Muhammed Ali Masood

1]

a. Create a Person class with Age as a property and read the age with the property

b. Implement a user defined exception to handle Age eligibility to vote (18-120 years eligible.

Others not eligible)

using Basic\_Program;

using System.Net.Http.Headers;

class Demo1

{

public static void Main(string[] args)

{

Console.WriteLine("Enter the age");

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

Person person = new Person(age);

Console.WriteLine("Reading age with property.....");

Console.WriteLine($"Age of person is {person.Age}");

try

{

person.validate\_age(age);

}

catch (InvalidAgeException e)

{

Console.WriteLine(e);

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Basic\_Program

{

internal class Person

{

private int age;

public Person(int age)

{

this.Age = age;

}

public int Age { get => age; set => age = value; }

public void validate\_age(int age)

{

if ((age<18) || (age>120))

{

throw new InvalidAgeException("Sorry, Not eligible to vote");

}

else

{

Console.WriteLine("Eligible to vote");

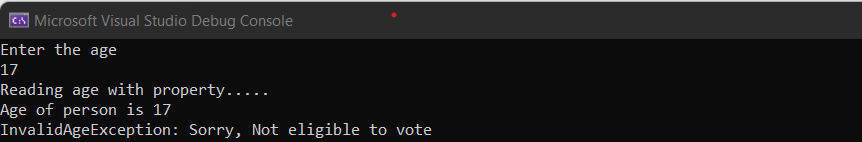
}

}

}

}





c. Implement a Currency Converter Class with choice for at least 5 conversions. Ask choice for

conversion from the users and implement class(es) with suitable methods to display the

converted currency value.

using Basic\_Program;

using System.Net.Http.Headers;

class Demo1

{

public static void Main(string[] args)

{

CurrencyConverter converter = new CurrencyConverter();

converter.Conversations();

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Basic\_Program

{

internal class CurrencyConverter

{

public void Conversations()

{

Console.WriteLine("Enter the choice for Conversion");

Console.WriteLine("1. INR to USD \n2. INR to UAE \n3. INR to EURO \n4. INR to AsutralianDollar \n5. INR to SriLankanRupee");

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

switch(ch)

{

case 1:

Console.WriteLine("Enter the Amount");

double amount1 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine($"{amount1} INR equals " + Math.Round(amount1 \* 0.012, 2) + " Dollar");

break;

case 2:

Console.WriteLine("Enter the Amount");

double amount2 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine($"{amount2} INR equals " + Math.Round(amount2 \* 0.045, 2) + " Dirham");

break;

case 3:

Console.WriteLine("Enter the Amount");

double amount3 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine($"{amount3} INR equals " + Math.Round(amount3 \* 0.011, 2) + " Euro");

break;

case 4:

Console.WriteLine("Enter the Amount");

double amount4 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine($"{amount4} INR equals " + Math.Round(amount4 \* 0.018, 2) + " Australian Dollar");

break;

case 5:

Console.WriteLine("Enter the Amount");

double amount5 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine($"{amount5} INR equals " + Math.Round(amount5 \* 3.89, 2) + " SriLankan Rupee");

break;

default:

Console.WriteLine("Please enter from option 1-5");

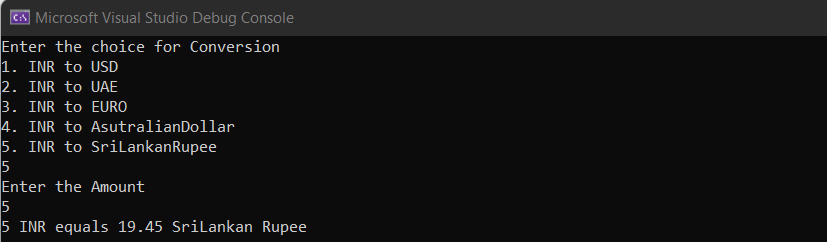
break;

}

}

}

}



Question 2

a. Implement an EB Calculator class with suitable methods to calculate the bill. The class should hold the methods EnterReading() &amp; CalculateBill(). Use appropriate OOP concepts wherever applicable

using Basic\_Program;

using System.Net.Http.Headers;

class Demo1

{

public static void Main(string[] args)

{

EbCalculator calculator = new EbCalculator();

calculator.EnterReading();

Console.WriteLine($"EB charge {calculator.Unit} units is Rs. " + calculator.CalculateBill());

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Basic\_Program

{

internal class EbCalculator

{

private double unit;

private double charge;

public double Unit { get => unit; set => unit = value; }

public double Charge { get => charge; set => charge = value; }

public void EnterReading()

{

Console.WriteLine("Enter the reading");

this.Unit = Convert.ToDouble(Console.ReadLine());

}

public double CalculateBill()

{

if (this.Unit < 200)

this.Charge = 1.20;

else if (this.Unit >= 200 && this.Unit < 400)

this.Charge = 1.50;

else if (this.Unit >= 400 && this.Unit < 600)

this.Charge = 1.80;

else

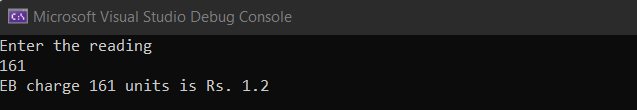
this.Charge = 2.00;

return this.Charge;

}

}

}



b. Implement a Temperature Converter class with suitable methods to display the converted

temperature

using Basic\_Program;

using System.Net.Http.Headers;

class Demo1

{

public static void Main(string[] args)

{

Temperature temperature = new Temperature();

temperature.Conversion();

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Basic\_Program

{

internal class Temperature

{

public void Conversion()

{

Console.WriteLine("1. Celsius to Farenheit Conversion \n2. Farenheit to Celsius COnversion");

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

switch(ch)

{

case 1:

Console.WriteLine("Enter the Temperature in Celsius");

double temp1 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine((temp1 \* 9 / 5) + 32 + "Deg Farenheit");

break;

case 2:

Console.WriteLine("Enter Fahrenheit Temperature:");

double Fahrenheit = Convert.ToDouble(Console.ReadLine()); ;

double Celsius = Math.Round((Fahrenheit - 32) \* 5 / 9, 2);

Console.WriteLine("The converted Celsius temperature is:" + Celsius);

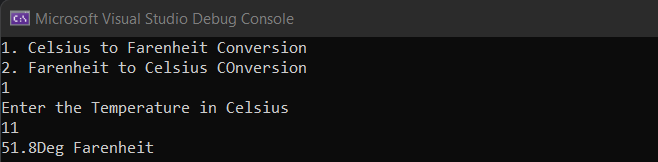
break;

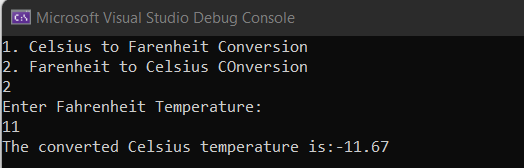
}

}

}

}





Question 3

Create the files. ( allyears.txt, leapyears.txt, nonleapyears.txt ) with appropriate data and do the

following actions.

i. Read the allyears.txt file

ii. Write the leapyears.txt with leap years based on the input from years.txt

iii. Write the nonleapyears.txt with non-leap years based on the input from years.txt

using Basic\_Program;

using System.Net.Http.Headers;

class Demo1

{

public static void Main(string[] args)

{

FileOperations2 fo = new FileOperations2();

fo.read\_file();

fo.write\_to\_leapyear();

fo.write\_to\_nonleapyear();

}

}

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Basic\_Program

{

internal class FileOperations2

{

string file\_path = "D:\\C# training\\allyears.txt";

string leap\_year\_file\_path = "D:\\C# training\\leapyears.txt";

string non\_leapyear\_file\_path = "D:\\C# training\\nonleapyears.txt";

public string File\_path { get => file\_path; set => file\_path = value; }

public string Leap\_year\_file\_path { get => leap\_year\_file\_path; set => leap\_year\_file\_path = value; }

public string Non\_leapyear\_file\_path { get => non\_leapyear\_file\_path; set => non\_leapyear\_file\_path = value; }

public void read\_file()

{

try

{

StreamReader sr = new StreamReader(this.File\_path);

string text = File.ReadAllText(this.File\_path);

Console.WriteLine(text);

sr.Close();

}

catch (FileNotFoundException e)

{

Console.WriteLine(e);

}

}

public void write\_to\_leapyear()

{

using (StreamReader ReaderObject = new StreamReader(File\_path))

{

string line;

while ((line = ReaderObject.ReadLine()) != null)

{

if (DateTime.IsLeapYear(Convert.ToInt32(line)))

{

//File.AppendAllLines(this.Leap\_year\_file\_path, line);

using (StreamWriter leap\_year = File.AppendText(this.Leap\_year\_file\_path))

{

leap\_year.WriteLine(line);

}

}

}

}

Console.WriteLine("Done");

}

public void write\_to\_nonleapyear()

{

using (StreamReader ReaderObject1 = new StreamReader(File\_path))

{

string line;

while ((line = ReaderObject1.ReadLine()) != null)

{

if (!(DateTime.IsLeapYear(Convert.ToInt32(line))))

{

using (StreamWriter non\_leap\_year = File.AppendText(this.Non\_leapyear\_file\_path))

{

non\_leap\_year.WriteLine(line);

}

}

}

}

Console.WriteLine("Done");

}

}

}

