development OF SOFTWARE APPLICATIONS- fall 2020- homework submission report

Author(s):

WU LIANG, HI1LZM

JING XIN, V0EL1R

TASK:

FLIGHT BOOKING MANAGE

# task 1

## Description of the exercise

**Can list available flights. (depending on the occupancy).**

## Solution, explanation

The example code below is written in the program.

private void InitData()

{

Flight flight=new Flight();

flight.FlightName = "AU2312";

flight.From = "Los Angeles";

flight.To = "New York";

flight.FlightDate = new DateTime(2020,12,28);

flight.EconomyAvailable = 60;

flight.BusinessAvailable = 20;

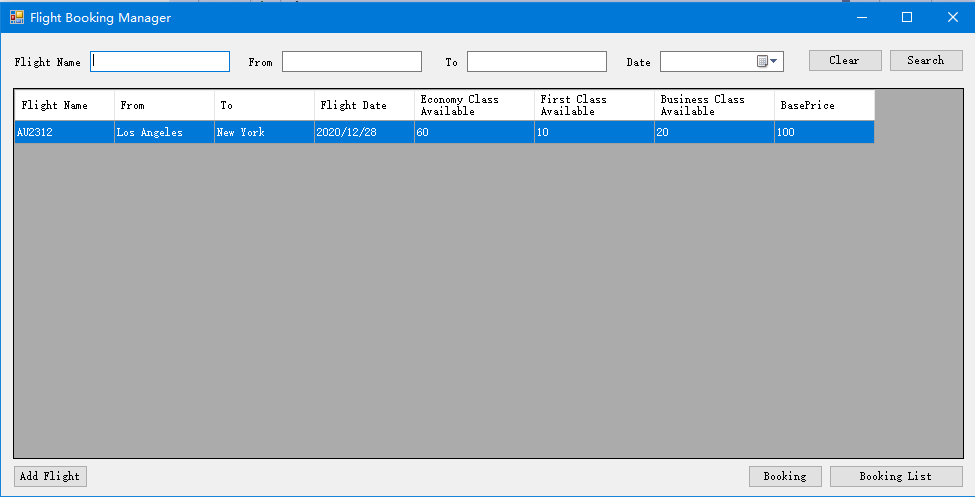
flight.FirstAvailable = 10;

flight.BasePrice = 100;

flights.Add(flight);

}

Below is a screenshot of the result.



List the detail of flight

According to the task requirements, list the specific information of the flight, including the name of the flight, destination, starting place, time.

**using System;**

**using System.Collections.Generic;**

**using System.Linq;**

**using System.Windows.Forms;**

**using FlightBooking.Model;**

**namespace FlightBooking**

**{**

**public partial class BookingListForm : Form**

**{**

**public BookingListForm()**

**{**

**InitializeComponent();**

**ShowTable();**

**}**

**private void ShowTable()**

**{**

**var result = MainForm.Bookings.Select(t => new**

**{**

**FlightName = t.Flight.FlightName,**

**Name = t.Passenger.Name,**

**Number = t.Passenger.Number,**

**IsVip = t.Passenger.IsVip,**

**LuggageType = t.LuggageType,**

**CabinType = t.CabinType,**

**Count = t.PremiumServices.Count,**

**FinalPrice = t.FinalPrice**

**}).ToList();**

**dataGridView1.DataSource = result;**

**}**

**private void Form1\_Load(object sender, EventArgs e)**

**{**

**ShowTable();**

**InitDateTimePicker(this.dateTimePicker1);**

**}**

**public void InitDateTimePicker(DateTimePicker dtp)**

**{**

**dtp.Format = DateTimePickerFormat.Custom;**

**dtp.CustomFormat = " ";**

**dtp.ValueChanged -= DateTimePicker\_ValueChanged;**

**dtp.ValueChanged += DateTimePicker\_ValueChanged;**

**dtp.KeyPress -= DateTimePicker\_KeyPress;**

**dtp.KeyPress += DateTimePicker\_KeyPress;**

**}**

**public void DateTimePicker\_ValueChanged(object sender, EventArgs e)**

**{**

**DateTimePicker dtp = (DateTimePicker)sender;**

**dtp.Format = DateTimePickerFormat.Long;**

**dtp.CustomFormat = null;**

**dtp.Checked = false;**

**}**

**public void DateTimePicker\_KeyPress(object sender, KeyPressEventArgs e)**

**{**

**if (e.KeyChar == (char)8)**

**{**

**DateTimePicker dtp = (DateTimePicker)sender;**

**dtp.Format = DateTimePickerFormat.Custom;**

**dtp.CustomFormat = " ";**

**}**

**}**

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

**{**

**if (dataGridView1.SelectedRows.Count <= 0)**

**{**

**MessageBox.Show("Please select a flight");**

**}**

**else**

**{**

**// BookingForm bookingForm = new BookingForm(flights[dataGridView1.CurrentRow.Index]);**

**// bookingForm.ShowDialog();**

**}**

**}**

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

**{**

**AddFlightForm addFlightForm = new AddFlightForm();**

**addFlightForm.ShowDialog();**

**}**

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

**{**

**String keyword1 = textBox1.Text;**

**List<Booking> temp = new List<Booking>();**

**temp = MainForm.Bookings;**

**if (!String.IsNullOrWhiteSpace(keyword1))**

**{**

**temp = MainForm.Bookings.Where(t => t.Flight.FlightName.Contains(keyword1)).ToList();**

**}**

**String keyword2 = textBox2.Text;**

**if (!String.IsNullOrWhiteSpace(keyword2))**

**{**

**temp = MainForm.Bookings.Where(t => t.Passenger.Name.Contains(keyword2)).ToList();**

**}**

**DateTime dateTime = dateTimePicker1.Value;**

**if (!String.IsNullOrWhiteSpace(dateTimePicker1.Text))**

**{**

**temp = MainForm.Bookings.Where(t => t.BookingDate == dateTime).ToList();**

**}**

**var result = temp.Select(t => new**

**{**

**FlightName = t.Flight.FlightName,**

**Name = t.Passenger.Name,**

**Number = t.Passenger.Number,**

**IsVip = t.Passenger.IsVip,**

**LuggageType = t.LuggageType,**

**CabinType = t.CabinType,**

**Count = t.PremiumServices.Count,**

**FinalPrice = t.FinalPrice**

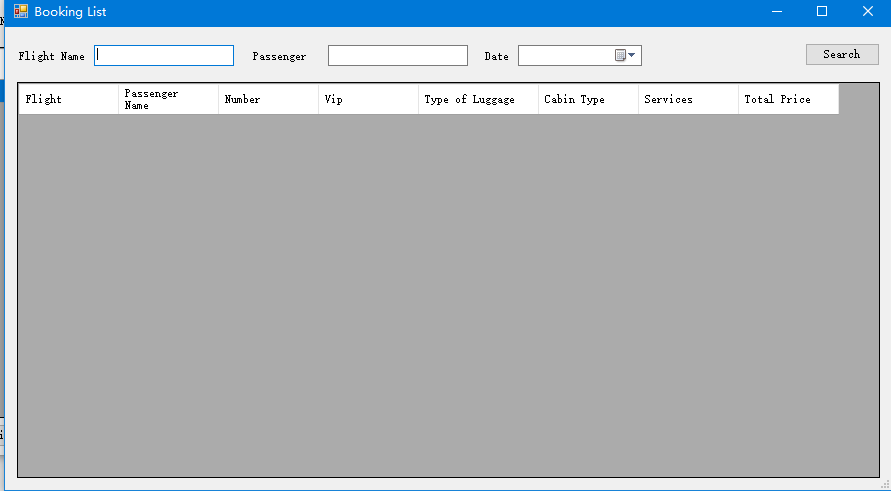
**}).ToList();**

**dataGridView1.DataSource = result;**

**}**

**}**

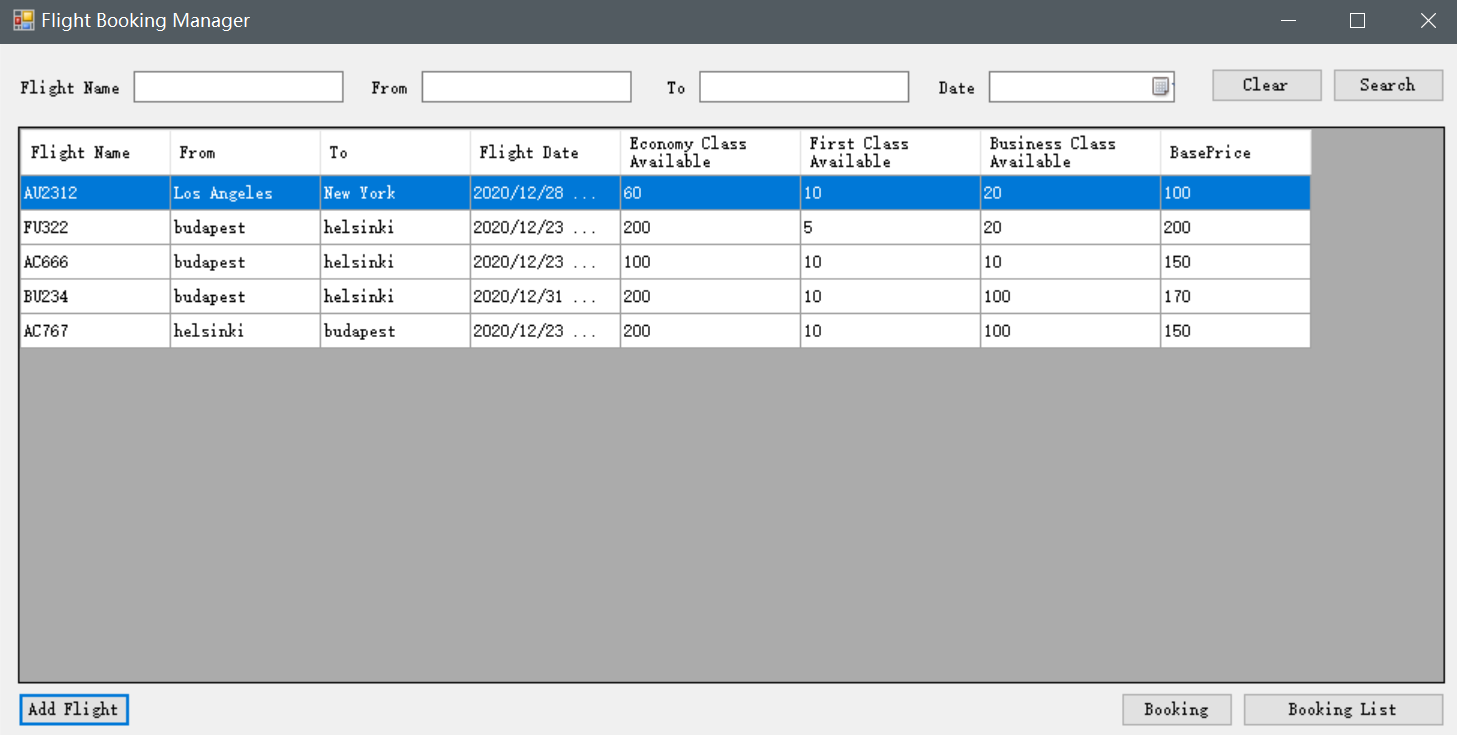
**}**



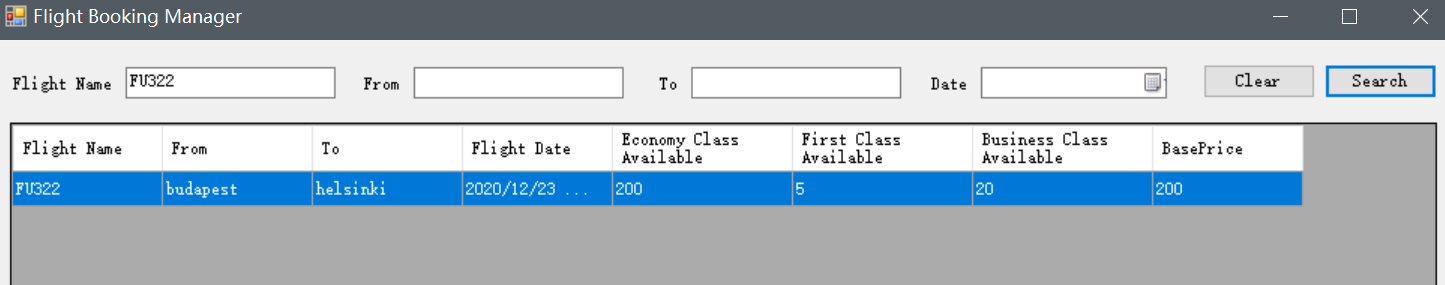
It is Lists

We set up a special interface to see the list of available flights

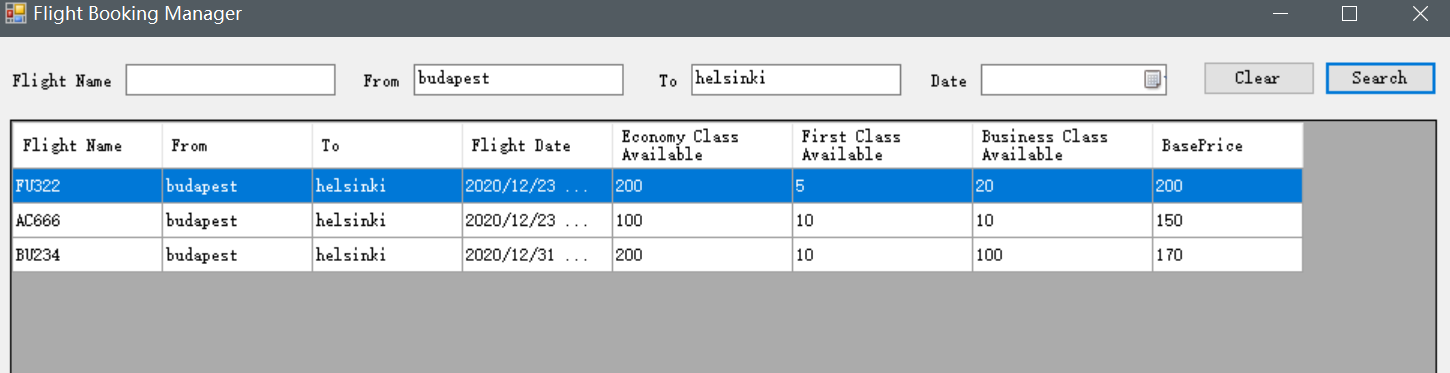
For examples, we added several flight information for different flight number, date, departure and arrival city to test the program.



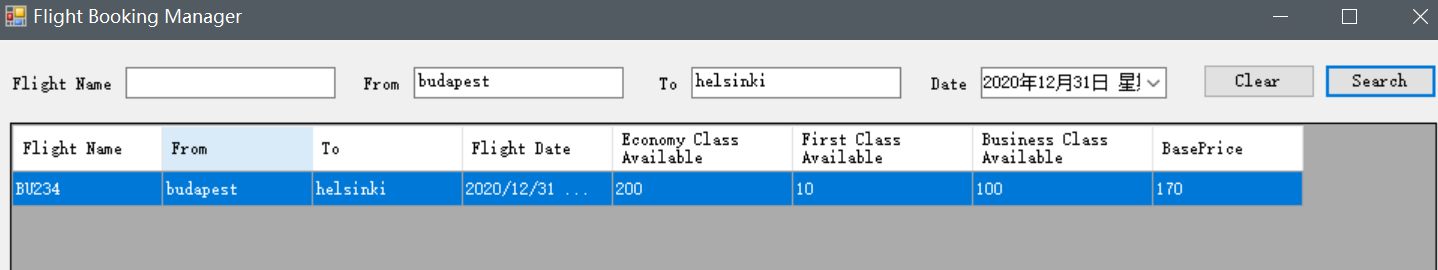
If we want to search the flight according to the flight name:



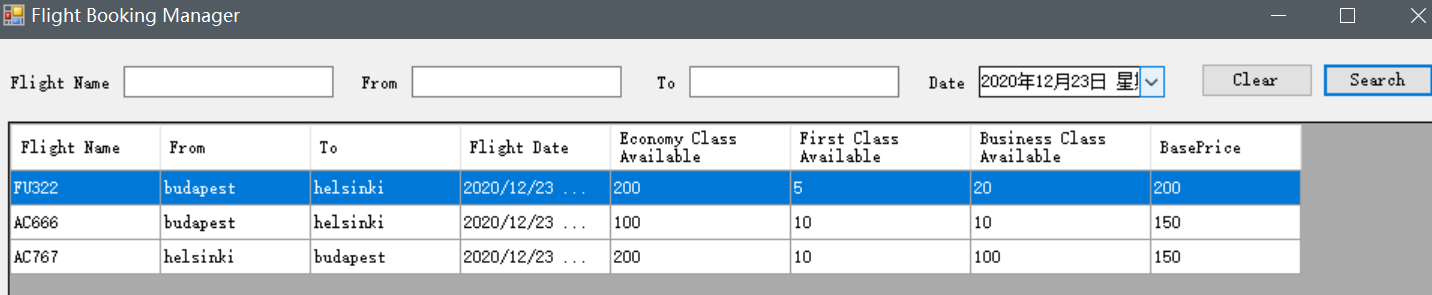
If we want to search the flight according to the departure and arrival city, it can list all flights in different flight name and date:



If we want to search the flight according to the date, departure and arrival city:



If we want to search the flight according to the date, it can list all flights in different flight name and cities:



# task 2

## Description of the exercise

**Can check for free places on a dedicated flight.**

## Solution

The code below is written.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Linq;

using System.Windows.Forms;

using FlightBooking.Model;

namespace FlightBooking

{

public partial class MainForm : Form

{

public static List<Flight> flights = new List<Flight>();

public static List<Flight> searchedFlights = new List<Flight>();

public static List<Booking> Bookings = new List<Booking>();

public MainForm()

{

InitializeComponent();

InitData();

ShowTable();

InitDateTimePicker(this.dateTimePicker1);

}

private void ShowTable()

{

dataGridView1.DataSource = new BindingList<Flight>( flights);

}

private void InitData()

{

Flight flight=new Flight();

flight.FlightName = "AU2312";

flight.From = "Los Angeles";

flight.To = "New York";

flight.FlightDate = new DateTime(2020,12,28);

flight.EconomyAvailable = 60;

flight.BusinessAvailable = 20;

flight.FirstAvailable = 10;

flight.BasePrice = 100;

flights.Add(flight);

}

private void Form1\_Load(object sender, EventArgs e)

{

searchedFlights = flights;

}

public void InitDateTimePicker(DateTimePicker dtp)

{

dtp.Format = DateTimePickerFormat.Custom;

dtp.CustomFormat = " ";

dtp.ValueChanged -= DateTimePicker\_ValueChanged;

dtp.ValueChanged += DateTimePicker\_ValueChanged;

dtp.KeyPress -= DateTimePicker\_KeyPress;

dtp.KeyPress += DateTimePicker\_KeyPress;

}

public void DateTimePicker\_ValueChanged(object sender, EventArgs e)

{

DateTimePicker dtp = (DateTimePicker)sender;

dtp.Format = DateTimePickerFormat.Long;

dtp.CustomFormat = null;

dtp.Checked = false;

}

public void DateTimePicker\_KeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == (char)8)

{

DateTimePicker dtp = (DateTimePicker)sender;

dtp.Format = DateTimePickerFormat.Custom;

dtp.CustomFormat = " ";

}

}

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

{

if (dataGridView1.SelectedRows.Count <= 0)

{

MessageBox.Show("Please select a flight");

}

else {

BookingForm bookingForm = new BookingForm(flights[dataGridView1.CurrentRow.Index]);

bookingForm.ShowDialog();

}

}

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

{

AddFlightForm addFlightForm = new AddFlightForm();

addFlightForm.ShowDialog();

if(addFlightForm.DialogResult==DialogResult.OK)

{

ShowTable();

}

}

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

{

BookingListForm bookingListForm = new BookingListForm();

bookingListForm.ShowDialog();

}

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

{

String keyword1 = textBox1.Text;

if (!String.IsNullOrWhiteSpace(keyword1))

{

searchedFlights = flights.Where(t => t.FlightName.Contains(keyword1)).ToList();

}

String keyword2 = textBox2.Text;

if (!String.IsNullOrWhiteSpace(keyword2))

{

searchedFlights = flights.Where(t => t.From.Contains(keyword2)).ToList();

}

String keyword3 = textBox3.Text;

if (!String.IsNullOrWhiteSpace(keyword3))

{

searchedFlights = flights.Where(t => t.To.Contains(keyword3)).ToList();

}

DateTime dateTime = dateTimePicker1.Value;

if ( !String.IsNullOrWhiteSpace( dateTimePicker1.Text ))

{

searchedFlights = flights.Where(t => t.FlightDate.Date== dateTime).ToList();

}

dataGridView1.DataSource = searchedFlights;

if (String.IsNullOrWhiteSpace(keyword1)&& String.IsNullOrWhiteSpace(keyword2)&& String.IsNullOrWhiteSpace(keyword3)

&& String.IsNullOrWhiteSpace(dateTimePicker1.Text))

{

dataGridView1.DataSource = flights;

}

}

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

{

InitDateTimePicker(this.dateTimePicker1);

textBox1.Text = "";

textBox2.Text = "";

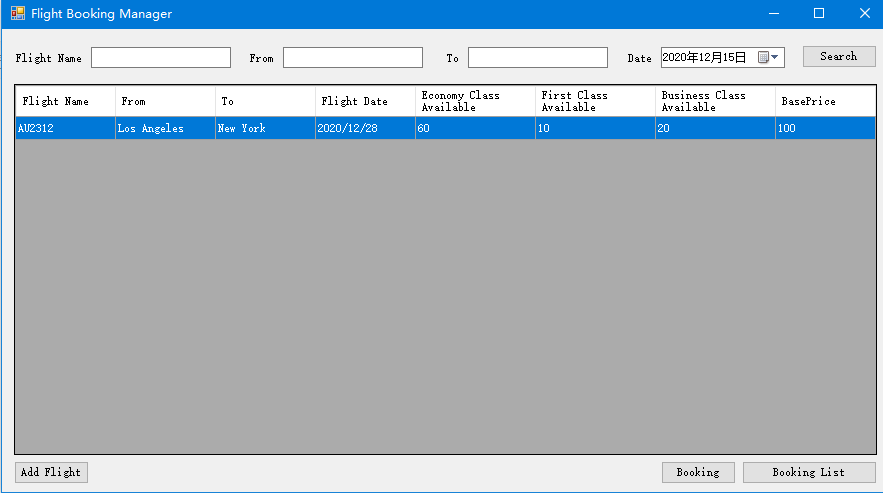
textBox3.Text = "";

}

}

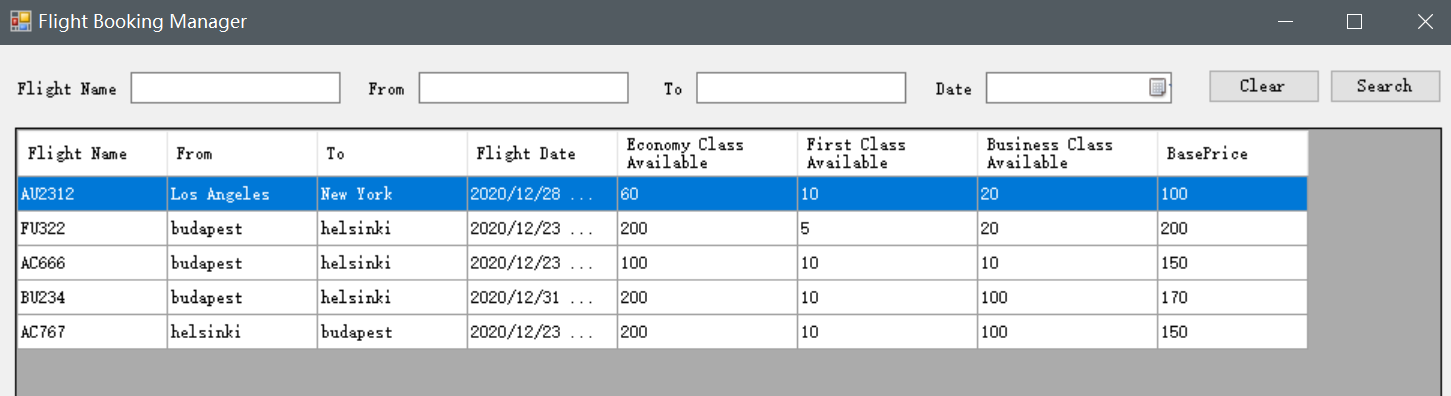
}

Below is a screenshot of the result.

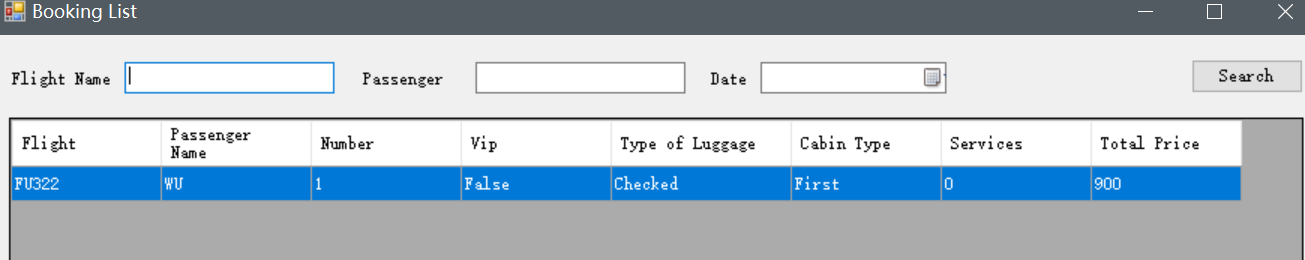


Give the view about surplus seat of three types

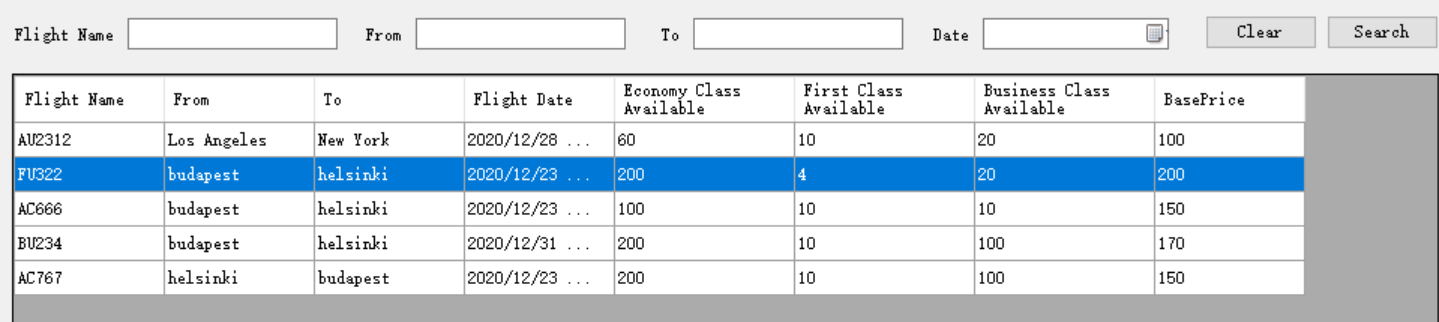
Give the number of seats left in three types of class. If we booked some tickets, we tested the remaining status of tickets.



After we book a FU322 first class ticket for one passenger,



The remaining tickets became:



# task 3

## Description of the exercise

**Are able to assembly booking for a passenger by setting the personal information, number and type of luggage.**

## Solution, explanation

// label7

//

this.label7.AutoSize = true;

this.label7.Location = new System.Drawing.Point(49, 176);

this.label7.Name = "label7";

this.label7.Size = new System.Drawing.Size(108, 19);

this.label7.TabIndex = 17;

this.label7.Text = "Cabin Class";

this.label7.Click += new System.EventHandler(this.label7\_Click);

//

// comboBox1

//

this.comboBox1.FormattingEnabled = true;

this.comboBox1.Location = new System.Drawing.Point(167, 140);

this.comboBox1.Name = "comboBox1";

this.comboBox1.Size = new System.Drawing.Size(175, 27);

this.comboBox1.TabIndex = 16;

//

// textBox7

//

this.textBox7.Location = new System.Drawing.Point(167, 43);

this.textBox7.Name = "textBox7";

this.textBox7.Size = new System.Drawing.Size(175, 26);

this.textBox7.TabIndex = 10;

//

// label5

//

this.label5.AutoSize = true;

this.label5.Location = new System.Drawing.Point(116, 46);

this.label5.Name = "label5";

this.label5.Size = new System.Drawing.Size(45, 19);

this.label5.TabIndex = 11;

this.label5.Text = "Name";

//

// textBox8

//

this.textBox8.Location = new System.Drawing.Point(167, 75);

this.textBox8.Name = "textBox8";

this.textBox8.Size = new System.Drawing.Size(175, 26);

this.textBox8.TabIndex = 12;

//

// label6

//

this.label6.AutoSize = true;

this.label6.Location = new System.Drawing.Point(98, 79);

this.label6.Name = "label6";

this.label6.Size = new System.Drawing.Size(63, 19);

this.label6.TabIndex = 13;

this.label6.Text = "Number";

//

// label11

//

this.label11.AutoSize = true;

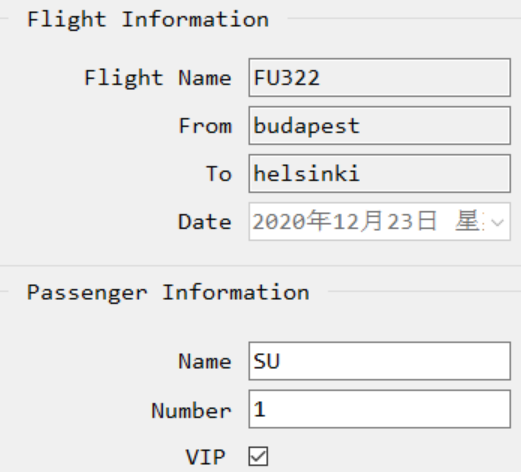
this.label11.Location = new System.Drawing.Point(17, 143);

this.label11.Name = "label11";

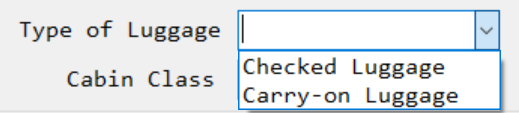
this.label11.Size = new System.Drawing.Size(144, 19);

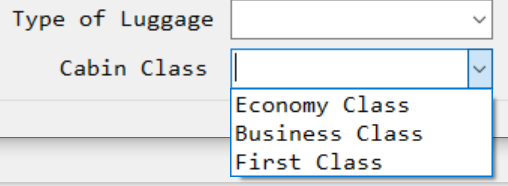
this.label11.TabIndex = 15;

this.label11.Text = "Type of Luggage";



Write personal information, number of luggage and decide whether the passenger is VIP





Fill in the box with name and number of passengers, as well as the type of baggage their need.

# task 4

## Description of the exercise

**Can calculate the price of the booking.**

## Solution, explanation

public partial class BookingForm : Form

{

private Flight flight;

public BookingForm(Flight flight)

{

InitializeComponent();

this.flight = flight;

}

private void label13\_Click(object sender, EventArgs e)

{

}

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

{

if (flight != null)

{

textBox1.Text = flight.FlightName;

textBox2.Text = flight.From;

textBox3.Text = flight.To;

dateTimePicker1.Value = flight.FlightDate;

}

}

private void label7\_Click(object sender, EventArgs e)

{

}

private void checkBox5\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox\_vip.Checked)

{

checkBox\_meal.Enabled = true;

checkBox\_pickup.Enabled = true;

}

else

{

checkBox\_meal.Enabled = false;

checkBox\_pickup.Enabled = false;

}

}

double totalPrice;

double discount;

double finalPrice;

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

{

totalPrice = flight.BasePrice;

double beishu = 1;

if (comboBox\_cabintype.SelectedIndex == 0)

{

beishu = 1;

}

else if (comboBox\_cabintype.SelectedIndex == 1)

{

beishu = 2;

}

else if (comboBox\_cabintype.SelectedIndex == 2)

{

beishu = 4.5;

}

totalPrice = totalPrice \* beishu;

if (checkBox\_meal.Checked)

{

totalPrice = totalPrice + 7;

}

if (checkBox\_pickup.Checked)

{

totalPrice = totalPrice + 12;

}

label10.Text = "Total: $ " + totalPrice;

if (checkBox\_vip.Checked)

{

label12.Text = "Discount: $ " + totalPrice \* 0.1;

discount = totalPrice \* 0.1;

label13.Text = "Final Price: $ " + totalPrice \* 0.9;

finalPrice = totalPrice \* 0.9;

}

else

{

label12.Text = "Discount: $ 0";

discount = 0;

label13.Text = "Final Price: $ " + totalPrice;

finalPrice = totalPrice;

}

}

}

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

**{**

**Booking booking = new Booking();**

**Passenger passenger = new Passenger();**

**passenger.Name = textBox\_pname.Text;**

**passenger.Number = textBox\_pnumber.Text;**

**passenger.IsVip = checkBox\_vip.Checked;**

**booking.Flight = flight;**

**booking.Passenger = passenger;**

**if (comboBox\_ltype.SelectedIndex == 0)**

**{**

**booking.LuggageType = LuggageType.Checked;**

**}**

**else {**

**booking.LuggageType = LuggageType.Carry\_on;**

**}**

**if (comboBox\_cabintype.SelectedIndex == 0)**

**{**

**booking.CabinType = CabinType.Economy;**

**}**

**else if (comboBox\_cabintype.SelectedIndex == 1)**

**{**

**booking.CabinType = CabinType.Business;**

**}**

**else**

**{**

**booking.CabinType = CabinType.First;**

**}**

**List<PremiumService> services = new List<PremiumService>();**

**if (checkBox\_meal.Checked)**

**{**

**PremiumService premiumService = new PremiumService();**

**premiumService.ServiceName = "Airline Meal";**

**premiumService.Price = 7;**

**services.Add(premiumService);**

**}**

**if(checkBox\_pickup.Checked)**

**{**

**PremiumService premiumService = new PremiumService();**

**premiumService.ServiceName = "Airport Pickup";**

**premiumService.Price = 12;**

**services.Add(premiumService);**

**}**

**booking.PremiumServices = services;**

**booking.FinalPrice = finalPrice;**

**booking.TotalPrice = totalPrice;**

**booking.Discount = discount;**

**MainForm.Bookings.Add(booking);**

**MessageBox.Show("successful reservation");**

**if (comboBox\_cabintype.SelectedIndex == 0)**

**{**

**MainForm.flights.First(t => t.FlightName == flight.FlightName).EconomyAvailable--;**

**}**

**else if (comboBox\_cabintype.SelectedIndex == 1)**

**{**

**MainForm.flights.First(t => t.FlightName == flight.FlightName).BusinessAvailable--;**

**}**

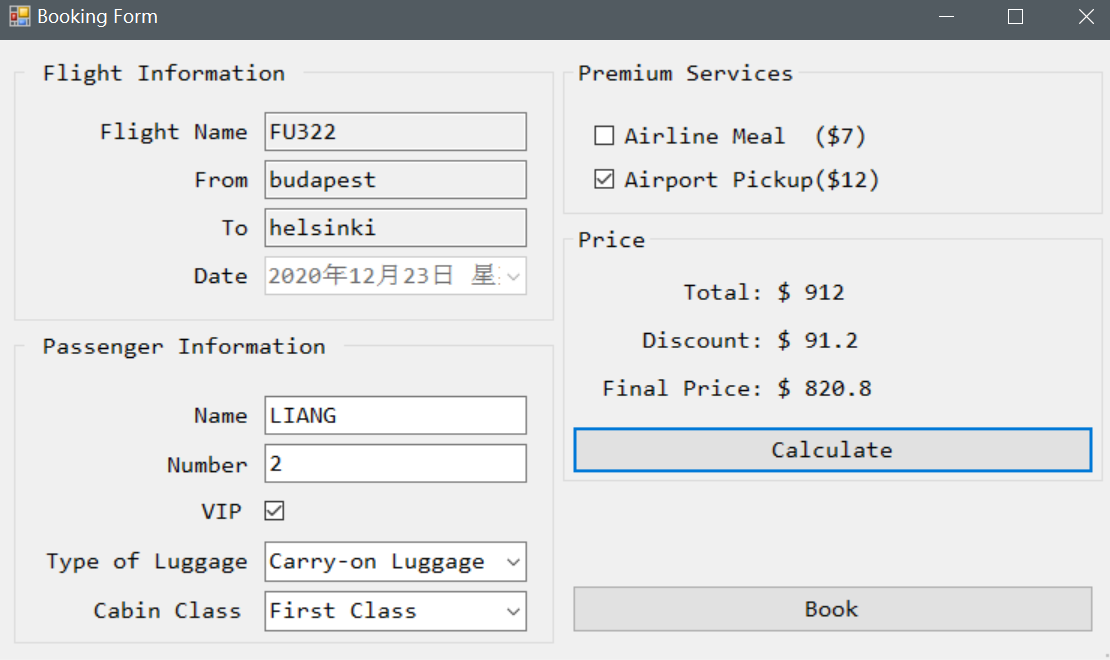
**else**

**{**

**MainForm.flights.First(t => t.FlightName == flight.FlightName).FirstAvailable--;**

**}**

**}**



Can calculate the total price when customer finish selection

When the customer has filled in the booking information, they can click the button to calculate the total price

# task 5

## Description of the exercise

**Can manage VIP passengers who can book** **premium services and can have reduced prices.**

## Solution, explanation.

namespace FlightBooking.Model

{

public class PremiumService

{

public string ServiceName { get; set; }

public double Price { get; set; }

public double VipDiscount { get; set; }

}

}

namespace FlightBooking.Model

{

public class Passenger

{

public string Name { get; set; }

public string Number { get; set; }

public bool IsVip { get; set; }

}

}

namespace FlightBooking.Model

{

public class PremiumService

{

public string ServiceName { get; set; }

public double Price { get; set; }

public double VipDiscount { get; set; }

}

}

if (checkBox1.Checked)

{

totalPrice = totalPrice + 7;

}

if (checkBox2.Checked) {

totalPrice = totalPrice + 12;

}

label10.Text = "Total: $ " + totalPrice;

if (checkBox5.Checked)

{

label12.Text = "Discount: $ " + totalPrice \* 0.1;

label13.Text = "Final Price: $ " + totalPrice \* 0.9;

}

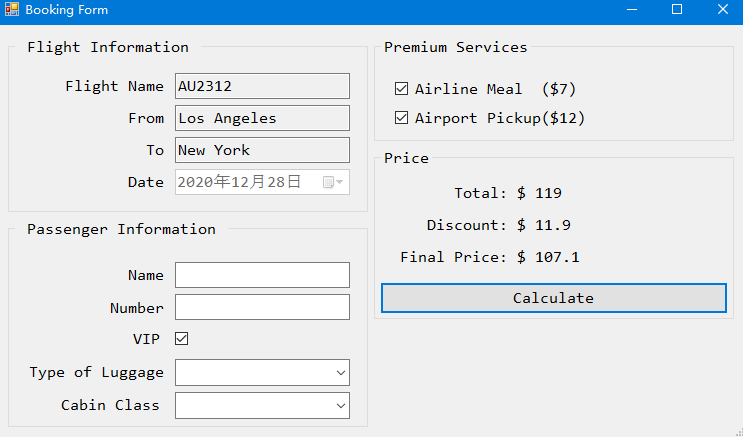
else {

label12.Text = "Discount: $ 0";

label13.Text = "Final Price: $ " + totalPrice ;

}

Below is a screenshot of the result.



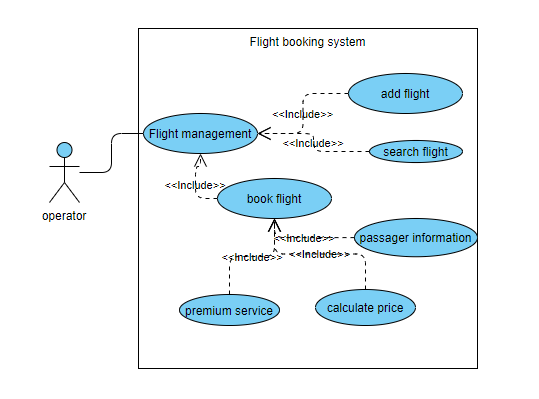
Whether is VIP？

Premium services

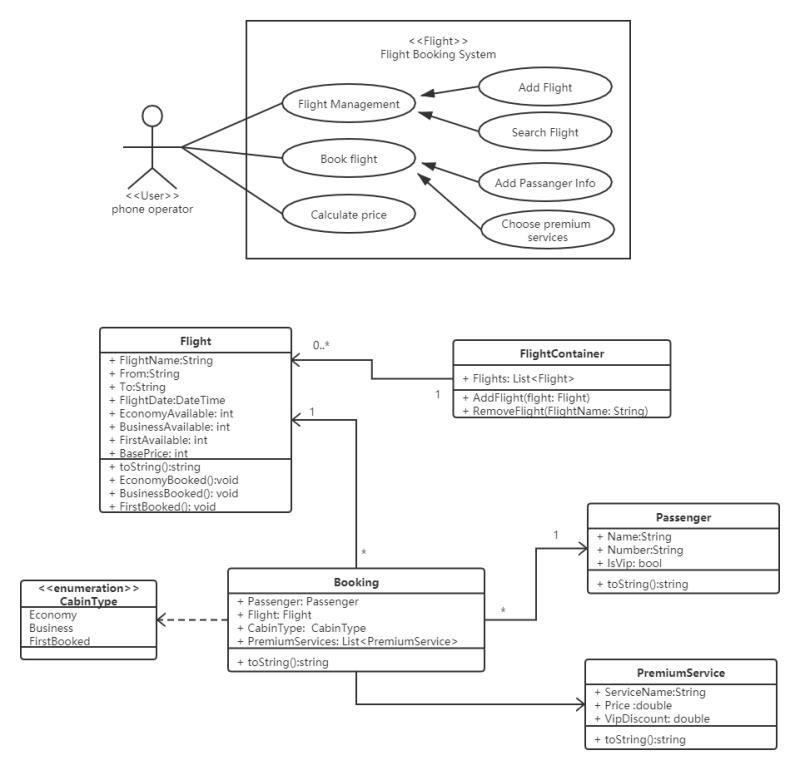
VIP has a discount

When customer booking, provide VIP selection box, he/she can select this box if he/she is a VIP customer. Set the type and price of premium services in advance. If customers need these services, they can choose. Offer 10% discount to VIP customers who choose these premium services

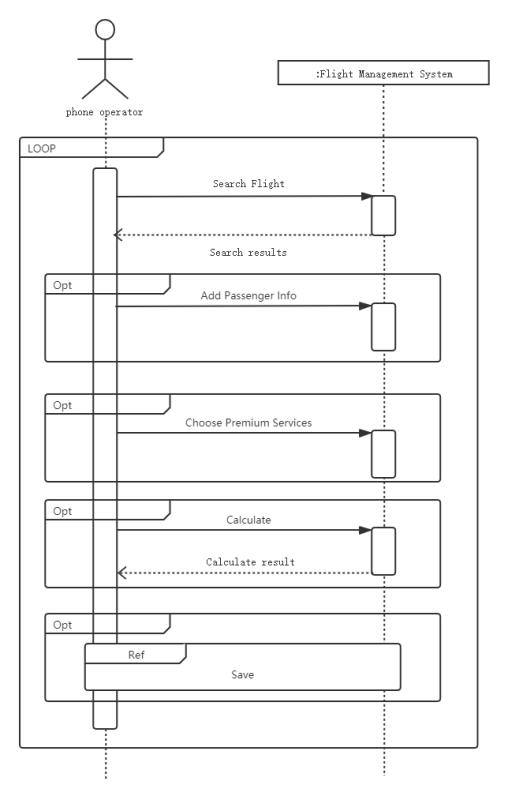
# UML USE CASE DIAGRAM



# UML Class diagram



# UML Sequence diagrams



# EXTRA WORK

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;

using FlightBooking.Model;

namespace FlightBooking

{

public partial class AddFlightForm : Form

{

public AddFlightForm()

{

InitializeComponent();

}

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

{

Flight flight = new Flight();

flight.FlightName = textBox\_name.Text;

flight.From = textBox\_from.Text;

flight.To = textBox\_to.Text;

flight.FlightDate = dateTimePicker\_date.Value.Date;

flight.EconomyAvailable = Convert.ToInt32(textBox\_economy.Text);

flight.BusinessAvailable = Convert.ToInt32(textBox\_bussiness.Text); ;

flight.FirstAvailable = Convert.ToInt32(textBox\_first.Text);

flight.BasePrice = Convert.ToDouble(textBox\_base.Text); ;

MainForm.flights.Add(flight);

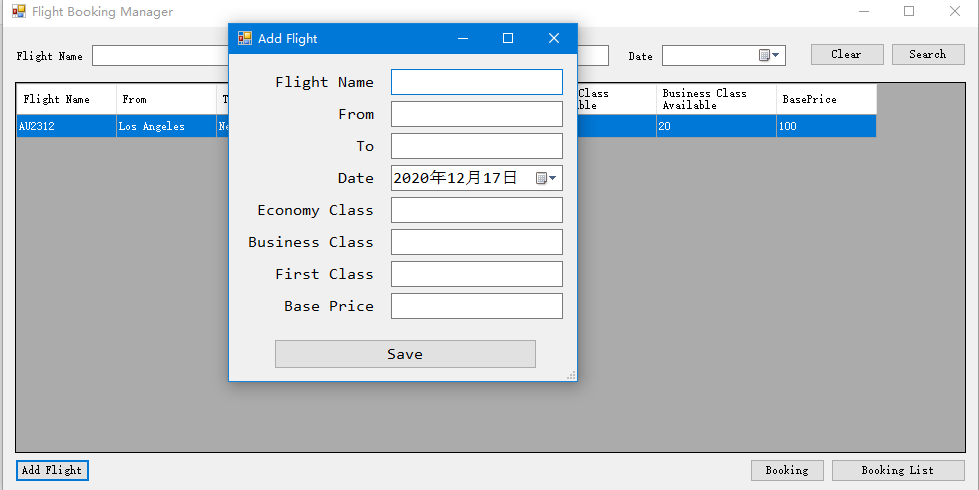
DialogResult = DialogResult.OK;

}

}

}

}



Firstly, the phone operator need to add flight details at the begining of using. In fact, these data should be automatic added from the arrangement of airlines. We just finished the passengers management program. In this work, we use winform to design an GUI to be easily used. It is a part of .Net application GUI. This program can friendly be used enough.