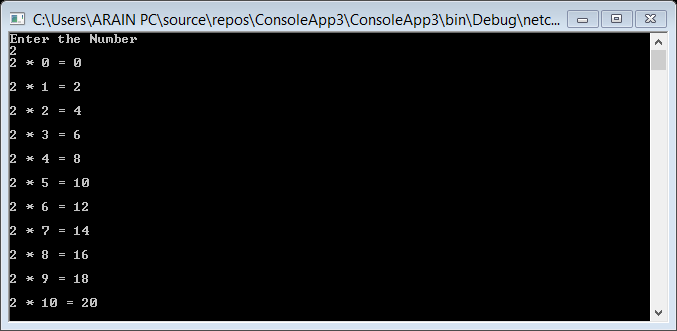
}

Console.ReadLine();

}

}

}

 **Output:**

1. **Write a program to print following output using for loop.  
   1  
   22  
   333  
   4444  
   55555**

***Hint: Use nested for loop***

**Program:**

**Source code:**

using System;

namespace Program

{

class Program

{

static void Main(string[] args)

{

int i,j,n;

Console.Write("Input the Number: ");

n = Int32.Parse(Console.ReadLine());

for(i=1;i<=n;i++)

{

for(j=1;j<=i;j++)

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

Console.WriteLine("\n");

}

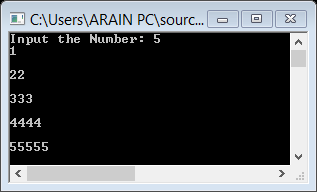
Console.ReadLine();

}

}

}

**Output:**

****

1. **Write a program using do…while loop that prints sum of four numbers, get input from user.**

**Program:**

**Source code:**

using System;

namespace Program\_Practice

{

class Program

{

static void Main(string[] args)

{

int sum = 0;

int i=1, input;

Console.WriteLine("Enter the number: ");

do

{

input = Int32.Parse(Console.ReadLine());

sum = sum + input;

i++;

}

while (i <= 4);

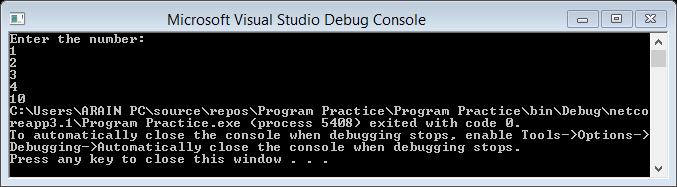
Console.Write("{0}", sum);

}

}

}

**Output:**

****

1. **Write a program using do-while loop that will calculate the sum of every third integer beginning with i=2, (i.e. sum = 2+5+8+11+….)  for values of ‘i’ that are less than 100.**

**Program:**

**Source code:**

using System;

namespace Program

{

class Program

{

static void Main(string[] args)

{

int i,n;

Console.Write("Input the Number: ");

n = Int32.Parse(Console.ReadLine());

for(i=2;i<=100;i=i+3)

{

Console.Write("{0}\t",i);

}

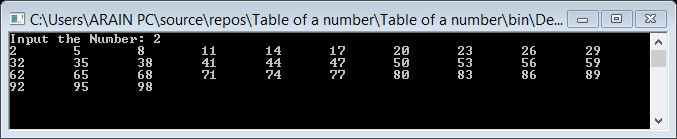
Console.ReadLine();

}

}

}

**Output:**

****

1. **Write a C# program to print first 20 integers in reverse order.**

**Program:**

**Source code:**

using System;

namespace Program

{

class Program

{

static void Main(string[] args)

{

int j , n;

Console.Write("Input the Number: ");

n = Int32.Parse(Console.ReadLine());

for (j=n; j>=1; j--)

{

Console.Write("{0} \t", j);

}

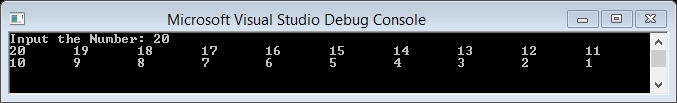
Console.ReadLine();

}

}

}

**Output:**

****

1. **Write a Program to print number in words**

**Program:**

**Source code:**

using System;

namespace Program

{

class Program

{

static void Main(string[] args)

{

int a=0 , n ;

Console.WriteLine("Enter the Number: ");

n = Int32.Parse(Console.ReadLine());

while(n != 0)

{

a = (a \* 10) + (n % 10);

n /= 10;

}

while (a != 0)

{

switch (a % 10)

{

case 0:

Console.WriteLine("Zero ");

break;

case 1:

Console.WriteLine("One ");

break;

case 2:

Console.WriteLine("Two ");

break;

case 3:

Console.WriteLine("Three ");

break;

case 4:

Console.WriteLine("Four ");

break;

case 5:

Console.WriteLine("Five ");

break;

case 6:

Console.WriteLine("Six ");

break;

case 7:

Console.WriteLine("Seven ");

break;

case 8:

Console.WriteLine("Eight ");

break;

case 9:

Console.WriteLine("Nine ");

break;

}

a = a / 10;

}

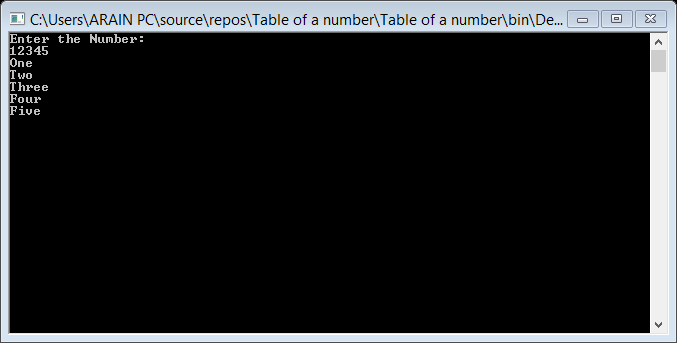
Console.ReadLine();

}

}

}

**Output:**

****

1. **Write Program to get power of number using loop.**

**Program:**

**Source code:**

using System;

namespace Program

{

class Program

{

static void Main(string[] args)

{

int i, n ;

Console.WriteLine("Enter the Number: ");

n = Int32.Parse(Console.ReadLine());

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

{

Console.WriteLine("Power of {0} is {1}",i,(i\*i));

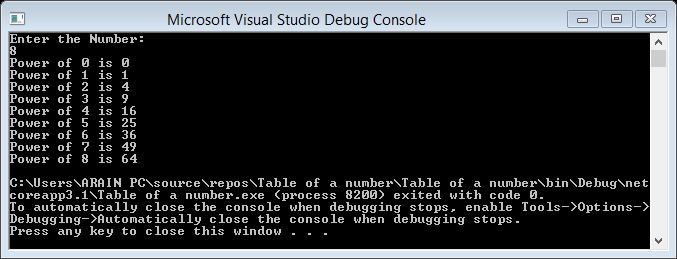
}

}

}

}

**Output:**



1. **Write C# program to prompt user to choose correct answer from given list of answer choices of a question. (use while loop)**

**Program:**

**Source code:**

using System;

namespace Program\_Practice

{

class Program

{

static void Main(string[] args)

{

String option;

string op = "y";

Console.WriteLine("Basic unit of object oriented programing is");

Console.WriteLine("a. Class");

Console.WriteLine("b. Method");

Console.WriteLine("c. Object");

Console.WriteLine("d. Overloading");

while(op=="y")

{

Console.Write("Enter your select option: ");

option = Console.ReadLine();

if (option == "c")

{

Console.WriteLine("Congratulation!");

}

else if (option == "q" || option == "e")

{

Console.WriteLine("Exiting...!");

break;

}

else Console.WriteLine("Incorrect!");

Console.Write("Again? press y to continue:");

option = Console.ReadLine().ToString();

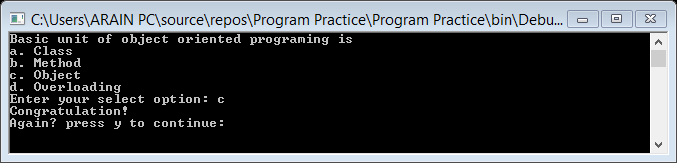
}

}

}

}

**Output:**

****

# **Lab# 04 is cancel by miss.**

# **Lab# 05**

***Task***

1. ***Write the output of the following program.***

using System;

public class Book

{

public string Title;

public string Author;

public short YearPublished;

public int NumberOfPages;

public char CoverType;

}

public class Exercise

{

static void Main()

{

var First = new Book();

First.Title = " Turbo C ";

First.Author = "Robert Lafore";

First.YearPublished = 1996;

First.NumberOfPages = 872;

First.CoverType = 'H';

Console.WriteLine("Book Characteristics");

Console.Write("Title: ");

Console.WriteLine(First.Title);

Console.Write("Author: ");

Console.WriteLine(First.Author);

Console.Write("Year: ");

Console.WriteLine(First.YearPublished);

Console.Write("Pages: ");

Console.WriteLine(First.NumberOfPages);

Console.Write("Cover: ");

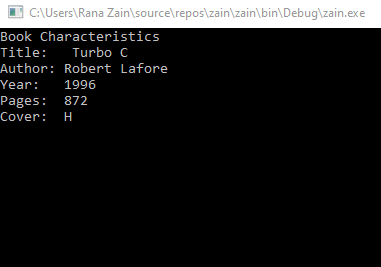
Console.WriteLine(First.CoverType);

Console.ReadKey();

}

}

**Output:**



1. ***Create a class student with a data members name, age, marks of English , marks of math, marks of science, total marks, obtained marks and percentage provide member functions CalculateTotalMarks and CalculatePercentage to calculate marks and percentage in main.***

**Program:**

**Source Code”**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Practice\_Program

{

class students

{

public void data()

{

string name;

int age;

double eng, math, sci, obt\_marks, total\_marks = 300, per;

Console.WriteLine("Enter Student Name:");

name = Console.ReadLine();

Console.WriteLine("Enter Student Age: ");

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

Console.WriteLine("Enter Marks of English: ");

eng = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter Marks of Math: ");

math = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter Student Science: ");

sci = Convert.ToDouble(Console.ReadLine());

obt\_marks = eng + math + sci;

per = (obt\_marks / total\_marks) \* 100;

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

Console.WriteLine("Name of Student is: {0}", name);

Console.WriteLine("Student Age is : {0}", age);

Console.WriteLine("Obtained Marks : {0}", obt\_marks);

Console.WriteLine("Percentage : {0}", per);

}

}

class Program

{

static void Main()

{

students i = new students();

i.data();

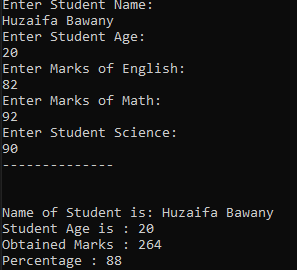
Console.ReadKey();

}

}

}

**Output:**

****

***3. Write a program that contains a class which has a method that takes user name as input and second functions which returns number of vowels present in it and Main program prints the number of vowels***

**Program:**

**Source Code”**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

public class Exercise9

{

public static void Main()

{

string str;

int i, len, vowel, cons;

Console.Write("\n\nCount total number of vowel or consonant :\n");

Console.Write("----------------------------------------------\n");

Console.Write("Input the string : ");

str = Console.ReadLine();

vowel = 0;

cons = 0;

len = str.Length;

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

{

if (str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' || str[i] == 'u' || str[i] == 'A' || str[i] == 'E' || str[i] == 'I' || str[i] == 'O' || str[i] == 'U')

{

vowel++;

}

else if ((str[i] >= 'a' && str[i] <= 'z') || (str[i] >= 'A' && str[i] <= 'Z'))

{

cons++;

}

}

Console.Write("\nThe total number of vowel in the string is : {0}\n", vowel);

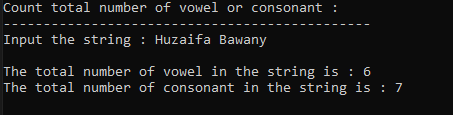
Console.Write("The total number of consonant in the string is : {0}\n\n", cons);

Console.ReadKey();

}

}

**Output:**

****

1. ***Create a rectangle class with two data members’ length and width Provide member functions to calculate the perimeter and area of the rectangle and a function square which returns true if the rectangle is a square otherwise it returns false.***

**Program:**

**Source Code”**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace Practice\_Program

{

public class Rectangle

{

public float width;

public float length;

public float area;

public float perimeter;

public float CalculateArea()

{

area = width \* length;

return area;

}

public float CalculatePerimeter()

{

perimeter = 2 \* (length + width);

return perimeter;

}

public bool Square()

{

if (width == length)

return true;

else

return false;

}

}

class Program

{

static void Main(string[] args)

{

Rectangle MyObj = new Rectangle();

MyObj.length = 45;

MyObj.width = 45;

Console.WriteLine("Length of Rectangle is {0}", MyObj.length);

Console.WriteLine("Width of Rectangle is {0}", MyObj.width);

Console.WriteLine("Area Of Rectangle is {0}", MyObj.CalculateArea());

Console.WriteLine("Perimeter Of Rectangle is {0}", MyObj.CalculatePerimeter());

Console.WriteLine("Square = " + MyObj.Square());

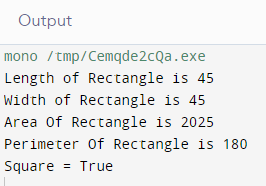
Console.ReadLine();

}

}

}

**Output:**

****

1. ***Write a class to simulate a bank account allowing for initialization, deposits, withdrawals, setting the interest rate, adding interest and giving the current balance. Provide some code to demonstrate the object in action***

**Program:**

**Source Code”**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace Practice\_Program

{

public class Bank

{

public double balance = 0;

public string name;

public double account, withdraw, deposit, interesRate, interest;

public void Initialize()

{

Console.Write("Enter Account Name : ");

name = Console.ReadLine();

Console.Write("Enter Account Number : ");

account = Convert.ToDouble(Console.ReadLine());

interesRate = 6.14;

Console.WriteLine("——————————\n");

Console.WriteLine("Account Name: " + name);

Console.WriteLine("Account Number: " + account);

Console.WriteLine("Interest Rate: " + interesRate + "\n");

}

public void Interest()

{

interest = balance \* interesRate / 100;

balance = balance + interest;

Console.WriteLine("New Balance after Adding Interest: " + balance + "\n");

}

public void CheckBalance()

{

Console.WriteLine("Balance in the Account: " + balance + "\n");

}

public void Deposit()

{

Console.Write("Enter Deposit Amount: ");

deposit = Convert.ToDouble(Console.ReadLine());

balance = balance + deposit;

Console.WriteLine("New Balance in the account: " + balance + "\n");

}

public void Withdraw()

{

Console.Write("Enter Withdraw Amount: ");

withdraw = Convert.ToDouble(Console.ReadLine());

if (withdraw <= balance)

{

balance = balance - withdraw;

Console.WriteLine("After Withdraw Acount Balance is : " + balance + "\n");

}

else

{

Console.WriteLine("\n\nWithdraw Amount does not Exist your Account.");

}

}

}

class Program

{

static void Main(string[] args)

{

Bank bankObj = new Bank();

bankObj.Initialize();

int option;

do

{

option = 0;

Console.WriteLine("Please Select Any Function.");

Console.WriteLine("\nPress 1 To Check Balance. \nPress 2 for Deposit an Amount. \nPress 3 for Withdraw an Amount. \nPress 4 To Add Interest. \nPress 5 To Exit. ");

option = Int32.Parse(Console.ReadLine());

Console.WriteLine("——————————\n");

Console.WriteLine("Account Name: " + bankObj.name);

Console.WriteLine("Account Number: " + bankObj.account);

Console.WriteLine("Interest Rate: " + bankObj.interesRate + "\n");

if (option == 1)

{

bankObj.CheckBalance();

}

else if (option == 2)

{

bankObj.Deposit();

}

else if (option == 3)

{

bankObj.Withdraw();

}

else if (option == 4)

{

bankObj.Interest();

}

else if (option == 5)

{

Console.Clear();

}

else

{

Console.WriteLine("Invalid Entry!!!");

}

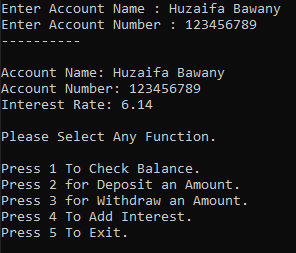
} while (option != 5);

}

}

}

**Output**



# **Lab# 06**

***Tasks***

1. ***Write a program to explain get set accessor.***

class Person

{

private string name;

public string Name

{

get { return name; }

set { name = value; }

}

}

class Program

{

static void Main(string[] args)

{

Person person = new Person();

person.Name = "Huzaifa bawany";

Console.WriteLine("Person's name: " + person.Name);

Console.readkey();

}

}

**Output:**

****

1. ***Write a program to explain method in C#. Create a static function factorial() that accept a number from user and returns factorial of the number.***

using System;

class Program

{

public static int Factorial(int n)

{

int result = 1;

for (int i = 1; i <= n; i++)

{

result \*= i;

}

return result;

}

static void Main(string[] args)

{

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

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

Console.WriteLine("Factorial of " + num + " is " + Factorial(num));

Console.Readkey();

}

}

**Output:**

******

1. ***Write a main method for your Vehicle class that creates a few vehicles and prints out their field values. Make the fields in your Vehicle class private, and add accessor and mutator methods for the fields.***

using System;

c

Console.Readkey();

}

}

**Output:**

****

1. ***Write a C# code which created a public class Employee with its public member variables name , dept and design. Now create method EmpName( ) which takes user input for emaplyee name and then pint the name also. Create another method DeptList( ) that takes input the name of department of employee and third method as EmpDesig( ) which just takes input about employee’s designation. In main call these methods and print appropriate results.***

using System;

public class Employee

{

public string name;

public string dept;

public string design;

public void EmpName()

{

Console.WriteLine("Enter employee name: ");

name = Console.ReadLine();

Console.WriteLine("Employee name: " + name);

}

public void DeptList()

{

Console.WriteLine("Enter employee department: ");

dept = Console.ReadLine();

}

public void EmpDesig()

{

Console.WriteLine("Enter employee designation: ");

design = Console.ReadLine();

}

}

class Program

{

static void Main(string[] args)

{

Employee emp = new Employee();

emp.EmpName();

emp.DeptList();

emp.EmpDesig();

Console.WriteLine("Employee information: ");

Console.WriteLine("Name: " + emp.name);

Console.WriteLine("Department: " + emp.dept);

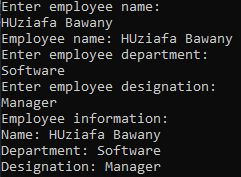
Console.WriteLine("Designation: " + emp.design);

Console.Readkey();

}

}

**Output:**



1. ***Write code which creates a class Calculation and declare methods like Add( ) and Subtract( ). Take two private variables a and b and take user input for these variables that would later be passed to the appropriate method for calculations.***

using System;

class Calculation

{

private int a;

private int b;

public void Add()

{

Console.WriteLine("Enter value of a: ");

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

Console.WriteLine("Enter value of b: ");

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

Console.WriteLine("Result of addition: " + (a + b));

}

public void Subtract()

{

Console.WriteLine("Enter value of a: ");

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

Console.WriteLine("Enter value of b: ");

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

Console.WriteLine("Result of subtraction: " + (a - b));

}

}

class Program

{

static void Main(string[] args)

{

Calculation calc = new Calculation();

Console.WriteLine("Addition: ");

calc.Add();

Console.WriteLine("Subtraction: ");

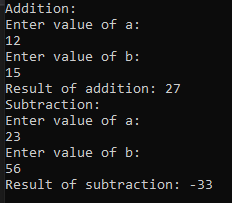
calc.Subtract();

Console.Readkey();

}

}

**Output:**

****

# **Lab# 07**

**TASK**

**Q1** Create an inheritance hierarchy that a bank might use to represent customers’ bank accounts.

All customers at this bank can deposit (i.e., credit) money into their accounts and withdraw (i.e., debit) money from their accounts. More specific types of accounts also exist. Savings accounts, for instance, earn interest on the money they hold. Checking accounts, on the other hand, charge a fee per transaction (i.e., credit or debit).

Create base class Account and derived classes Savings-Account and Checking Account that inherit from class Account. Base class **Account** should include

* Data member **account balance** (double)
* Member function **credit** should add an amount to the current balance.
* Member function **debit** should withdraw money from the Account and ensure that the debit amount does not exceed the Account’s balance. If it does, the balance should be left unchanged and the function should print the message "Debit amount exceeded account balance."
* Member function **getBalance** should return the current balance.

Derived class **SavingsAccount** should inherit Account, include

* A data member **interest rate** (percentage) assigned to the Account.
* Member function **calculateInterest** that returns a double indicating the amount of interest earned by an account.

(Member function calculateInterest should determine this amount by multiplying the interest rate by the account balance)

Derived class **CheckingAccount** should inherit Account and include

* An additional data member of type double that represents the fee charged per transaction.
* Member functions credit and debit so that they subtract the fee from the account balance whenever either transaction is performed successfully.
* CheckingAccount’s versions of these functions should invoke the base-class Account version to perform the updates to an account balance.
* CheckingAccount’s debit function should charge a fee only if money is actually withdrawn

(i.e., the debit amount does not exceed the account balance).

**Program:**

**Source Code:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace inherentance

{

public class Account

{

public double balance = 0;

public string name, accountType;

public double account, debit, credit;

public void Initialize()

{

Console.Write("Enter Your Account Name : ");

name = Console.ReadLine();

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.Write("Enter Your Account Number : ");

account = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

int chooseAccount;

do

{

chooseAccount = 0;

Console.WriteLine("Please Select Your Account Type.");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("\nPress 1 For Saving Account. \nPress 2 for Checking Account.");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

chooseAccount = Int32.Parse(Console.ReadLine());

if (chooseAccount == 1)

{

accountType = "Saving Account";

break;

}

else if (chooseAccount == 2)

{

accountType = "Checking Account";

}

else

{

Console.WriteLine("Invalid Entry!!!");

}

} while (chooseAccount != 2);

//interesRate = 6.14;

Console.WriteLine("——————————\n");

Console.WriteLine("Account Name: " + name);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Account Number: " + account);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Account Type: " + accountType);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

///Console.WriteLine("Interest Rate: " + interesRate + "\n");

}

public void getBalance()

{

Console.WriteLine("Balance in the Account: " + balance + "\n");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

public void Credit()

{

Console.Write("Enter Deposit/Credit Amount: ");

credit = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

balance = balance + credit;

Console.WriteLine("New Balance in the account: " + balance + "\n");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

public void Debit()

{

Console.Write("Enter Withdraw/Debit Amount: ");

debit = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

if (debit <= balance)

{

balance = balance - debit;

Console.WriteLine("After Withdraw/Debit Acount Balance is : " + balance + "\n");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

else

{

Console.WriteLine("\n\nWithdraw/Debit Amount does not Exist your Account.");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

}

}

class SavingsAccount : Account

{

public double interesRate = 6.2, interest;

public void calculateInterest()

{

interest = balance \* interesRate / 100;

///balance = balance + interest;

Console.WriteLine("Interest : " + interest + "\n");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

}

class CheckingAccount : Account

{

public double feePerTransaction = 0;

public void CreditWithFee()

{

Console.Write("Enter Deposit/Credit Amount: ");

credit = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

feePerTransaction = credit \* 1 / 100;

balance = balance + credit - feePerTransaction;

Console.WriteLine("Transaction Fees : " + feePerTransaction);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("New Balance in the account: " + balance + "\n");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

public void DebitWithFee()

{

Console.Write("Enter Withdraw/Debit Amount: ");

debit = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

feePerTransaction = debit \* 1 / 100;

if (debit + feePerTransaction <= balance)

{

balance = balance - debit - feePerTransaction;

Console.WriteLine("Transaction Fees : " + feePerTransaction);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("After Withdraw/Debit Account Balance is : " + balance + "\n");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

else

{

Console.WriteLine("\n\nWithdraw/Debit Amount does not Exist your Account.");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

}

}

class Program

{

static void Main(string[] args)

{

Account accountObj = new Account();

accountObj.Initialize();

if (accountObj.accountType == "Saving Account")

{

SavingsAccount savAccObj = new SavingsAccount();

int option;

do

{

option = 0;

Console.WriteLine("Please Select Any Function.");

Console.WriteLine("\nPress 1 To Check Balance. \nPress 2 for Deposit an Amount. \nPress 3 for Withdraw an Amount. \nPress 4 To Calculate Interest. \nPress 5 To Exit. ");

option = Int32.Parse(Console.ReadLine());

Console.WriteLine("——————————\n");

Console.WriteLine("Account Name: " + accountObj.name);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Account Number: " + accountObj.account);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Account Type: " + accountObj.accountType + "\n");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

if (option == 1)

{

savAccObj.getBalance();

}

else if (option == 2)

{

savAccObj.Credit();

}

else if (option == 3)

{

savAccObj.Debit();

}

else if (option == 4)

{

savAccObj.calculateInterest();

}

else if (option == 5)

{

Console.Clear();

}

else

{

Console.WriteLine("Invalid Entry!!!");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

} while (option != 5);

}

else

{

CheckingAccount checkAccObj = new CheckingAccount();

int option;

do

{

option = 0;

Console.WriteLine("Please Select Any Function.");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("\nPress 1 To Check Balance. \nPress 2 for Deposit an Amount. \nPress 3 for Withdraw an Amount. \nPress 4 To Calculate Interest. \nPress 4 To Exit. ");

option = Int32.Parse(Console.ReadLine());

Console.WriteLine("——————————\n");

Console.WriteLine("Account Name: " + accountObj.name);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Account Number: " + accountObj.account);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Account Type: " + accountObj.accountType + "\n");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

if (option == 1)

{

checkAccObj.getBalance();

}

else if (option == 2)

{

checkAccObj.CreditWithFee();

}

else if (option == 3)

{

checkAccObj.DebitWithFee();

}

else if (option == 4)

{

Console.Clear();

}

else

{

Console.WriteLine("Invalid Entry!!!");

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

}

} while (option != 4);

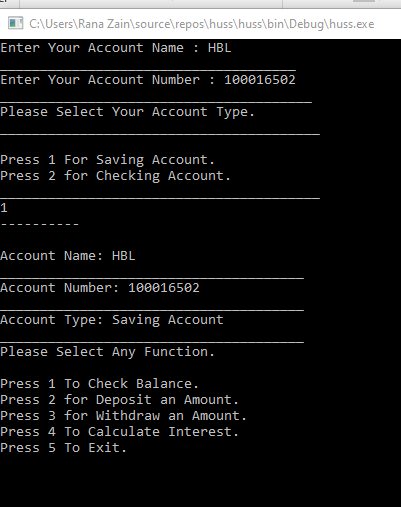
}

}

}

}

**Output:**

****

**Q2**. Generate a marksheet of different departments with proper record and calculate percentage with average of different subjects using Multilevel inheritance.

**Program:**

**Source Code:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace inheretence

{

class Record

{

public string name;

public int age;

public string Department;

public int numberOfSubjects;

public string[] subjectsName = new string[100];

public double[] subjectsMarks = new double[100];

public void initialize()

{

Console.Write("Enter your Full Name : ");

name = Console.ReadLine();

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.Write("Enter your Zip Code Of City : ");

age = Int32.Parse(Console.ReadLine());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.Write("Enter your Department : ");

Department = Console.ReadLine();

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.Write("Enter number of Subjects : ");

numberOfSubjects = Int32.Parse(Console.ReadLine());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

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

{

Console.WriteLine("Enter Subject Name : ");

subjectsName[i] = Console.ReadLine();

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Enter Marks of " + subjectsName[i] + " : ");

subjectsMarks[i] = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

}

}

}

class TotalMarks : Record

{

public double totalMarks = 0;

public double CalculateTotalMarks()

{

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

{

totalMarks = totalMarks + subjectsMarks[i];

}

return totalMarks;

}

}

class Percentage : TotalMarks

{

public double percentage = 0;

public double CalculatePercentage()

{

return percentage = totalMarks / numberOfSubjects;

}

}

class Program

{

static void Main(string[] args)

{

Record RecObj = new Record();

TotalMarks TotalObj = new TotalMarks();

Percentage PerObj = new Percentage();

PerObj.initialize();

Console.WriteLine("Name : " + PerObj.name);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Age : " + PerObj.age);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

for (int i = 0; i < PerObj.numberOfSubjects; i++)

{

Console.WriteLine("Marks of " + PerObj.subjectsName[i] + " : " + PerObj.subjectsMarks[i]);

}

Console.WriteLine("Total Marks : " + PerObj.CalculateTotalMarks());

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

Console.WriteLine("Percentage : " + PerObj.CalculatePercentage() + "%");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ");

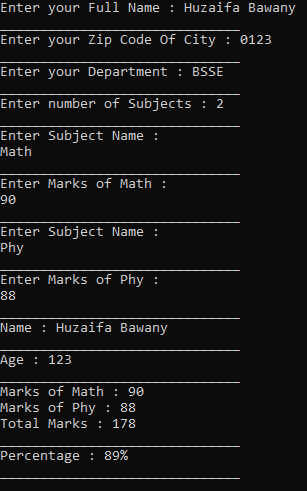
Console.ReadLine();

}

}

}

**output:**

****

# **Lab# 08**

**TASK**

1. **Create a class Vehicle which contains following data members, constructors and functions**

**string model;**

**string color;**

**string engine;**

**int door;**

**constructors:**

**Vehicle(),**

**Vehicle(string, string, string, string,int),**

**function:**

**void start();**

**void stop();**

**void run();**

**virtual void calculateSpeed()=0;**

**void printInfo();**

**Inherit car and bike class from the vehicle class and override the virtual method calculateSpeed() in both classes**

**Program:**

**Source Code:**

using System;

namespace Constructor\_and\_Polymorphism

{

public class Vehicle

{

public string model;

public string color;

public string engine;

public int door;

public void start()

{

Console.WriteLine(model + "has been started!");

}

public void stop()

{

Console.WriteLine(model + "has been stopped!");

}

public void run()

{

Console.WriteLine(model + "has been running!");

}

public virtual void calculateSpeed(int distance, int time)

{

}

public void printInfo()

{

Console.WriteLine("Model : " + model);

Console.WriteLine("Engine : " + engine);

Console.WriteLine("Color : " + color);

Console.WriteLine("Doors : " + door);

}

public Vehicle()

{

}

public Vehicle(string model, string color, string engine, int door)

{

this.model = model;

this.color = color;

this.engine = engine;

this.door = door;

}

}

public class Car : Vehicle

{

public override void calculateSpeed(int distance, int time)

{

double speed = distance / time;

Console.WriteLine("Speed : " + speed);

Console.WriteLine();

}

}

public class Bike : Vehicle

{

public override void calculateSpeed(int distance, int time)

{

double speed = distance / time;

Console.WriteLine("Speed : " + speed);

Console.WriteLine();

}

}

class Program

{

static void Main(string[] args)

{

Bike Honda = new Bike();

Honda.model = "Honda 125 2022";

Honda.color = "Black";

Honda.engine = "70cc";

Honda.door = 0;

Honda.printInfo();

Honda.calculateSpeed(100, 10);

Car Corolla = new Car();

Corolla.model = "Altis 2022";

Corolla.color = "white";

Corolla.engine = "650cc";

Corolla.door = 4;

Corolla.printInfo();

Corolla.calculateSpeed(1000, 20);

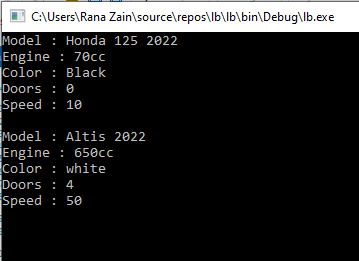
Console.ReadKey();

}

}

}

**Output:**



1. **Write program which performs static polymorphism.**

**Program:**

**Source Code:**

using System;

namespace Constructor\_and\_Polymorphism

{

class Addition

{

public double add(double num1, double num2)

{

return num1 + num2;

}

public double add(double num1, double num2, double num3)

{

return num1 + num2 + num3;

}

}

class Program

{

static void Main(string[] args)

{

Addition ObjAdd = new Addition();

Console.WriteLine("Addition of Two Numbers : " + ObjAdd.add(2, 3));

Console.WriteLine("Addition of Three Numbers : " + ObjAdd.add(2, 4, 6));

Console.ReadKey();

}

}

}

**Output**

# 

# **Lab# 09**

**Lab Assignment:**

* 1. **Write C# code for implementing arrays that accepts user inputs and then displays them. Take different arrays like this: stringArray, integerArray, doubleArray.**

**Program:**

**Source Code:**

using System;

namespace Practice\_program\_array

{

class Program

{

static void Main(string[] args)

{

string[] stringArray = new string[2];

int[] integerArray = new int[2];

double[] DoubleArray = new double[2];

for (int i = 0; i < 2; i++)

{

Console.WriteLine("Enter the string Array", i+1);

stringArray[i] = Console.ReadLine();

Console.WriteLine("Enter the Integer Array " +i+1);

integerArray[i] = int.Parse(Console.ReadLine());

Console.WriteLine("Enter the Doubble Array " +i+1);

DoubleArray[i] = Convert.ToDouble(Console.ReadLine());

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

}

for (int i = 0; i < 2; i++)

{

Console.WriteLine(stringArray[i]);

Console.WriteLine(integerArray[i]);

Console.WriteLine(DoubleArray[i]);

}

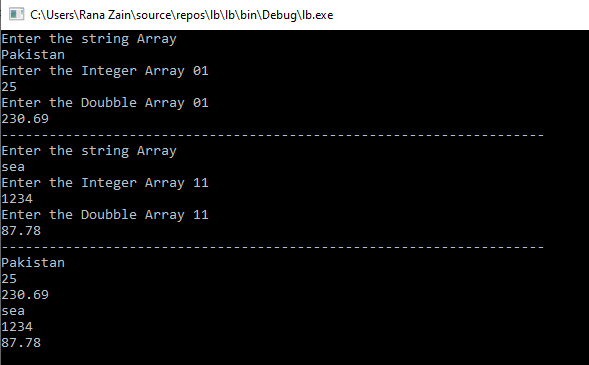
Console.ReadLine();

}

}

}

**Output:**



* 1. **Write a program that adds two arrays and then displays the result. Initialize static array.**

**Program:**

**Source Code:**

// (b) Write a program that adds two arrays and then displays the result. Initialize static array.

using System;

namespace Practice\_program\_array

{

class Program

{

static void Main(string[] args)

{

int[] stringArray1 = {11,12,13};

int[] stringArray2 = { 11,12,13 };

int[] finalstringArray = new int[stringArray1.Length];

for (int i = 0; i < stringArray1.Length; i++)

{

Console.WriteLine("Enter the First Array is " + i);

finalstringArray[i] = stringArray1[i] + stringArray1[i];

}

Console.WriteLine("Addition of no of to Arrays is");

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

{

Console.WriteLine(finalstringArray[i]);

}

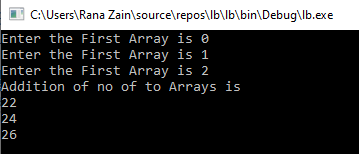
Console.ReadLine();

}

}

}

**Output**



* 1. **Write a program that takes input from user for intMyArray1 and intMyArray2 and then adds up these two arrays and stores result in third array inyMyRsultantArr. The result must also be displayed.**

**Program:**

**Source Code:**

using System;

namespace Practice\_program\_array

{

class Program

{

static void Main(string[] args)

{

int[] stringArray1 = new int[3];

int[] stringArray2 = new int[3];

int[] finalstringArray = new int[3];

for (int i = 0; i < 3; i++)

{

Console.WriteLine("Enter the First Array is " + i);

stringArray1[i] = int.Parse(Console.ReadLine());

Console.WriteLine("Enter the Second Array is " + i);

stringArray2[i] = int.Parse(Console.ReadLine());

finalstringArray[i] = stringArray1[i] + stringArray1[i];

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

}

Console.WriteLine("Addition of no of to Arrays is");

for (int i = 0; i < 3; i++)

{

Console.WriteLine(finalstringArray[i]);

}

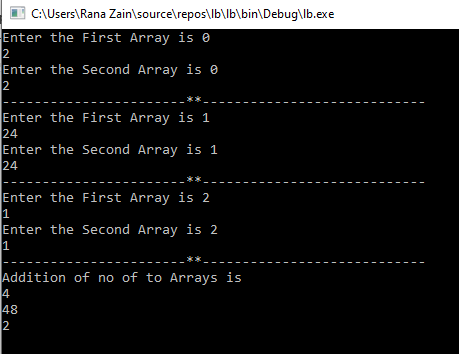
Console.ReadLine();

}

}

}

**Output:**



* 1. **Write a program that displays the elements of array initialized by user after being multiplied by 2.**

**Program:**

**Source Code;**

using System;

namespace Practice\_program\_array

{

class Program

{

static void Main(string[] args)

{

int[] Array1 = new int[3];

for (int i = 0; i < 3; i++)

{

Console.WriteLine("Enter the Elements of Array is " + i);

Array1[i] = int.Parse(Console.ReadLine());

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

}

Console.WriteLine("Addition of no of to Arrays is");

for (int i = 0; i < 3; i++)

{

Console.WriteLine(Array1[i]\*2);

}

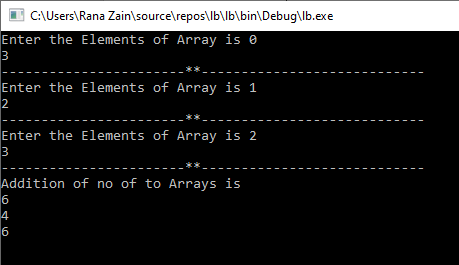
Console.ReadLine();

}

}

}

**Output:**



* 1. **Write a program that inputs two arrays from user and then multiply both arrays and displays result into another array.**

**Program:**

**Source Code:**

using System;

namespace Practice\_program\_array

{

class Program

{

static void Main(string[] args)

{

int[] Array1 = new int[3];

int[] Array2 = new int[3];

int[] Array3 = new int[3];

for (int i = 1; i < 3; i++)

{

Console.WriteLine("Enter the Elements of First Array is ");

Array1[i] = int.Parse(Console.ReadLine());

Console.WriteLine("Enter the Elements of Second Array is ");

Array2[i] = int.Parse(Console.ReadLine());

Array3[i] = Array1[i] \* Array2[i];

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

}

Console.WriteLine("Multiplication of Two Arrays is");

for (int i = 0; i < 3; i++)

{

Console.WriteLine(Array3[i]);

}

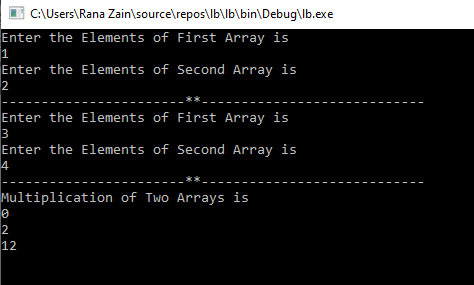
Console.ReadLine();

}

}

}

**Output:**



* 1. **Write a program that implements a PrintArray function which displays array being passed to it. Initialize a string array in Main and then pass it to the function.**

**Program;**

**Source Code:**

using System;

namespace Practice\_program\_array

{

class Program

{

public void PrintArray(string[] args)

{

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

{

Console.WriteLine(args[i]);

}

}

static void Main(string[] args)

{

string[] Array1 = new string[3];

for (int i = 0; i < 3; i++)

{

Console.WriteLine("Enter a string is "+i);

Array1[i] = Console.ReadLine();

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

}

Console.WriteLine("Value of StringArray Of A FUNCTION / Method is ");

Program obj = new Program();

obj.PrintArray(Array1);

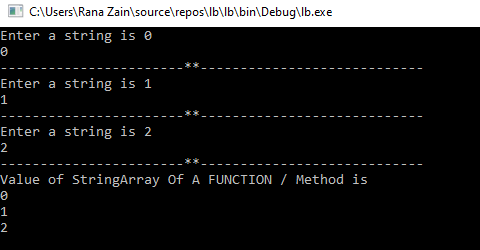
Console.ReadLine();

}

}

}

**Output:**



**[Instruction: The code must be well commented.]**

# **Lab# 10**

**TASK**

1. Write a program to demonstrates the multiple inheritance

**Program:**

**Source Code:**

using System;

namespace Practiceinterface

{

interface Addition

{

int add(int a, int b);

}

interface Subtraction

{

int sub(int a, int b);

}

class Calculate : Addition, Subtraction

{

public int add(int a, int b)

{

return a + b;

}

public int sub(int a, int b)

{

return a - b;

}

}

class Program

{

static void Main(string[] args)

{

Calculate obj = new Calculate();

Console.WriteLine("Enter First Number");

int num1 = int.Parse(Console.ReadLine());

Console.WriteLine("Enter Second Number");

int num2 = int.Parse(Console.ReadLine());

Console.WriteLine("Addition : " + obj.add(num1, num2));

Console.WriteLine("Subtraction : " + obj.sub(num1, num2));

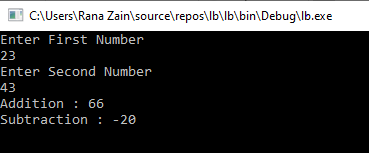
Console.ReadKey();

}

}

}

**Output:**



1. Create an interface **Itransaction** contain void showTransaction(), double getAmount() ,Create class **Transaction** that inherit **Itransaction** and contains private string tCode; private string date; private double amount; add default constructor and parametric constructor to initialize values and implement void showTransaction(), double getAmount()in class.

**Program:**

**Source Code;**

using System;

namespace Practiceinterface

{

interface Itransaction

{

void showTransaction();

double getAmount();

}

class Transaction : Itransaction

{

private string tCode;

private string date;

private double amount;

public Transaction()

{

}

public Transaction(string tcode, string date, double amount)

{

this.tCode = tcode;

this.date = date;

this.amount = amount;

}

public void showTransaction()

{

Console.WriteLine("\nTrassaction Code : " + tCode);

Console.WriteLine("Amount : " + amount + "Rs");

Console.WriteLine("Date : " + date);

}

public double getAmount()

{

return amount;

}

}

class Program

{

static void Main(string[] args)

{

Console.Write("Enter Transaction Code : ");

string tcode = Console.ReadLine();

Console.Write("Enter Date for Transaction : ");

string date = Console.ReadLine();

Console.Write("Enter Amount : ");

double amount = double.Parse(Console.ReadLine());

Transaction Obj = new Transaction(tcode, date, amount);

Obj.showTransaction();

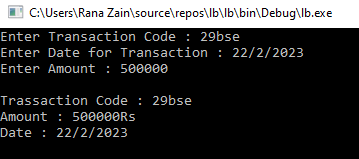
Console.ReadKey();

}

}

}

**Output:**



1. Write C# code for implementing interface ‘**IAnimal**’ which declares a method **AName( )** in it and then class **InterImp** implements this method being inherited from interface IAnimal. Method AName will simply print the name of the animal like “ I am a cat and my name is:” +name.

**Program:**

**Source Code:**

using System;

namespace PracticeInterface

{

interface IAnimal

{

void AName();

}

class InterImp : IAnimal

{

string name;

public void AName()

{

Console.WriteLine("The Name of ur Animal is " + name);

}

public InterImp(string name)

{

this.name = name;

}

}

class Animal

{

static void Main(string[] args)

{

Console.Write("Enter Name of your Animal : ");

string name = Console.ReadLine();

InterImp Obj = new InterImp(name);

Obj.AName();

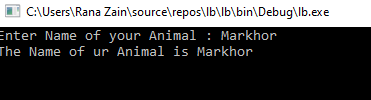
Console.ReadKey();

}

}

}

**Output:**



1. Update the above program by taking user input in method **AName( ).** Input the type of animal whether dog or cat and then the name of the animal. Then this method will be implemented in inherited class.

**Program:**

**Source Code:**

using System;

namespace Practiceinterface

{

interface IAnimal

{

void AName();

}

class InterImp : IAnimal

{

string name;

string type;

public void AName()

{

Console.WriteLine("\nThe Name of ur "+type+" is " + name);

}

public InterImp(string type, string name)

{

this.name = name;

this.type = type;

}

}

class Animal

{

static void Main(string[] args)

{

Console.Write("Enter your Pet's Type : ");

string type = Console.ReadLine();

Console.Write("Enter name of ur "+type+" : ");

string name = Console.ReadLine();

InterImp Obj = new InterImp(type,name);

Obj.AName();

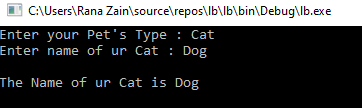
Console.ReadKey();

}

}

}

**Output:**



1. Write a program that implements an interface for AC and declares two methods **ACOn( )** and **ACOff( ).** Both the methods will print apprpiate messages of “AC Off” or “AC On” when implemented and called in inherited class **ACInterImp.**

**Program:**

**Source Code:**

using System;

namespace Practiceinterface

{

interface BG

{

void ACOn();

void ACOff();

}

class ACInterImp : BG

{

public void ACOn()

{

Console.WriteLine("AC On");

}

public void ACOff()

{

Console.WriteLine("AC Off");

}

}

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Press 1 to Turn On AC\nPress 0 to Turn Off AC");

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

ACInterImp Obj = new ACInterImp();

if (key == 1)

{

Obj.ACOn();

}

else if (key == 0)

{

Obj.ACOff();

}

else

{

Console.WriteLine("Invalid Key Pressed...!!");

}

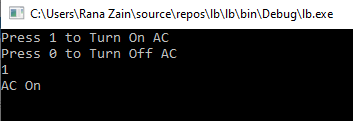
Console.ReadKey();

}

}

}

**Output:**



1. Write a program that implements interface **IShape** that declares method **SName** that inputs name of shape from user and also method **ShArea** that calculates area of shape. [hint: method for area calculation will implement if-then-else and compare the input with shape string to calculate area of a specific shape.]

**Program:**

**Source Code:**

using System;

namespace Practiceinterface

{3

interface IShape

{

void SName();

void ShArea();

}

class Shape : IShape

{

string name;

public void SName()

{

Console.WriteLine("Square , Rectangle , Triangle , Circle");

Console.Write("Enter Name of any Shape from the Given List : ");

name = Console.ReadLine().ToUpper();

}

public void ShArea()

{

if (name == "SQUARE")

{

Console.WriteLine("Enter Length of any Side of Square");

double length = double.Parse(Console.ReadLine());

double area = length \* length;

Console.WriteLine("The Area of Square is " + area + "Sq.meter");

}

else if (name == "TRIANGLE")

{

Console.WriteLine("Enter Lenght of Side 1");

double side1 = double.Parse(Console.ReadLine());

Console.WriteLine("Enter Lenght of Side 2");

double side2 = double.Parse(Console.ReadLine());

Console.WriteLine("Enter Lenght of Side 3");

double side3 = double.Parse(Console.ReadLine());

double s = side1 + side2 + side3 / 2;

double area = Math.Sqrt((s) \* (s - side1) \* (s - side2) \* (s - side3));

Console.WriteLine("The Area of Triangle is " + area + " Sq.meter");

}

else if (name == "CIRCLE")

{

Console.WriteLine("Enter Radius of Circle");

double radius = double.Parse(Console.ReadLine());

double area = Math.PI \* radius \* radius;

Console.WriteLine("The Area of Circle is " + area + " Sq.meter");

}

else if (name == "RECTANGLE")

{

Console.WriteLine("Enter Height of Rectangle");

double height = double.Parse(Console.ReadLine());

Console.WriteLine("Enter Width of Rectangle");

double width = double.Parse(Console.ReadLine());

double area = height \* width;

Console.WriteLine("The Area of Rectangle is " + area + " Sq.meter");

}

else

{

Console.WriteLine("You have Entered An Invalid Shape");

}

}

}

class AreaCalculation

{

static void Main(string[] args)

{

Shape Obj = new Shape();

Obj.SName();

Obj.ShArea();

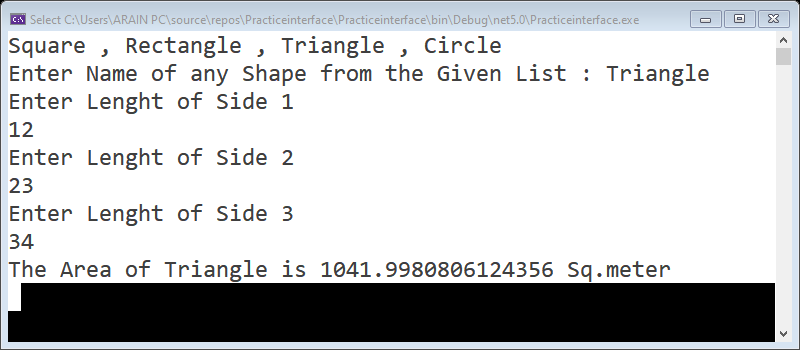
Console.ReadKey();

}

}

}

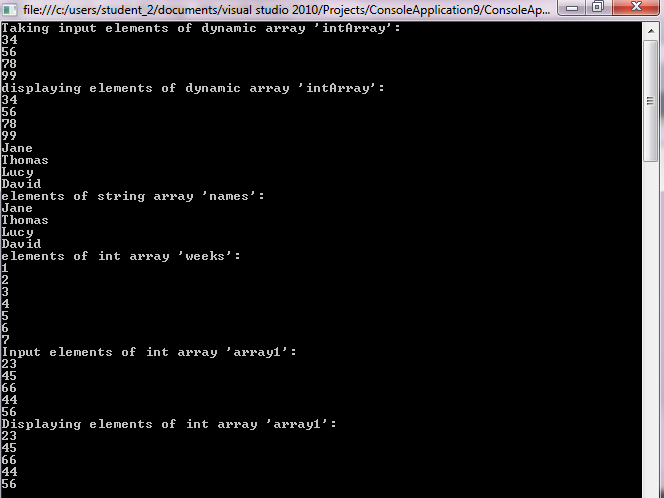
**Output**

****

# 

# **LAB# 11**

**Exercise-1**

****

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

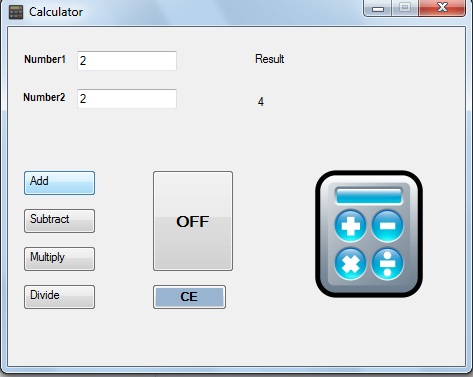
namespace WindowsFormsApp

{

public partial class Calculator : Form

{

public Calculator()

 {

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

double num1 = double.Parse(textBox1.Text);

double num2 = double.Parse(textBox2.Text);

double result = num1 + num2;

label3.Visible = true;

label4.Text = "The Addition of "+num1+" ,and "+num2+" is "+result.ToString();

label4.Visible = true;

textBox1.Text = "";

textBox2.Text = "";

}

private void button2\_Click(object sender, EventArgs e)

{

double num1 = double.Parse(textBox1.Text);

double num2 = double.Parse(textBox2.Text);

double result = num1 - num2;

label3.Visible = true;

label4.Text = "The Subtratcion of " + num1 + " ,and " + num2 + " is " + result.ToString();

label4.Visible = true;

textBox1.Text = "";

textBox2.Text = "";

}

private void button3\_Click(object sender, EventArgs e)

{

double num1 = double.Parse(textBox1.Text);

double num2 = double.Parse(textBox2.Text);

double result = num1 \* num2;

label3.Visible = true;

label4.Text = "The Multiplication of " + num1 + " ,and " + num2 + " is " + result.ToString();

label4.Visible = true;

textBox1.Text = "";

textBox2.Text = "";

}

private void button4\_Click(object sender, EventArgs e)

{

double num1 = double.Parse(textBox1.Text);

double num2 = double.Parse(textBox2.Text);

double result = num1 / num2;

label3.Visible = true;

label4.Text = "The Division of " + num1 + " ,and " + num2 + " is " + result.ToString();

label4.Visible = true;

textBox1.Text = "";

textBox2.Text = "";

}

private void button5\_Click(object sender, EventArgs e)

{

textBox1.Text = "";

textBox2.Text = "";

}

private void button6\_Click(object sender, EventArgs e)

{

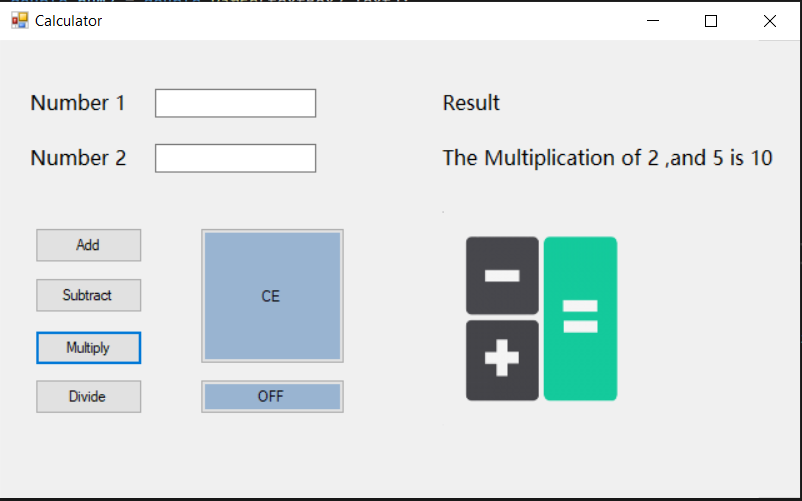
this.Visible = false;

this.Close();

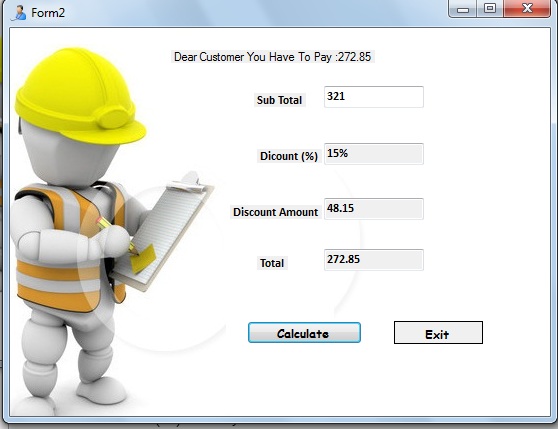
}

}

}



**Exercise-2**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp

{

public partial class Calculator : Form

{

public Calculator()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

double num1 = double.Parse(textBox1.Text);

double num2 = double.Parse(txtSubtotal.Text);

double result = num1 + num2;

label3.Visible = true;

label4.Text = "The Addition of "+num1+" ,and "+num2+" is "+result.ToString();

label4.Visible = true;

textBox1.Text = "";

txtSubtotal.Text = "";

}

private void button2\_Click(object sender, EventArgs e)

{

double num1 = double.Parse(textBox1.Text);

double num2 = double.Parse(txtSubtotal.Text);

double result = num1 - num2;

label3.Visible = true;

label4.Text = "The Subtratcion of " + num1 + " ,and " + num2 + " is " + result.ToString();

label4.Visible = true;

textBox1.Text = "";

txtSubtotal.Text = "";

}

private void button3\_Click(object sender, EventArgs e)

{

double num1 = double.Parse(textBox1.Text);

double num2 = double.Parse(txtSubtotal.Text);

double result = num1 \* num2;

label3.Visible = true;

label4.Text = "The Multiplication of " + num1 + " ,and " + num2 + " is " + result.ToString();

label4.Visible = true;

textBox1.Text = "";

txtSubtotal.Text = "";

}

private void button4\_Click(object sender, EventArgs e)

{

double num1 = double.Parse(textBox1.Text);

double num2 = double.Parse(txtSubtotal.Text);

double result = num1 / num2;

label3.Visible = true;

label4.Text = "The Division of " + num1 + " ,and " + num2 + " is " + result.ToString();

label4.Visible = true;

textBox1.Text = "";

txtSubtotal.Text = "";

}

private void button5\_Click(object sender, EventArgs e)

{

textBox1.Text = "";

txtSubtotal.Text = "";

}

private void button6\_Click(object sender, EventArgs e)

{

this.Visible = false;

this.Close();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

double amount = double.Parse(txtSubtotal.Text);

string discountP = "";

if(amount >= 500)

{

discountP = "20 %";

textBox1.Text = discountP;

textBox4.Text = (amount\*0.2).ToString()+" RS";

textBox3.Text = (amount - (amount \* 0.2)).ToString() + " RS";

label5.Text = "Dear Customer you have to pay " + (amount - (amount \* 0.20)).ToString() + " RS";

}

else if (amount < 500)

{

discountP = "15 %";

textBox1.Text = discountP;

textBox4.Text = (amount \* 0.15).ToString() + " RS";

textBox3.Text = (amount - (amount \* 0.15)).ToString() + " RS";

label5.Text = "Dear Customer you have to pay " + (amount - (amount \* 0.15)).ToString() + " RS";

}

else if (amount < 250)

{

discountP = "10 %";

textBox4.Text = (amount \* 0.10).ToString() + " RS";

textBox3.Text = (amount - (amount \* 0.10)).ToString() + " RS";

label5.Text = "Dear Customer you have to pay " + (amount - (amount \* 0.10)).ToString() + " RS";

}

else if (amount < 100)

{

discountP = "Sorry No discount";

textBox4.Text = (amount \* 0).ToString() + " RS";

textBox3.Text = (amount - (amount \* 0)).ToString() + " RS";

label5.Text = "Dear Customer you have to pay " + (amount - (amount \* 0.0)).ToString() + " RS";

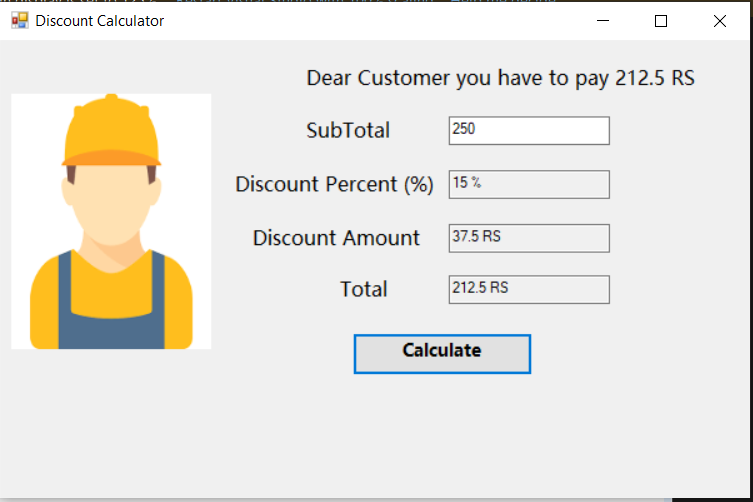
}

label5.Visible = true;

}

}

}



Pressing the Calculate button should give us the following procedure:

Value entered into Subtotal TextBox is transferred to the variable Subtotal.

double Subtotal=Convert.ToDouble(txtSubtotal.Text);

Determine DiscountPercent from the following requirements:

If Subtotal >= 500 then DiscountPercent = 20%.

If Subtotal < 500 and Subtotal >= 250 then DiscountPercent = 15%.

If Subtotal < 250 and Subtotal >= 100 then DiscountPercent = 10%.

Otherwise, DiscountPercent = 0%.

Multiply the Subtotal by the DiscountPercent to get a DiscountAmount.

DiscountAmount = Subtotal \* DiscountPercent;

Subtract the DiscountAmount from the Subtotal to get the InvoiceAmount.

InvoiceAmount = Subtotal - DiscountAmount.

Display the DiscountPercent in the DiscountPercent TextBox.

txtDiscountPercent.Text = DiscountPercent.ToString("p1");

Display the DiscountAmount in the DiscountAmount TextBox.

txtDiscountAmount = DiscoutAmount.ToString("c");

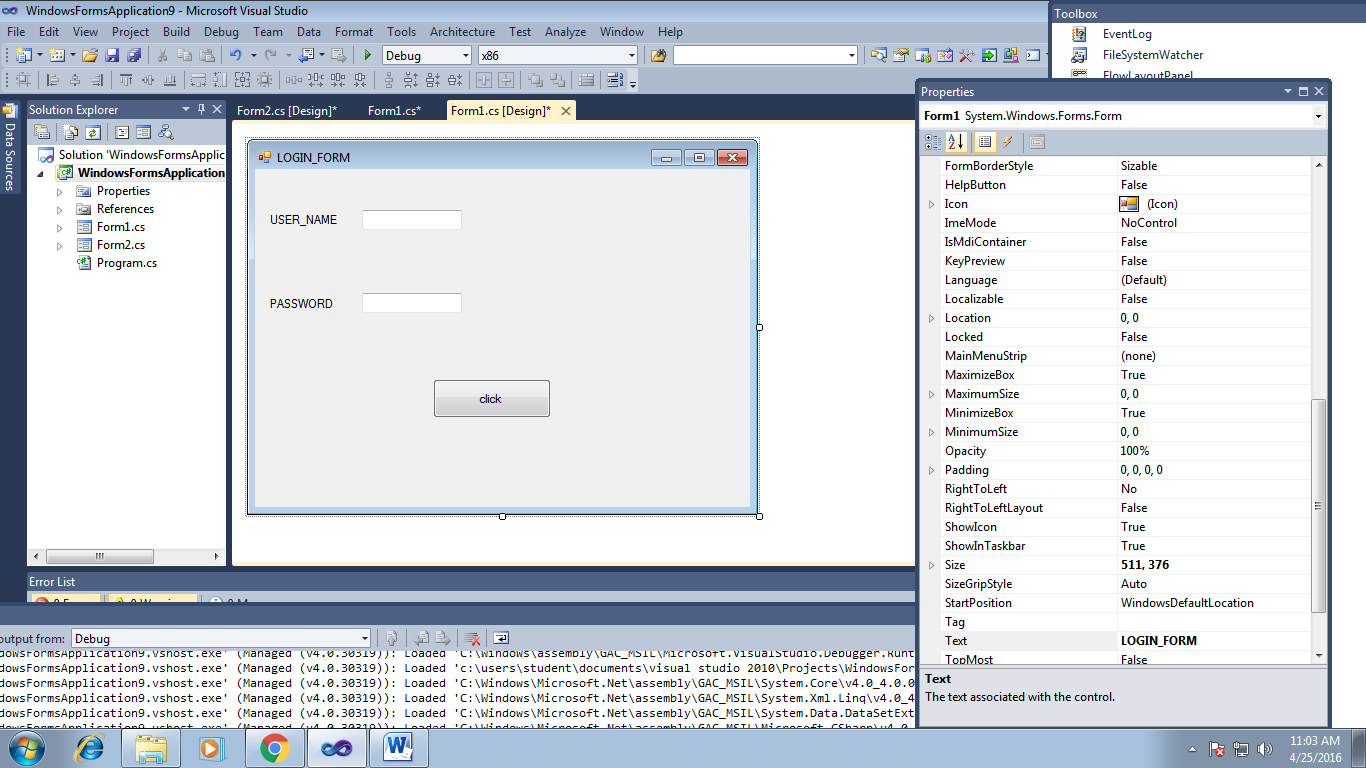
Display the InvoiceTotal in the InvoiceTotal TextBox.

txtInvoiceTotal = InvoiceTotal.ToString("c");

# **Lab# 12**

***Task***

Design following form, using controls discussed in lab manual



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp

{

public partial class Calculator : Form

{

public Calculator()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void button4\_Click(object sender, EventArgs e)

{

}

private void button5\_Click(object sender, EventArgs e)

{

textBox1.Text = "";

textBox2.Text = "";

}

private void button6\_Click(object sender, EventArgs e)

{

this.Visible = false;

this.Close();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

}

private void button1\_Click\_2(object sender, EventArgs e)

{

string user = textBox1.Text;

string pass = textBox2.Text;

if(user == ""&&pass=="")

{

MessageBox.Show("Please Enter all feild");

}

else

{

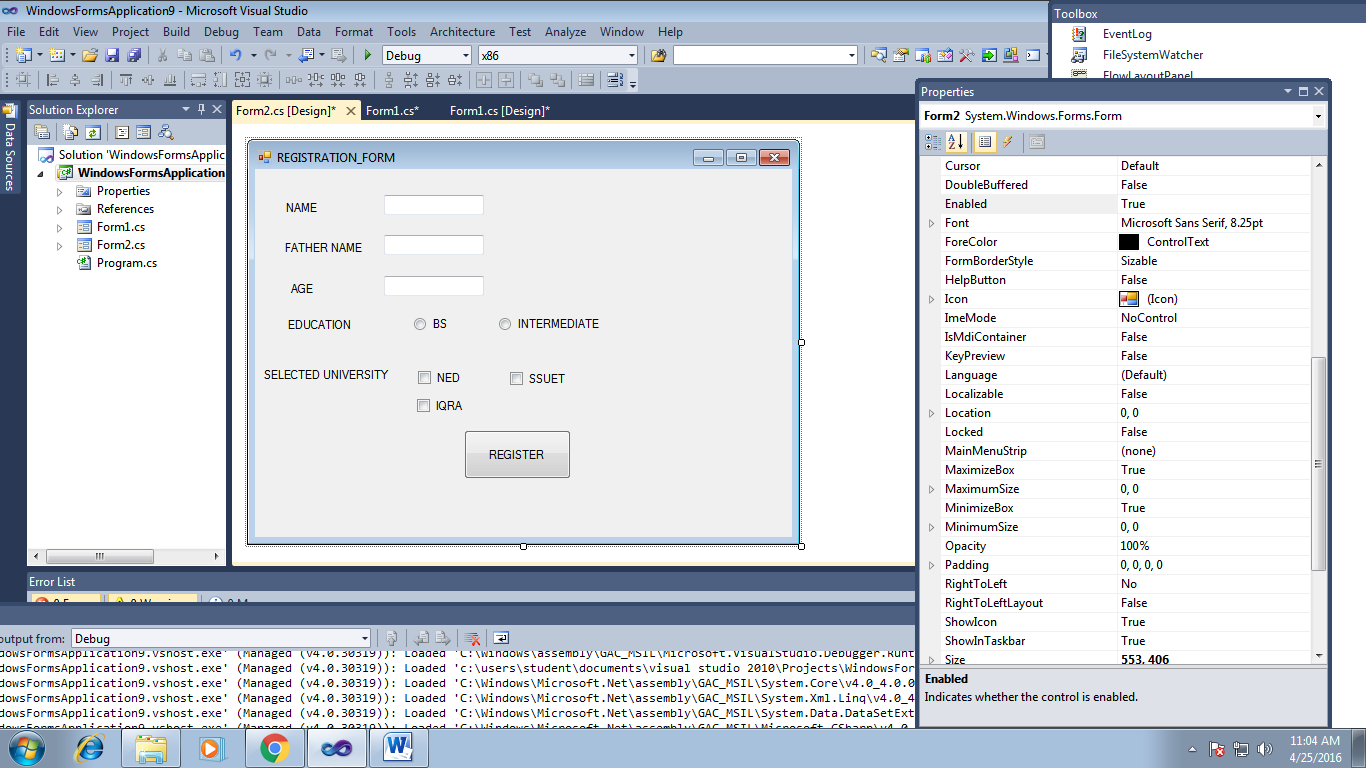
MessageBox.Show("You have successfully login!");

}

}

}

}



**FORM1**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp

{

public partial class Calculator : Form

{

public static string name;

public static string fname;

public static int age;

public static string education;

public static string Uni = "";

public Calculator()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void button4\_Click(object sender, EventArgs e)

{

}

private void button5\_Click(object sender, EventArgs e)

{

}

private void button6\_Click(object sender, EventArgs e)

{

}

private void button1\_Click\_1(object sender, EventArgs e)

{

}

private void button1\_Click\_2(object sender, EventArgs e)

{

}

private void button1\_Click\_3(object sender, EventArgs e)

{

name = textBox2.Text;

fname = textBox1.Text;

age = int.Parse(textBox3.Text);

education = "";

Uni ="";

if (radioButton1.Checked == true)

{

education = "Bechealors";

}

else if (radioButton2.Checked == true)

{

education = "Intermediate";

}

else if (radioButton3.Checked == true)

{

education = "Masters";

}

else {

education = "PHD";

}

if (checkBox1.Checked == true)

{

Uni += " NED";

}

if (checkBox2.Checked == true)

{

Uni += " NUST";

}

if (checkBox3.Checked == true)

{

Uni += " SMIU";

}

if (checkBox4.Checked == true)

{

Uni += " KU";

}

if (checkBox5.Checked == true)

{

Uni += " MAJU";

}

Form2 form = new Form2(name,fname,age,education,Uni);

form.Show();

}

}

}

**FORM2**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp

{

public partial class Form2 : Form

{

public static string name;

public static string fname;

public static int age;

public static string education;

public static string Uni = "";

public Form2()

{

InitializeComponent();

}

public Form2(string names,string father , int ages,string edu,string uni)

{

InitializeComponent();

name = names;

fname = father;

age = ages;

education = edu;

Uni = uni;

}

private void label6\_Click(object sender, EventArgs e)

{

}

private void Form2\_Load(object sender, EventArgs e)

{

label6.Text = name;

label9.Text = fname;

label8.Text = age.ToString();

label7.Text = education;

label10.Text = Uni;

label6.Visible = true;

label7.Visible = true;

label8.Visible = true;

label9.Visible = true;

label10.Visible = true;

}

}}

# **Lab# 13**

**Task:**

Make C# application with database connectivity and perform 4 basic function of add, delete, update and select of database.

**Form 1:**

conn.Open();

string name = textBox2.Text;

string pass = textBox1.Text;

string passv;

textBox2.Text = "";

textBox1.Text ="";

SqlCommand cmd1 = conn.CreateCommand();

cmd1.CommandType = CommandType.Text;

cmd1.CommandText = "SELECT DPASS FROM [TABLE2] WHERE DEMAIL='"+ name + "'";

cmd1.ExecuteNonQuery();

DataTable Table = new DataTable();

SqlDataAdapter sqlDataAda = new SqlDataAdapter(cmd1);

sqlDataAda.Fill(Table);

conn.Close();

try

{

passv = Table.Rows[1][0].ToString();

if (pass == passv)

{

MessageBox.Show("Login Sucessful");

Form2 form2 = new Form2();

form2.Show();

}

else

{

MessageBox.Show("Login UnSucessful");

}

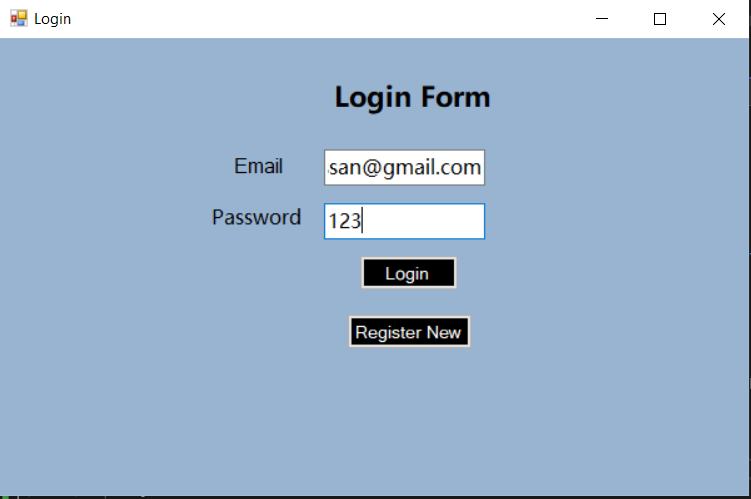
}

catch (Exception ex)

{

MessageBox.Show("Login Unsuccessful "+ex);

}



**Form 2:**

private void button1\_Click(object sender, EventArgs e)

{

string email = textBox1.Text;

string pass = textBox2.Text;

string adminpass = textBox3.Text;

if (adminpass == "iknowpass720")

{

textBox2.Text = "";

textBox1.Text = "";

textBox3.Text = "";

try {

conn.Open();

SqlCommand cmd2 = conn.CreateCommand();

cmd2.CommandType = CommandType.Text;

cmd2.CommandText = "INSERT INTO [TABLE2](DEMAIL,DPASS) VALUES('" + email + "','" + pass + "')";

cmd2.ExecuteNonQuery();

conn.Close();

MessageBox.Show("Registration Sucessful");

this.Hide();

Calculator form = new Calculator();

form.Show();

}

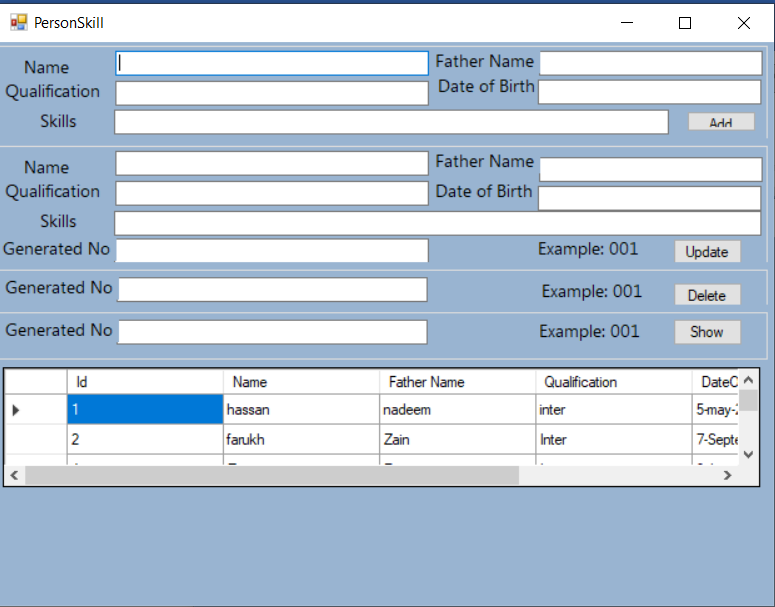
catch(Exception ex)

{

MessageBox.Show("SQL Failure"+ex);

}

}



**Form 3:**

private void button2\_Click(object sender, EventArgs e)

{

try

{

conn.Open();

SqlCommand cmd2 = conn.CreateCommand();

cmd2.CommandType = CommandType.Text;

cmd2.CommandText = "UPDATE [PERSONS] SET Name = '"+textBox7.Text+"', [FATHER NAME] = '"+textBox10.Text+"',Qualification ='"+textBox11.Text+"' ,DateOfBirth= '"+textBox9.Text+"',Skills = '"+textBox8.Text+"' WHERE id ="+int.Parse(textBox6.Text);

cmd2.ExecuteNonQuery();

conn.Close();

MessageBox.Show("Sucessfully Updated!");

}

catch (Exception ex)

{

MessageBox.Show("Update Failed" + ex.Message);

}

}

private void button3\_Click(object sender, EventArgs e)

{

try

{

conn.Open();

SqlCommand cmd2 = conn.CreateCommand();

cmd2.CommandType = CommandType.Text;

cmd2.CommandText = "DELETE FROM [PERSONS] WHERE Id='"+int.Parse(textBox13.Text)+"'";

cmd2.ExecuteNonQuery();

conn.Close();

MessageBox.Show("Sucessfully deleted!");

}

catch (Exception ex)

{

MessageBox.Show("deletion Failed" + ex.Message);

}

}

private void button4\_Click(object sender, EventArgs e)

{

try

{

conn.Open();

SqlCommand cmd2 = conn.CreateCommand();

cmd2.CommandType = CommandType.Text;

cmd2.CommandText = "SELECT \* FROM [PERSONS] WHERE Id='" + int.Parse(textBox12.Text) + "'";

DataTable Table = new DataTable();

SqlDataAdapter sqlDataAda = new SqlDataAdapter(cmd2);

sqlDataAda.Fill(Table);

cmd2.ExecuteNonQuery();

conn.Close();

MessageBox.Show(" Name is "+Table.Rows[0][1].ToString()+"\n Father Name is "+ Table.Rows[0][2].ToString()+"\n Qualification is "+ Table.Rows[0][3].ToString()+"\n Date of Birth is "+ Table.Rows[0][4].ToString()+"\n and Skills is "+ Table.Rows[0][5].ToString());

}

catch (Exception ex)

{

MessageBox.Show("Showing Failed" + ex.Message);

}

}

private void button1\_Click(object sender, EventArgs e)

{

try

{

conn.Open();

SqlCommand cmd2 = conn.CreateCommand();

cmd2.CommandType = CommandType.Text;

cmd2.CommandText = "INSERT INTO [PERSONS](Name,[Father Name],Qualification,DateOfBirth,Skills) VALUES('" + textBox1.Text + "','" + textBox2.Text + "','" + textBox5.Text + "','"+textBox3.Text +"','"+textBox4.Text+ "')";

cmd2.ExecuteNonQuery();

conn.Close();

conn.Open();

SqlCommand cmd1 = conn.CreateCommand();

cmd1.CommandType = CommandType.Text;

cmd1.CommandText = "Select \* from [PERSONS]";

DataTable Table = new DataTable();

SqlDataAdapter sqlDataAda = new SqlDataAdapter(cmd1);

sqlDataAda.Fill(Table);

cmd1.ExecuteNonQuery();

conn.Close();

int num = Table.Rows.Count;

MessageBox.Show("Add Sucessfully and your code is 00"+num);

}

catch (Exception ex)

{

MessageBox.Show("SQL Failure" + ex.Message);

}

textBox2.Text = "";

textBox1.Text = "";

textBox3.Text = "";

textBox4.Text = "";

textBox5.Text = "";

}