Main function

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace AppointmentConsoleApp

{

class Program

{

static void Main(string[] args)

{

int i = 1;

while (i != 0 )

{

Console.WriteLine("Options:\n\n1.Check availability of date and time and schedule an appointment.\n2.Check Appointment is in afternoon or not.\n3.Check appointments on that date.\n4.Anniversary date\n5.Exit");

Console.Write("\nSelect one option : ");

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

if (id == 1)

{

Console.WriteLine("\nChecking The Date and Time availability...");

Boolean val = Appointment.HasPassed(new TakeDateTime().appointDateTime());

if (val == true)

{

Console.WriteLine("\nThis Date and Time is not available,enter another time.");

Appointment.Schedule(new TakeDateTime().appointDateTime());

}

else

{

continue;

}

}

else if (id == 2)

{

Boolean b = Appointment.IsAfternoonAppointment(new TakeDateTime().appointDate());

if (b == true)

{

Console.WriteLine("\nAfternoon Appointment.");

}

else

{

Console.WriteLine("\nNo Afternoon Appointment.");

}

}

else if (id == 3)

{

string desc = Appointment.Description(new TakeDateTime().appointDate());

Console.WriteLine(desc);

}

else if (id == 4)

{

Console.WriteLine("\nAnniversary date is on {0}", Appointment.AnniversaryDate());

}

else if (id == 5)

{

Console.WriteLine("Press any key to exit");

break;

}

else

{

Console.WriteLine("\nWrong Choice!");

continue;

}

i++;

}

Console.ReadKey();

}

}

}

Appointment Class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace AppointmentConsoleApp

{

static class Appointment

{

static List<DateTime> list = new List<DateTime>();

static DateTime dt1,dt2,dt3,dt4,dtTm;

static string value,desc,anni;

public static void Schedule(DateTime dateTime1)

{

dt1 = dateTime1;

list.Add(dt1);

Console.WriteLine("Appointment is scheduled for :{0}", dt1.ToString());

}

public static Boolean HasPassed(DateTime dateTime2)

{

dt2 = dateTime2;

foreach(var date in list)

{

if (date.Date == dt2.Date)

dtTm = date.Date;

}

if (dtTm == dt2)

{

return true;

}

else

{

Console.WriteLine("This Date and Time is available,applying for an appointment for this date and time.");

Schedule(dt2);

return false;

}

}

public static Boolean IsAfternoonAppointment(DateTime dateTime3)

{

dt3 = dateTime3;

TimeSpan start = new TimeSpan(12, 0, 0);

TimeSpan end = new TimeSpan(18, 0, 0);

TimeSpan onlyTime = new TimeSpan(00,0,0);

foreach (var item in list)

{

if(item.Date == dt3)

onlyTime = item.TimeOfDay;

}

if ((onlyTime >= start) && (onlyTime < end))

{

return true;

}

else

{

return false;

}

}

public static string Description(DateTime dateTime4)

{

dt4 = dateTime4;

foreach(var item in list)

{

if(item.Date == dt4)

{

desc = item.ToString();

}

}

return desc;

}

public static string AnniversaryDate()

{

DateTime anvDt = new DateTime(2021, 01, 15);

anni = anvDt.Date.ToString();

return anni;

}

}

}

TakeDateTime Class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace AppointmentConsoleApp

{

class TakeDateTime

{

public DateTime appointDateTime()

{

Console.WriteLine("Enter Date for Appointment : DD/MM/YYYY HH:MM:SS");

var line = Console.ReadLine();

var separate = line.Split('/',' ',':');

int day = int.Parse(separate[0]);

int month = int.Parse(separate[1]);

int year = int.Parse(separate[2]);

int hour = int.Parse(separate[3]);

int min = int.Parse(separate[4]);

int seconds = int.Parse(separate[5]);

DateTime date = new DateTime(year, month, day, hour, min, seconds);

return date;

}

public DateTime appointDate()

{

Console.WriteLine("Enter Date : DD/MM/YYYY");

var line = Console.ReadLine();

var separate = line.Split('/', ' ', ':');

int day = int.Parse(separate[0]);

int month = int.Parse(separate[1]);

int year = int.Parse(separate[2]);

DateTime date = new DateTime(year, month, day);

return date;

}

}

}