NET Centric Computing (CSC-367)
Lab Assignment - Unit 1

Q.1. Write a program to print "Hello world, I am student of B.SC.CSIT".

⇒
```
using System;
class HelloWorld
  {
  static void Main()
     {
     Console.WriteLine("Hello world, I'm student of
       B.SC.CSIT");
     }
  }
```

Output:-
```
Hello world, I'm student of B.SC.CSIT
```

Q.2. Write a program, to check the input year is leap or not.

⇒ Program :-
```
using System;
class LeapYearChecker
  {
  static void Main()
     {
     Console.WriteLine("Enter a year: ");
     int year = int.Parse(Console.ReadLine());
if( (year % 4==0  && year % 100 != 0) || (year % 400==0))
     {
     Console.WriteLine("{0} is a leap year.", year);
     }
else
 {
 Console.WriteLine("{0} is not a leap year.", year);
  }

 }

}
```

Output: Run 1

```
Enter a year: 1992
1992 is a leap year.
```

Run 2

```
Enter a year: 2023
2023 is not a leap year.
```

Q.3. Write a program, for the given condition with the appropriate approach (loop, if, switch), also explain why did you use the specific approach.

a. Age >=18 "Person can get citizenship and vote".

b. Age>=16 "Person can get citizenship but cannot vote".

c. Age<16, "Person cannot get citizenship"

```
Using System;
Class Program
      {
Static void Main(String[] args)
        {
   Console.WriteLine("Enter your age:");
   int age= int.Parse(Console.ReadLine());
   if(age >=18)
         {
     Console.WriteLine("You can get citizenship and vote.");
         }
 else if (age>=16)
        {
   Console.WriteLine("You can get citizenship but cannot vote.");
        }
 else
    {
 Console.WriteLine("You cannot get citizenship.");
    }
   }
 }
```

Output:

```
Enter your age: 25
 You can get citizenship and vote.
```

Explanation:

In this code, I have used if-else statement to check for t different age conditions. If age >=18, then the person can get citizenship and vote. (and others).. I chose to use an if-else statem because it is simple and efficient way to handle multiple conditions in a program.

Q.4. Write a program, create a class named 'Vehicle' with attribute members traveled distance, fuel used, no of seats, action to steer. Add two methods to calculate the kilometers that can be travelled with 1 litre of fuel, and the amount of fuel needed to travel 100 km. Extend the class vehicle by making two new classes Motorcycle and Car. Two methods added should work for Motorcycle and Car class. On object creation, the class Car has to set no of seats to 5 and the motorcycle class has to set to 2. create a method called Steering which takes parameter called direction, when this method is called, it has to set the variable action to steer to "lean over <direction>" for class motorcycle and "rotate steering wheel <direction>" for class Car. Use concept of oop, properties, indexers, constructors wherever applicable.

⇒ Program:

```
  using System;

public class Vehicle {
  private double distanceTraveled;
  private double fuelUsed;
  public int NumSeats { get; set; }
  protected String SteerAction;

  public Vehicle()
      {
      NumSeats = 0;
      SteerAction = " ";
      }
  public void Travel (double distance, double fuel)
      {
      distanceTraveled += distance;
   fuelUsed += fuel;
      }
public double KmPerLiter(){
    if(fuelUsed ==0){
        return 0;
        }
    return distanceTraveled/fuelUsed;
    }
public double FuelPer100km(){
    if(distanceTraveled==0){
        return 0;
        }
      return fuelUsed/(distanceTraveled /100);
      }
   public void Steering(string direction){
        SteerAction= direction;
        }
    }
```

Q.5. Create a simple console application in visual studio. (Book Inventory System, Quiz Application, Student Information System)

Requirements:

a. Use of every concept studied in unit 1(as far as possible)

b. No need to focus on UI, just functionality is needed.

c. Console Application

d. Choose any one from above

Student Information System

```
 Using System;
 Using System.Collections.Generic;
Public class Student {
   Public String Name {get; set;}
   Public int Age {get; set;}
   Public String GradeLevel {get;set;}
Public Student (String name, int age, String gradeLevel){
        Name= name;
        Age=age;
        GradeLevel= gradelevel;
        }

   public void Display() {
      Console.WriteLine ("Name: " + Name);
      Console.WriteLine("Age: "+Age);
      Console.WriteLine("Grade Level: "+ GradeLevel);
         }
     }

class Program {
   static void Main(String[] args) {
   List <Student> students = new List <Student> ();
   while (true) {
       Console.WriteLine("1. Add Student");
      Console.WriteLine ("2. Display All Students");
      Console.Write ("Enter choice: ");
       String choice = Console.ReadLine();

       Switch ( choice )
             {
```

```
 Case "1";
Console.Write(" Enter name:");
String name = Console.ReadLine();
 Console.Write(" Enter age :");
 int age = int.Parse(Console.Readline());
 Console.Write(" Enter grade level:");
 String gradeLevel = Console.ReadLine();
 Student student = new Student(name, age, gradeLevel);
  Students.Add(Student);
 Console.WriteLine ("Student added successfully!");
  break;

Case "2";
      if(Students.Count == 0)
         {
        Console.WriteLine ("No students added yet!");
            }
        else {
        Console.WriteLine(" Student List: ");
      foreach (Student s in students) {
                S.Display();
             Console.WriteLine();
             }
          }
        break;
 Case "3";
      Console. Write Line(" Good Bye!");
        return;
 default:
     Console. WriteLine("Invalid choice, Please try again")
         break;
        }    Console.WriteLine(); } } }
```

Output:-

```
 1. Add student
 2. Display All Students
  3. Exit
Enter choice: 1
Enter name: ankit
Enter age: 21
Enter grade level: 16
Student added successfully!
 1. Add student
 2. Display All Studens
 3. Exit
  Enter choice : 2
  Student List:
       Name: ankit
       Age: 21
        Grade Level: 5
  1. Add student
  2. Display All Students
  3. Exit
    Enter choice: 3
     Good Bye!
```