Province of the
EASTERN CAPE
EDUCATION

NATIONAL SENIOR CERTIFICATE

GRADE 12

SEPTEMBER 2023

INFORMATION TECHNOLOGY P1

MARKS: **150**

TIME: **3 hours**

Font size 18

This question paper consists of 44 pages.

INSTRUCTIONS AND INFORMATION

1. This question paper is divided into FOUR sections. Candidates must answer ALL the questions.

2. The duration of this examination is three hours. Because of the nature of this examination, it is important to note that you will not be permitted to leave the examination room before the end of the examination session.

3. This question paper is set with programming terms that are specific to Delphi programming language.

4. Make sure that you answer the questions according to the specifications that are given in each question. Marks will be awarded according to the set requirements.

5. Answer only what is asked in each question. For example, if the question does not ask for data validation, then no marks will be awarded for data validation.

6. Your programs must be coded in such a way that they will work with any data and not just the sample data supplied or any data extracts that appear in the question paper.

7. Routines such as search, sort and selection must be developed from first principles. You may NOT use the built-in features of a programming language for any of these routines.

8. All data structures must be defined by you, the programmer, unless the data structures are supplied.

9. You must save your work regularly on the disk/CD/DVD/flash disk you have been given, or on the disk space allocated to you for this examination session.

10. Make sure that your name appears as a comment in every program that you code, as well as on every event indicated.

11. If required, print the programming code of all the programs/classes that you completed. You will be given half an hour printing time after the examination session.

12. At the end of this examination session, you must hand in a disk/CD/DVD/ flash disk with all your work saved on it OR you must make sure that all your work has been saved on the disk space allocated to you for this examination session. Make sure that all files can be read.

13. Save your work regularly.

14. The files that you need to complete this question paper have been given to you on the disk/CD/DVD/flash disk or on the disk space allocated to you. The files are provided in the form of password-protected executable files.

    Do the following:

    - Double click on the password-protected executable file.
    - Click on the extract button.
    - Enter the following password: **eMSm&tMR@23**

    Once extracted, the following list of files will be available in the folder **DataSept2023**:

    Question 1:

    Question1_p.dpr
    Question1_p.res
    Question1_u.dfm
    Question1_u.pas

    Question 2:

    FlightInfo.mdb
    FlightInfoBackUp.mdb
    dbConnection_u.pas
    Question2_p.dpr
    Question2_p.res
    Question2_u.dfm
    Question2_u.pas

Question 3:

Folder: Images\Flags – (7 Flag Images)
Folder: Images\Aircraft – (17 Aircraft Images)

Aircraft_List.csv
Question3_p.dpr
Question3_p.res
Question3_u.dfm
Question3_u.pas
Question3ClassDefinition.pas

Question 4:

Folder: Destination Textfiles – (5 text files)

Airplane.png
Question4_p.dpr
Question4_p.res
Question4_u.dfm
Question4_u.pas

QUESTION 1: GENERAL PROGRAMMING SKILLS

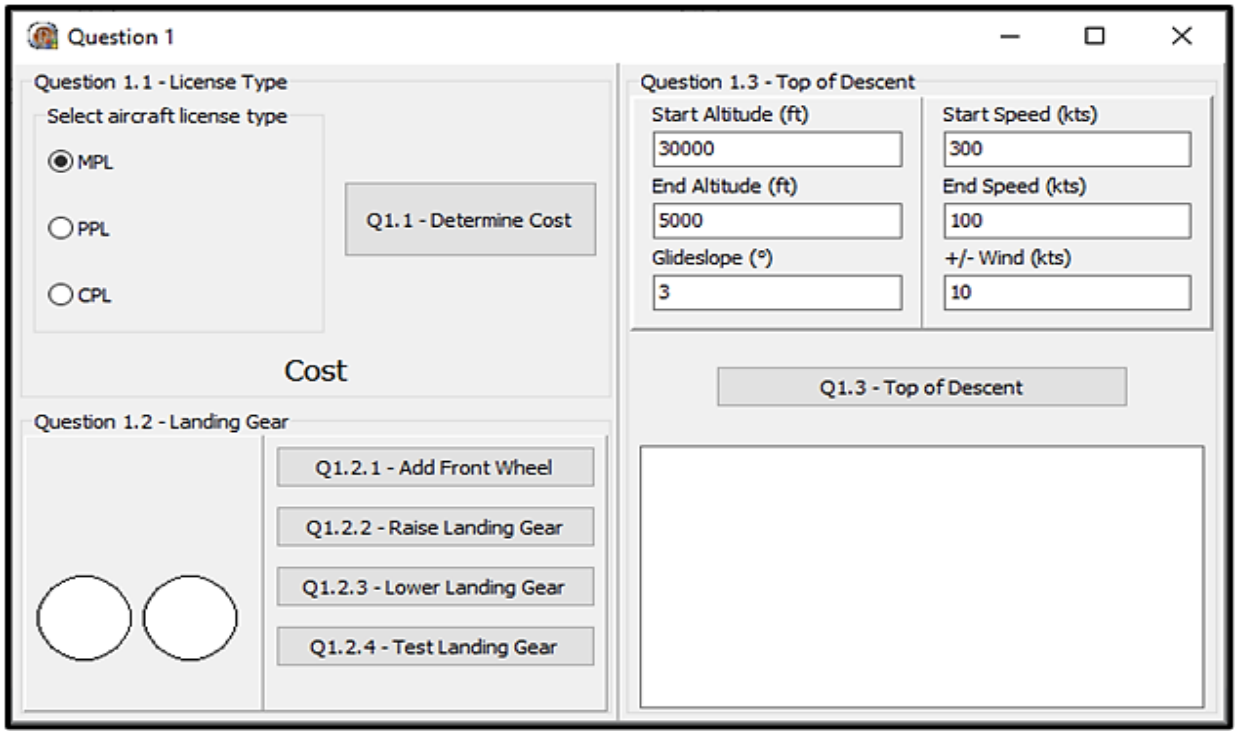
SCENARIO:

You would like to become a pilot; therefore, you need to start researching how to become a pilot and which aircraft license type you should obtain. You decide to code an application that will assist you in your research as well as studies towards your pilot license.

Do the following:

- Open the incomplete program in the **Question 1** folder.
- Enter your full name as a comment in the first line of the **Question1_u.pas** file.
- Compile and execute the program. Currently the program has no functionality.
- Follow the instructions below to complete the code for each section of QUESTION 1, as described in QUESTION 1.1, QUESTION 1.2 and QUESTION 1.3.

An example of the GUI is given below:



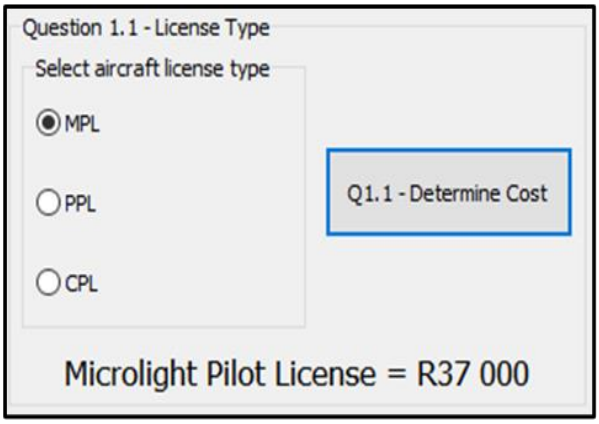
1.1 Button [Q1.1 – Determine cost]

Firstly, you need to determine which aircraft license type you can afford.
The three major types of pilot licenses are listed in the radio group **rgpQ1.1_License**.

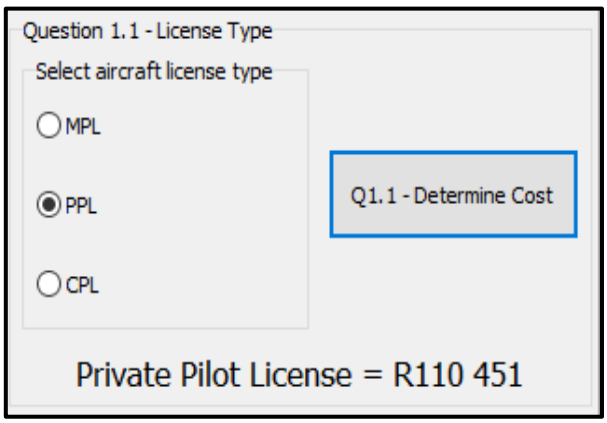
When the user clicks on the button **btnQ1_1_Cost**, it displays the information based on the license type selected in the radio group. *You may add the information using simple strings.*

Example Screenshots:

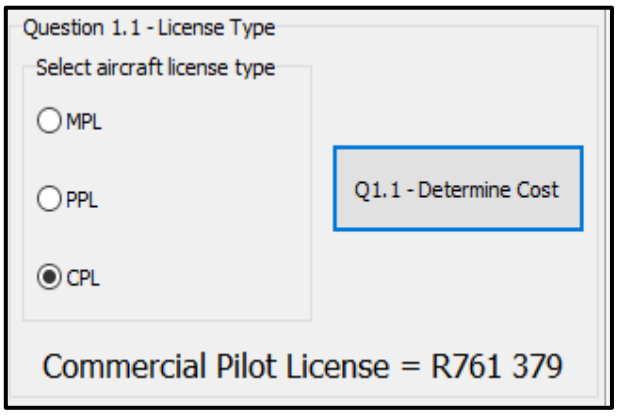
MPL



PPL



CPL



(4)

1.2 You decide to start learning to fly based on the license type PPL (Private Pilot License) using an aircraft called a Cessna 182 RG (retractable gear). On the dashboard of the Cessna 182 RG, three lights represent whether the landing gear is retracted (in the up position) or extended and locked (in the down position).

1.2.1 **Button [Q1.2.1 – Add the front wheel]**

You will notice when you run the application that the interface only displays the two rear wheels of the Cessna 182 RG and you need to add the front (nose) wheel.

Using dynamic code, add the front nose wheel with the following properties:

- Name the component: shpFrontWheel *(variable has already been declared for you)*
- Change the shape to a circle
- Set the position of shpFrontWheel to:
  - Top = 25
  - Left = 35
- Set the size of shpFrontWheel to:
  - Width = 50
  - Height = 50
- Change the colour of shpFrontWheel to clWhite

NOTE: *If you cannot complete the dynamic code for this question, add the shape without dynamic code so that you can complete QUESTIONS 1.2.2–1.2.4. Marks for the dynamic code will be forfeited.* (6)

1.2.2 **Button [Q1.2.2 – Raise landing gear]**

In the Cessna 182 RG, when you raise the landing gear, all three lights on the dashboard turn Red.

Code the button **btnQ1_2_2_Up** to change all three lights to Red. (1)

1.2.3 **Button [Q1.2.3 – Lower landing gear]**

In the Cessna 182 RG, when you lower the landing gear, all three lights on the dashboard turn Green.

Code the button **btnQ1_2_3_Down** to change all three lights to Green. (1)

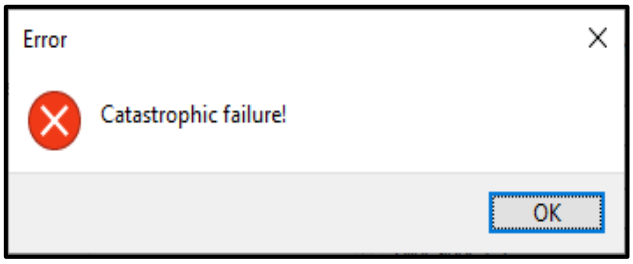
1.2.4 **Button [Q1.2.4 – Test landing gear]**

When landing any aircraft with retractable landing gear, all landing gear must be in the locked position and the lights on the dashboard must be green.

Provided code will randomise the landing gear position each time the user clicks the button.

Code the button **btnQ1_2_4 – Test** to test whether it is safe to land. Display a message using a Message Dialog box to indicate the different landing conditions. *Note the type of messages in each screenshot.*

- 3 Red Lights → Catastrophic failure!



- 1 or 2 Red Lights → Caution, not safe to land

- 3 Green Lights → Safe to land

(8)

1.3 Button [Q1.3 – Top of Descent]

In aviation, the Top of Descent is the transition from the cruising speed phase of a flight to the descent phase. This is the point at which the planned descent to final approach altitude and landing is initiated. The Top of Descent is usually calculated by an on-board flight management system. Your Cessna 182 RG does not have an on-board flight management system. Thus, you decide to create your own program to determine the Top of Descent.

3 Constant values have been created for you:

- DEG_TO_RAD (degrees to radians)
- NM_TO_FT (nautical miles to feet)
- FT_TO_NM (feet to nautical miles)

The user will enter the following information:

- Start altitude and end altitude (measured in feet)
- Start speed and end speed (measured in knots)
- Wind speed (measured in knots)
- Glideslope (measured in degrees)

Vertical speed formula:

- *Average speed = (Start speed x End speed) ÷ (2 + Wind speed)*
- *VerticalSpeed_NM_P_Min = tan(Glideslope x DEG_TO_RAD) x Average speed ÷ 60*
- *Vertical speed = VerticalSpeed_NM_P_Min x NM_TO_FT*

Distance formula:

- *Delta altitude = Starting altitude – Ending altitude*
- *Delta Speed = Starting speed – Ending speed*
- *Distance = (Delta altitude x FT_TO_NM) ÷ tan(Glideslope x DEG_TO_RAD)*
- *Distance = Distance + the ceiling function of Delta speed divided by 10*
- *Distance = Distance + the ceiling function of Wind speed divided by 10*

Estimated time formula:

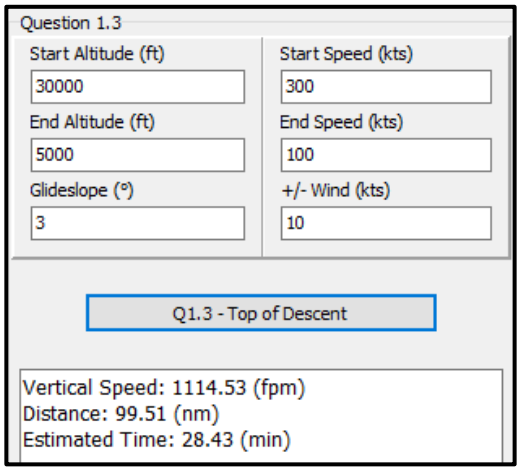
Estimated time = (Distance ÷ average speed) x 60

Using the formulas above, code the button **btnQ1_3_Top_of_Descent** to output the following information into the rich edit **redQ1_3**

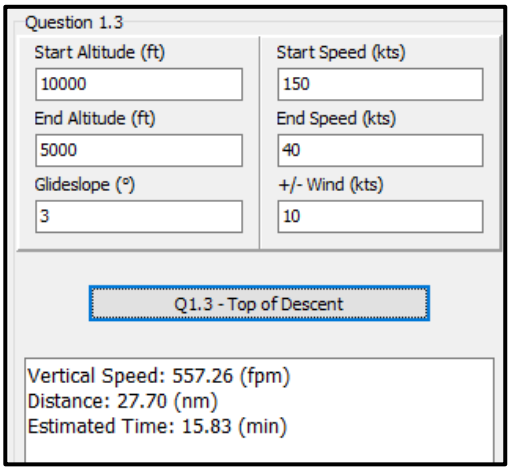
- Vertical speed of descent (measured in feet per minute)
- Distance to landing area (measured in nautical miles)
- Estimated time to arrive at landing area (measured in minutes) (20)

See example screenshots on the following page:

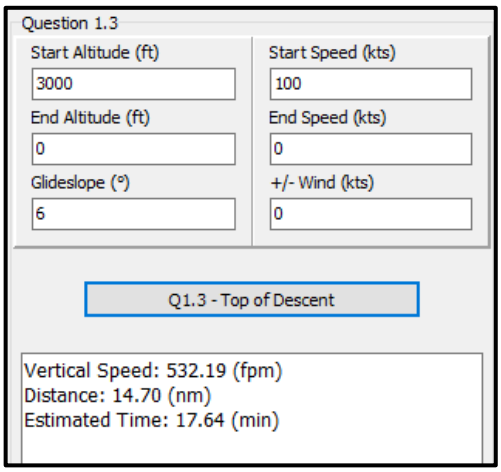
EXAMPLE 1



EXAMPLE 2



EXAMPLE 3



- Enter your name and surname as a comment in the first line of the program file.
- Save your program.
- A printout of the code may be required.

[40]

QUESTION 2: SQL AND DATABASE PROGRAMMING

The data pages attached at the end of the question paper provide information on the design of the **FlightInfo.mdb** database and its contents.

Do the following:

- Compile and execute the program in the Question 2 folder. The program currently has limited functionality.
- Enter your full name as a comment in the **Question2_u.pas.**
- Follow the instructions below to complete the code for each section as described in QUESTION 2.1 and QUESTION 2.2.
- Use SQL statements to answer QUESTION 2.1 and Delphi code to answer QUESTION 2.2.

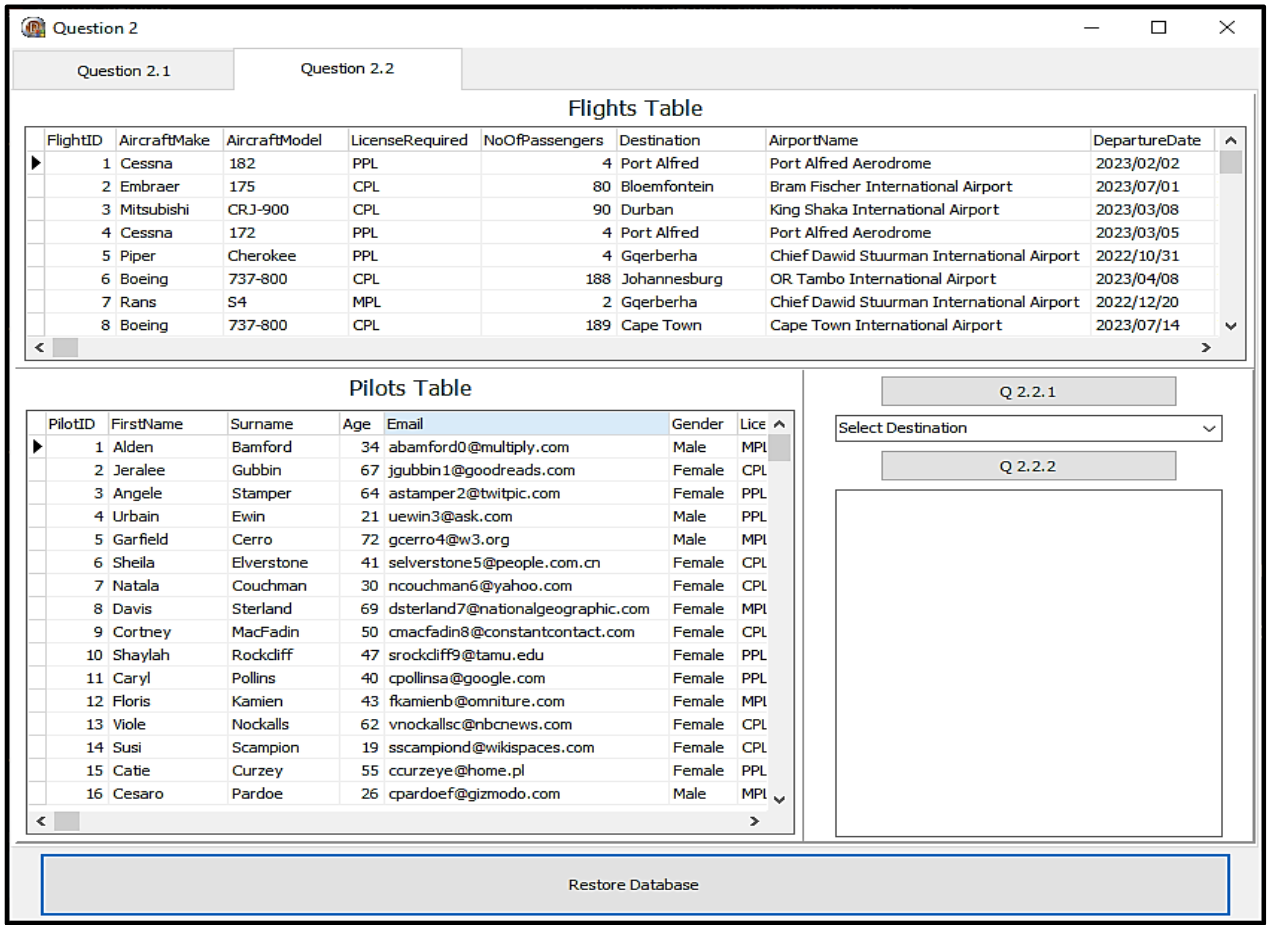
NOTE:

- The 'Restore database' button is provided to restore the data contained in the database to the original content.
- The contents of the database is password-protected, i.e. you will NOT be able to gain access to the content of the database using Microsoft Access.
- Code is provided to link the GUI components to the database. Do NOT change any of the code provided.
- THREE variables are declared as public variables. Descriptions of these variables are listed in the table below:

Variable	Data type	Description
tblFlight	TADOTable	Refers to the data stored in the table tblFlights
tblPilot	TADOTable	Refers to the data stored in the table tblPilots
qryInfo	TADOQuery	Query component that will query the two tables tblFlights and tblPilots

- Use the ADO components tblFlight and tblPilot only in **QUESTION 2.2**.

Example of graphical user interface (GUI):



2.1 In this section you may ONLY use **SQL statements** to answer QUESTION 2.1.1 to QUESTION 2.1.5.

Code to execute the **SQL statements** and display the results of the queries is provided. The SQL statements are incomplete.

NOTE*:* *If you have coded and run QUESTION 2.2, then your results of QUESTION 2.1 will be different to the results in the example output. Please ensure you click the* ***Restore Database*** *button before executing the SQL statements.*

2.1.1 **Button [Q2.1.1]**

Write SQL code to display the Name, Surname, Age, and Email of all the pilots between the ages of 36 (inclusive) and 42 (inclusive), sorted by the age from oldest to youngest.

Example of the output:

FirstName	Surname	Age	Email
Guenna	Louden	42	glouden1x@google.pl
Teador	Egre	42	tegrer@chron.com
Barclay	Silverwood	42	bsilverwoodt@google.ru
Giordano	Benne	42	gbenne1c@ucoz.com
Putnem	Ferrara	42	pferrara3u@about.com
Sibel	Potteridge	41	spotteridge33@ucsd.edu
Jaime	Rozsa	41	jrozsa2l@joomla.org
Sheila	Elverstone	41	selverstone5@people.com.cn
Kaine	Curd	40	kcurd4d@dagondesign.com
Fayette	Sirrell	40	fsirrellh@sakura.ne.jp
Caryl	Pollins	40	cpollinsa@google.com
Shandeigh	Verity	40	sverity1t@telegraph.co.uk
Danna	Ferretti	38	dferretti4c@bing.com
Arline	Persicke	37	apersickeq@prnewswire.com
Hewett	Maase	36	hmaase2m@myspace.com
Luca	Glenfield	36	lglenfield3l@mapquest.com

(4)

2.1.2 **Button [Q2.1.2]**

If the user wishes to see which flights are available to their required destination, they will enter their destination using an input box. *The input box code has been provided for you and will store the value in the variable sLine.*

Write SQL code to display all the fields of all the records that contain the destination inputted into the input box.

Example of the output if the word “Alfred” was entered:

FlightID	AircraftMake	AircraftModel	LicenseRequired	NoOfPassengers	Destination	AirportName	DepartureDate	CoPilotRequired	FlightCost	PilotID
1	Cessna	182	PPL	4	Port Alfred	Port Alfred Aerodrome	2023/02/02	False	2200	64
4	Cessna	172	PPL	4	Port Alfred	Port Alfred Aerodrome	2023/03/05	False	2200	118
13	Rans	S4	MPL	2	Port Alfred	Port Alfred Aerodrome	2023/08/21	False	1100	171
41	Piper	Cub	PPL	2	Port Alfred	Port Alfred Aerodrome	2023/04/05	False	1100	171

(4)

2.1.3 **Button [Q2.1.3]**

You would like to know how many aircrafts are flying in the month of September.

Write SQL code to display the number of flights in September. Name the calculated field “Flights in September”.

Example of the output:

Flights in September
6

(3)

2.1.4 **Button [2.1.4]**

The airlines would like to see the cost breakdown of all the flights based on the destination.

Write a SQL statement to display the Destination, Pilot Cost, Flight Cost, and Total Cost of all the flights based on the destination.

Example of the output:

Destination	Pilot Cost	Flight Cost	Total Cost
Bloemfontein	R10 103.00	R952 128.00	R962 231.00
Cape St Francis	R11 517.00	R4 400.00	R15 917.00
Cape Town	R15 330.00	R1 766 844.00	R1 782 174.00
Durban	R11 079.00	R650 016.00	R661 095.00
East London	R17 928.00	R335 820.00	R353 748.00
George	R1 740.00	R11 900.00	R13 640.00
Gqerberha	R9 667.00	R6 600.00	R16 267.00
Graaff Reinet	R1 815.00	R3 300.00	R5 115.00
Johannesburg	R10 662.00	R3 211 380.00	R3 222 042.00
King Williams Town	R2 331.00	R4 400.00	R6 731.00
Nelspruit	R6 844.00	R228 888.00	R235 732.00
Port Alfred	R9 818.00	R6 600.00	R16 418.00

(12)

2.1.5 Button [Q2.1.5]

All the CPL (Commercial Pilot License) certified pilots have been given a 7% raise on their Cost Per Flight.

Write a SQL statement to increase only the CPL certified pilot's salaries by 7%.

NOTE: If you click on Button 2.1.4 after Button 2.1.5, the following output will be displayed. Only George, Graaff-Reinet, and King William's Town had no change in pilot cost. This is because there are no CPL certified pilots in those towns.

Destination	Pilot Cost	Flight Cost	Total Cost
Bloemfontein	R10 583.13	R952 128.00	R962 711.13
Cape St Francis	R12 102.55	R4 400.00	R16 502.55
Cape Town	R16 326.45	R1 766 844.00	R1 783 170.45
Durban	R11 746.38	R650 016.00	R661 762.38
East London	R18 864.95	R335 820.00	R354 684.95
George	R1 740.00	R11 900.00	R13 640.00
Gqerberha	R10 175.90	R6 600.00	R16 775.90
Graaff Reinet	R1 815.00	R3 300.00	R5 115.00
Johannesburg	R11 228.79	R3 211 380.00	R3 222 608.79
King Williams Town	R2 331.00	R4 400.00	R6 731.00
Nelspruit	R7 241.11	R228 888.00	R236 129.11
Port Alfred	R10 404.04	R6 600.00	R17 004.04

(3)

2.2 In this section, only Delphi programming code may be used to answer QUESTION 2.2.1 and QUESTION 2.2.2.

Use the global variables, tblFlight and tblPilot, provided.

NO marks will be awarded for SQL statements in QUESTION 2.2.

2.2.1 **Button [Q2.2.1]**

A government regulation has been passed that a pilot can only be certified for a license if they are over the age of 21.

Write code to delete all the pilots who are under the age of 21. Output to the rich edit redQ2_Output as per example screenshot.

Example of the output:

```
Pilots before regulation change: 200
Pilots after regulation change: 193
```

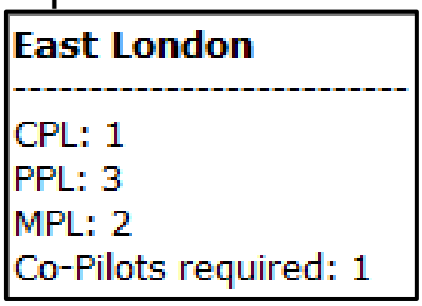
(3)

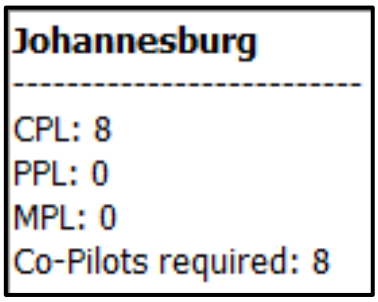
2.2.2 **Button [Q2.2.2]**

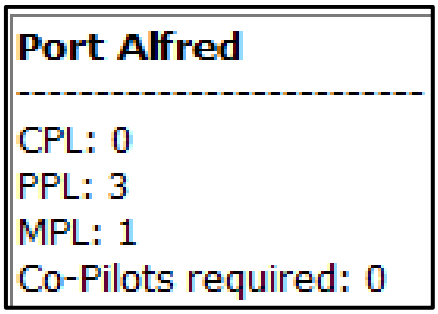
A breakdown of the different aircraft types (license requirement) needs to be displayed. To simplify output, the breakdown will be based on the destination airport.

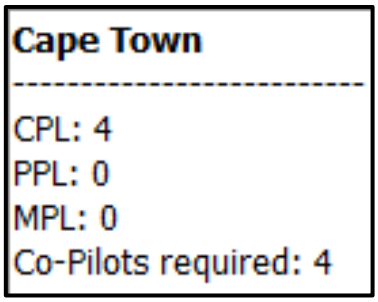
Write code to search, based on the destination selected in the combo box cmbQ2_2_2_Destination. Calculate and display how many different aircraft are there per license type. A list of how many co-pilots are required per the selected destination must also be displayed. Output to the rich edit redQ2_Output as per example screenshots.

Example screenshots:









(6)

- Enter your name and surname as a comment in the first line of the program file.
- Save your program.
- A printout of the code may be required.

[35]

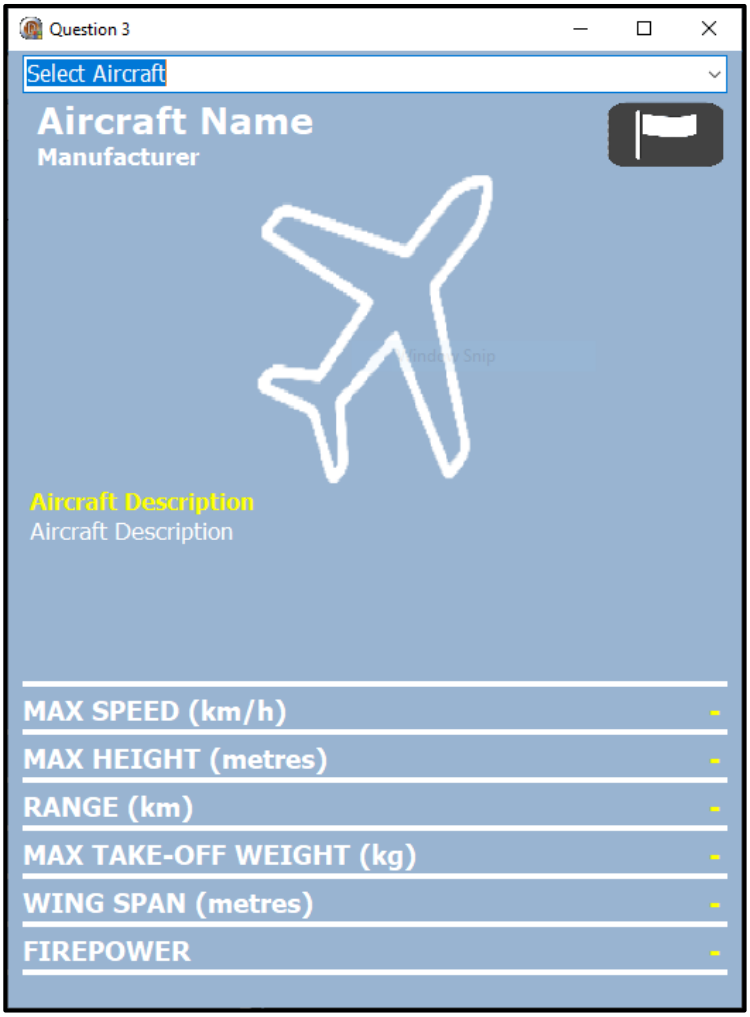
QUESTION 3: OBJECT-ORIENTED PROGRAMMING

Aircraft Flashcards are an excellent tool for learning about different types of aircraft. Inspired by this idea, you have decided to develop an application that will digitally replicate these flashcards. By loading various aircraft onto the digital platform, yourself and other users can not only gain knowledge about different aircraft, but also engage in interactive flashcard games with each other.

Do the following:

- Compile and execute the program in the QUESTION 3 folder. The program currently has limited functionality.
- Enter your full name as a comment in both the **Question3ClassDefinition_u.pas** and the **Question3_u.pas** files.
- Do NOT remove or change any provided code.

The following user interface is displayed:



Follow the instructions below to complete the code for each section of QUESTION 3.1 and QUESTION 3.2.

NOTE: For this question, you are NOT allowed to include any additional attributes or user-defined methods unless explicitly stated in the question.

Open the incomplete object class **Question3ClassDefinition_u.pas.**

3.1 The incomplete class (TAircraft) contains the declaration of attributes that describe the objAircraft object.

NAME OF ATTRIBUTES	DESCRIPTION
fName	String value containing Aircraft name
fManufacturer	String value containing Manufacturer Name
fSpeed	Real value containing Speed in Knots
fHeight	Real value containing Height in Feet
fRange	Real value containing Range in Miles
fWeight	Real value containing Weight in Pounds
fWingspan	Real value containing Wingspan in Feet
fFirepower	Integer value containing Range value between 1 – 10
fImageName	String value containing Image name
fCountry	String value containing Country Image name
fDescription	Wide string containing the Aircraft description

3.1.1 Write code for a **constructor** method named **Create** that will receive the following parameters:

Name

Manufacturer
Speed
Height
Range
Weight
Wingspan
Firepower
ImageName
Country
Description

NOTE: These data items will be read from the Aircraft_List.csv text file in the **Question3_u.pas** unit and sent through to this constructor.

Assign these parameter values to the correct attributes. (5)

3.1.2 The attributes (Speed, Height, Range, Weight, and Wingspan) need to be converted to a South African measurement system. Write code for a **mutator** method named setValues to convert the current attributes to their respective South African measurement values. The conversions are as follows:

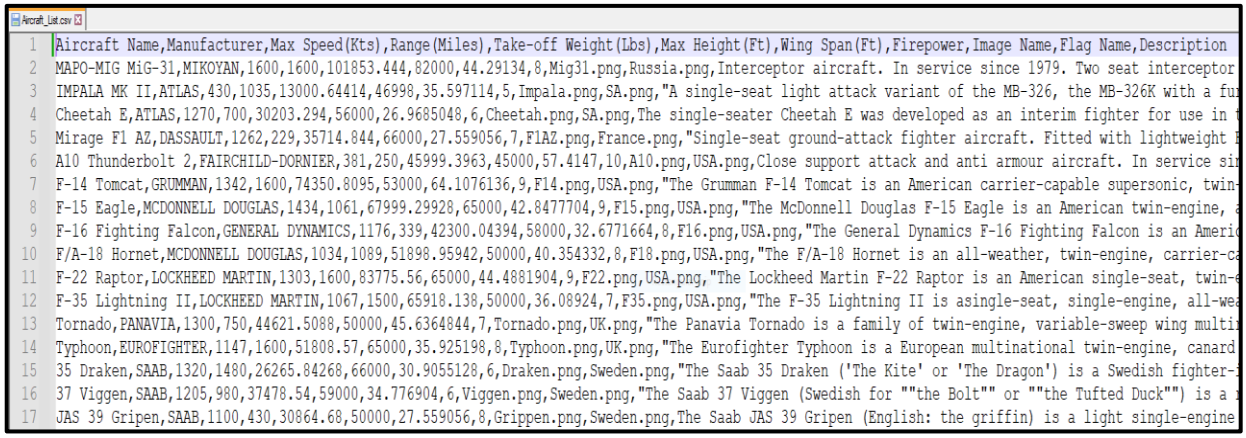
Knots to Kilometres per hour = 1.852
Feet to Meters = 0.3048
Miles to Kilometres = 1.60934
Pounds to Kilograms = 0.45359 (5)

3.2 An incomplete unit **Question3_u.pas** has been provided.

It contains code for the object class to be accessible and a global object variable, **objAircraft**, already declared.

Do NOT delete or change any provided code.

The text file, Aircraft_List.csv, contains the details of 16 aircrafts shown below:


Aircraft_List.csv

```
Aircraft Name,Manufacturer,Max Speed(Kts),Range(Miles),Take-off Weight(Lbs),Max Height(Ft),Wing Span(Ft),Firepower,Image Name,Flag Name,Description
MAPO-MIG MiG-31,MIKOYAN,1600,1600,101853.444,82000,44.29134,8,Mig31.png,Russia.png,Interceptor aircraft. In service since 1979. Two seat interceptor
IMPALA MK II,ATLAS,430,1035,13000.64414,46998,35.597114,5,Impala.png,SA.png,"A single-seat light attack variant of the MB-326, the MB-326K with a fu
Cheetah E,ATLAS,1270,700,30203.294,56000,26.9685048,6,Cheetah.png,SA.png,The single-seater Cheetah E was developed as an interim fighter for use in t
Mirage F1 AZ,DASSAULT,1262,229,35714.844,66000,27.559056,7,F1AZ.png,France.png,"Single-seat ground-attack fighter aircraft. Fitted with lightweight
A10 Thunderbolt 2,FAIRCHILD-DORNIER,381,250,45999.3963,45000,57.4147,10,A10.png,USA.png,Close support attack and anti armour aircraft. In service si
F-14 Tomcat,GRUMMAN,1342,1600,74350.8095,53000,64.1076136,9,F14.png,USA.png,"The Grumman F-14 Tomcat is an American carrier-capable supersonic, twin-
F-15 Eagle,MCDONNELL DOUGLAS,1434,1061,67999.29928,65000,42.8477704,9,F15.png,USA.png,"The McDonnell Douglas F-15 Eagle is an American twin-engine,
F-16 Fighting Falcon,GENERAL DYNAMICS,1176,339,42300.04394,58000,32.6771664,8,F16.png,USA.png,"The General Dynamics F-16 Fighting Falcon is an Americ
F/A-18 Hornet,MCDONNELL DOUGLAS,1034,1089,51898.95942,50000,40.354332,8,F18.png,USA.png,"The F/A-18 Hornet is an all-weather, twin-engine, carrier-ca
F-22 Raptor,LOCKHEED MARTIN,1303,1600,83775.56,65000,44.4881904,9,F22.png,USA.png,"The Lockheed Martin F-22 Raptor is an American single-seat, twin-e
F-35 Lightning II,LOCKHEED MARTIN,1067,1500,65918.138,50000,36.08924,7,F35.png,USA.png,"The F-35 Lightning II is asingle-seat, single-engine, all-wea
Tornado,PANAVIA,1300,750,44621.5088,50000,45.6364844,7,Tornado.png,UK.png,"The Panavia Tornado is a family of twin-engine, variable-sweep wing multi
Typhoon,EUROFIGHTER,1147,1600,51808.57,65000,35.925198,8,Typhoon.png,UK.png,"The Eurofighter Typhoon is a European multinational twin-engine, canard
35 Draken,SAAB,1320,1480,26265.84268,66000,30.9055128,6,Draken.png,Sweden.png,"The Saab 35 Draken ('The Kite' or 'The Dragon') is a Swedish fighter-
37 Viggen,SAAB,1205,980,37478.54,59000,34.776904,6,Viggen.png,Sweden.png,"The Saab 37 Viggen (Swedish for ""the Bolt"" or ""the Tufted Duck"") is a
JAS 39 Gripen,SAAB,1100,430,30864.68,50000,27.559056,8,Grippen.png,Sweden.png,The Saab JAS 39 Gripen (English: the griffin) is a light single-engine
```

NOTE: The first line of the text file contains the headings of the different categories of the aircraft.

The images for the flags can be found in the Folder: Images\Flags\
The images for the aircraft can be found in the Folder: Images\Aircraft\

Follow the instructions below to code the solution.

3.2.1 Combobox [Q3.2.1 – cmbQ3_SelectAircraft]

When the program executes, the user will see a blank flashcard. The user will choose which aircraft they would like to display by selecting the different aircraft from the combo box.

Code the **OnChange** event handler of the cmbQ3_SelectAircraft combo box to do the following:

1. Extract the user's selection from the combo box. (1)

2. Test to see if the text file exists and assign the file. If the file does not exist, display a suitable message and exit. (5)

3. Loop through the text file until the user's selected aircraft has been found. (6)

4. If the aircraft has been found:

   - Write code **in the loop** to extract the information from the text file and store it in the already declared local variables. (5)
   - Using the extracted values in the variables, instantiate (create) the object **objAircraft**. (3)

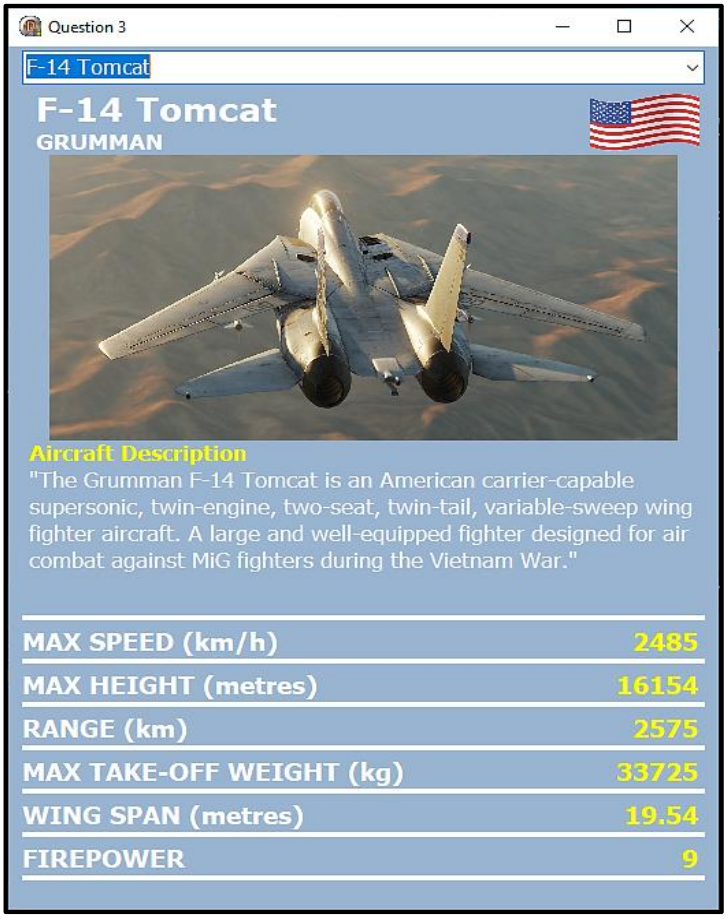
   NOTE: Only one aircraft will be loaded into the object. This is based on the user's selection from the combo box.

5. Once the aircraft has been found and the loop has exited, use the object **objAircraft** to call the setValues method. (1)

6. Using the object **objAircraft,** load the data into the following components:

lblQ3_AircraftName
lblQ3_Manufacturer
imgQ3_CountryFlag
imgQ3_AircraftImage
lblQ3_AircraftDescription
lblQ3_MaxSpeed
lblQ3_MaxHeight
lblQ3_Range
lblQ3_MaxTakeoffWeight
lblQ3_Wingspan
lblQ3_Firepower (4)

Example: Screenshot of the flashcard when an aircraft has been successfully found, extracted from the text file, instantiated into the object and then object's data values loaded into the flashcard.



- Enter your name and surname as a comment in the first line of the program file.
- Save your program.
- A printout of the code may be required.

[35]

QUESTION 4: PROBLEM-SOLVING PROGRAMMING

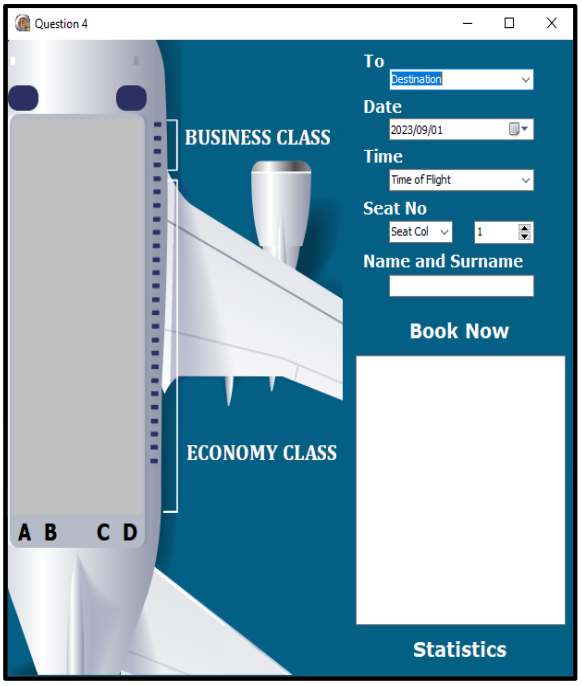
One of the local airline companies has asked you to assist them with creating a booking application for their new aircraft, the Mitsubishi SpaceJet M100. The SpaceJet M100's cabin is configured with 15 rows of two seats on the left and two seats on the right of the aisle. The first two rows of the cabin are assigned to business class, and the remaining rows are assigned to economy class.

Do the following:

- Compile and execute the program in the Question 4 folder. The program currently has limited functionality.
- Enter your full name as a comment in the first line of the **Question4_u.pas** file.
- Do NOT remove or change any provided code.
- Assume the user will enter the correct information into the input components. You will not be required to enter any data validation code on the input components.

The following user interface is displayed:

If the image on your GUI does not line up like the image in the GUI screenshot, then set Windows Scaling to 100%.



Aircraft image source: Skypark Secure

https://www.skyparksecure.com/blog/fastest-plane-boarding-methods/

The user interface makes use of a String Grid component to display the availability of seats on the aircraft. All the data indicating whether a seat has been booked or not, is stored in a 2D array, named ar2Booking. This 2D array will be used to update the String Grid component and change the seat colour.

NOTE: All String Grid code has been provided for you. Do not delete or alter the String Grid code. You will NOT need to write any String Grid code.

Breakdown of the data:

Character code stored in the 2D array: ar2Booking	Availability	String Grid Cell Colour
'B'	Booked seat	Blue
'A'	Available seat	White
'I'	Aisle	Gray

Follow the instructions below to complete the code for each section of QUESTION 4, as described in QUESTION 4.1, QUESTION 4.2, and QUESTION 4.3.

4.1 The airline allows passengers to book flights from Gqeberha. The information regarding to the availability of seats to the various destinations is stored in the following globally declared 2D arrays. *This code has been done for you and must not be altered or removed:*

- Bloemfontein ar2Bloemfontein
- Cape Town ar2CapeTown
- Durban ar2Durban
- East London ar2EastLondon
- Johannesburg ar2Johannesburg

Write code on the **onChange** event handler of cmbQ4_1_Destination to do the following:

1. Extract the user's destination from the combo box. (1)
2. Transfer the data from the destination 2D array to the globally declared 2D array, ar2Booking. (5)

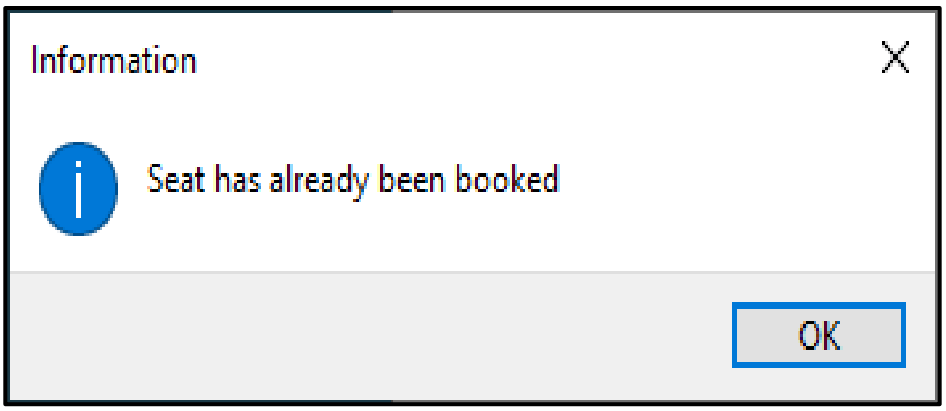
4.2 The application must provide the user the ability to book a seat. After they have selected their destination, the user will fill in their details into the remaining input components.

Write code for the **onClick** event handler of the panel named pnlQ4_2_Book (the panel acts like a button) to do the following:

1. Extract the seat number and loop through ar2Booking to determine if the seat has been booked or not.
   'A' = Available
   'B' = Booked
   'I' = Aisle

NOTE: That there are only four seat columns A..D, however, ar2Booking has 5 columns of which the 3rd column stores the aisle. Write your code to ensure that columns A..D match with the correct columns in ar2Booking.

If the seat has been booked, display a message that the seat has already been booked.



(5)

2. If the seat has not been booked, book the seat in ar2Booking and extract all the data from the various input components.

   The prices for the various flights are stored in a global array named arrPrices. Each destination has a different price based on the destination. Business class passengers pay 95% more for their flights than economy class passengers.

   Output the seat ticket information into the rich edit redQ4_Output as per example screenshots.

(12)

4.3 The airline would like to see statistics of the various flights.

Write code for the **onClick** event handler of the panel named pnlQ4_3_Stats (the panel acts like a button) to do the following:

1. Extract the destination.

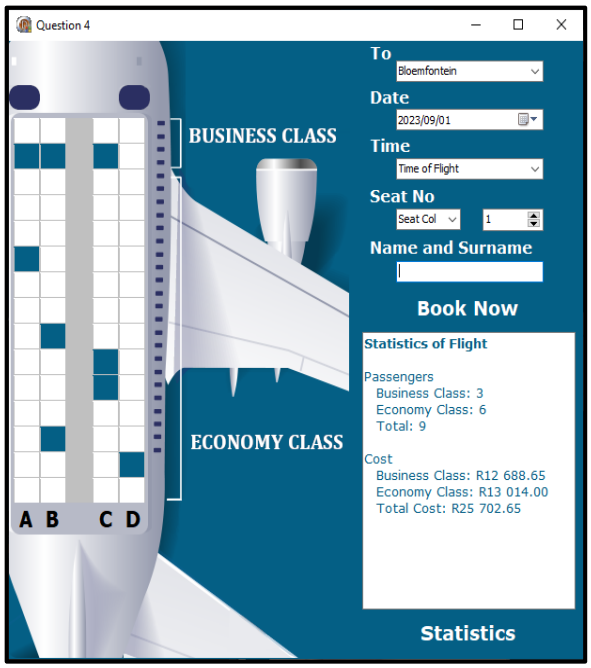
2. Loop through ar2Booking to determine how many business class and economy class passengers are there.

3. Determine the cost for business class as well as economy class. Business class passengers pay 95% more than economy class passengers. Prices are stored in the global array, arrPrices.

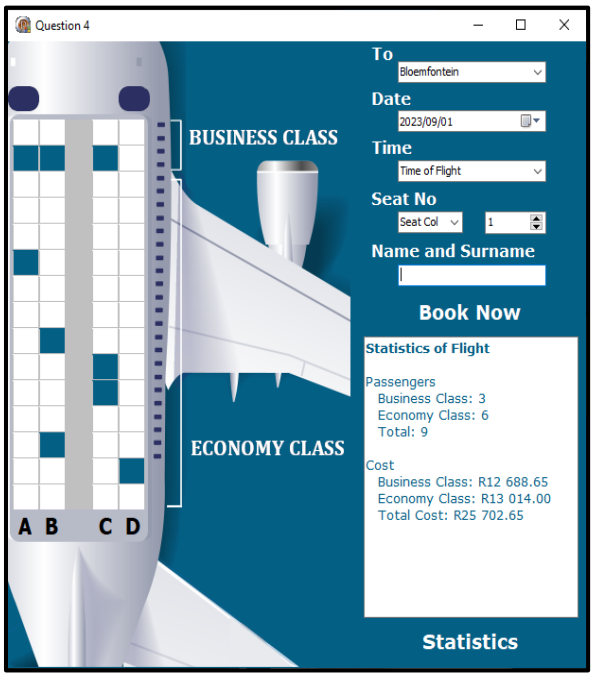
4. Output the statistics of the flight into the rich edit redQ4_Output as per example screenshots on the following page.

These screenshots are of the different destinations without extra seats booked.

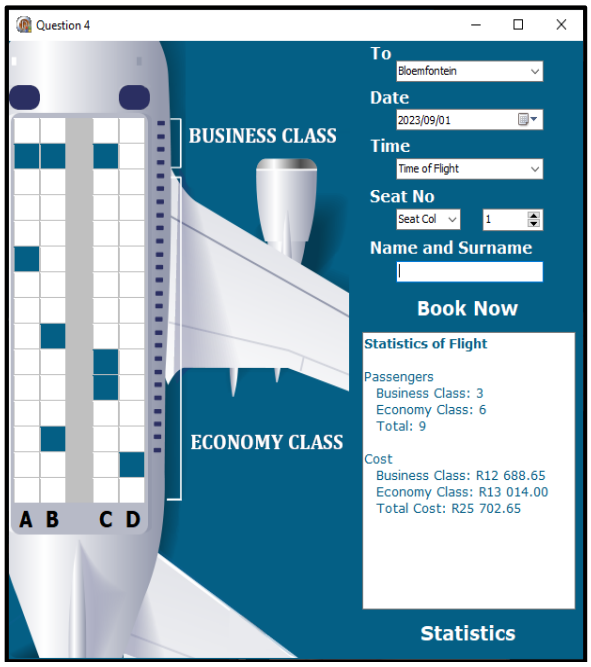
Bloemfontein



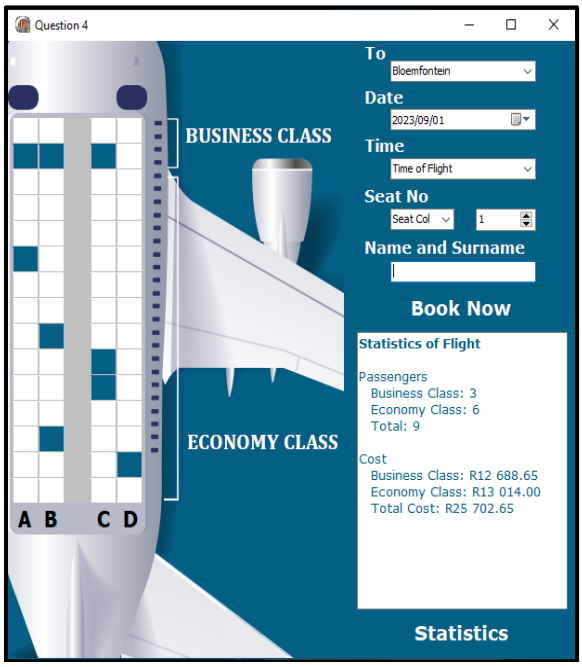
Cape Town



Durban



Johannesburg



- Enter your name and surname as a comment in the first line of the program file.
- Save your program.
- A printout of the code may be required.

[40]

GRAND TOTAL: 150

DATABASE INFORMATION QUESTION 2:

The design of the database tables is as follows:

Table: **tblFlights**

The table contains the information of the various flights.

Field Name	Data Type
FlightID	AutoNumber
AircraftMake	Short Text
AircraftModel	Short Text
LicenseRequired	Short Text
NoOfPassengers	Number
Destination	Short Text
AirportName	Short Text
DepartureDate	Date/Time
CoPilotRequired	Yes/No
FlightCost	Currency
PilotID	Number

Example of the first twenty records in the **tblFlights** table:

FlightID	AircraftMak	AircraftMod	LicenseReq	NoOfPasser	Destination	AirportNam	DepartureD	CoPilotReq	FlightCost	PilotID
1	Cessna	182	PPL	4	Port Alfred	Port Alfred Ae	2023/02/02	False	R2 200.00	64
2	Embraer	175	CPL	80	Bloemfontein	Bram Fischer I	2023/07/01	True	R175 680.00	44
3	Mitsubishi	CRJ-900	CPL	90	Durban	King Shaka Int	2023/03/08	True	R197 640.00	118
4	Cessna	172	PPL	4	Port Alfred	Port Alfred Ae	2023/03/05	False	R2 200.00	118
5	Piper	Cherokee	PPL	4	Gqerberha	Chief Dawid St	2022/10/31	False	R2 200.00	195
6	Boeing	737-800	CPL	188	Johannesburg	OR Tambo Inte	2023/04/08	True	R412 848.00	148
7	Rans	S4	MPL	2	Gqerberha	Chief Dawid St	2022/12/20	False	R1 100.00	41
8	Boeing	737-800	CPL	189	Cape Town	Cape Town Int	2023/07/14	True	R415 044.00	7
9	Piper	Cherokee	PPL	4	King Williams	Bhisho Airport	2022/12/19	False	R2 200.00	98
10	Rans	S6	MPL	2	East London	King Phalo Airp	2023/09/15	False	R1 100.00	94
11	Cessna	210	PPL	6	Nelspruit	Kruger Mpuma	2023/05/20	False	R10 800.00	34
12	Piper	Chieftain	PPL	7	Nelspruit	Kruger Mpuma	2023/09/04	False	R12 600.00	59
13	Rans	S4	MPL	2	Port Alfred	Port Alfred Ae	2023/08/21	False	R1 100.00	171
14	Beechcraft	1900D	CPL	19	Cape Town	Cape Town Int	2022/11/20	True	R34 200.00	131
15	Piper	Cub	PPL	2	Cape St Francis	St Francis Airfi	2023/04/20	False	R1 100.00	77
16	Stream	TL-Ultralight	MPL	2	Cape St Francis	St Francis Airfi	2022/12/11	False	R1 100.00	113
17	Beechcraft	1900D	CPL	19	Nelspruit	Kruger Mpuma	2022/11/30	True	R34 200.00	105
18	Piper	Cub	PPL	2	Cape St Francis	St Francis Airfi	2023/05/21	False	R1 100.00	144
19	Bombardier	CRJ-700	CPL	78	Nelspruit	Kruger Mpuma	2023/05/07	True	R171 288.00	139
20	Boeing	777-300	CPL	360	Cape Town	Cape Town Int	2023/03/20	True	R790 560.00	136

Table: **tblPilots**

This table contains the information of the pilots.

Field Name	Data Type
PilotID	AutoNumber
FirstName	Short Text
Surname	Short Text
Age	Number
Email	Short Text
Gender	Short Text
LicenseType	Short Text
PilotCostPerFlight	Currency

Example of the first twenty records in the **tblPilots** table:

PilotID	FirstName	Surname	Age	Email	Gender	LicenseType	PilotCostPe
1	Alden	Bamford	34	abamford0@m	Male	MPL	R376.00
2	Jeralee	Gubbin	67	jgubbin1@goo	Female	CPL	R2 943.00
3	Angele	Stamper	64	astamper2@tv	Female	PPL	R1 437.00
4	Urbain	Ewin	21	uewin3@ask.c	Male	PPL	R569.00
5	Garfield	Cerro	72	gcerro4@w3.o	Male	MPL	R416.00
6	Sheila	Elverstone	41	selverstone5@	Female	CPL	R9 679.00
7	Natala	Couchman	30	ncouchman6@	Female	CPL	R5 870.00
8	Davis	Sterland	69	dsterland7@ni	Female	MPL	R345.00
9	Cortney	MacFadin	50	cmacfadin8@c	Female	CPL	R9 978.00
10	Shaylah	Rockcliff	47	srockcliff9@ta	Female	PPL	R846.00
11	Caryl	Pollins	40	cpollinsa@goc	Female	PPL	R1 412.00
12	Floris	Kamien	43	fkamienb@om	Female	MPL	R314.00
13	Viole	Nockalls	62	vnockallsc@nt	Female	CPL	R3 840.00
14	Susi	Scampion	19	sscampiond@v	Female	CPL	R3 588.00
15	Catie	Curzey	55	ccurzeye@hon	Female	PPL	R921.00
16	Cesaro	Pardoe	26	cpardoef@gizr	Male	MPL	R424.00
17	Arney	Ilyukhov	46	ailyukhovg@b	Male	MPL	R317.00
18	Fayette	Sirrell	40	fsirrellh@saku	Female	PPL	R802.00
19	Joleen	Andover	29	jandoveri@vis	Female	PPL	R1 096.00
20	Ariana	Ledgerton	62	aledgertonj@r	Female	CPL	R3 619.00

NOTE: Connection code has been provided.

The following one-to-many relationship exists between the two tables in database: