



NASIONALE SENIORSERTIFIKAAT

GRAAD 12

SEPTEMBER 2023

INLIGTINGSTEGNOLOGIE V1

PUNTE: 150

TYD: 3 uur

Hierdie vraestel bestaan uit 24 bladsye.

INSTRUKSIES EN INLIGTING

1. Hierdie vraestel bestaan uit VIER vrae. Kandidate moet AL die vrae beantwoord.
2. Die duur van hierdie eksamen is drie uur. As gevolg van die aard van hierdie eksamen is dit belangrik om daarop te let dat jy nie toegelaat sal word om die eksamenlokaal voor die einde van die eksamensessie te verlaat nie.
3. Hierdie vraestel is opgestel met programmeringsterme wat kenmerkend vir die Delphi-programmeringstaal is.
4. Maak seker dat jy die vrae beantwoord volgens die spesifikasies wat in elke vraag gegee word. Punte sal volgens die vasgestelde vereistes toegeken word.
5. Beantwoord slegs wat in elke vraag gevra word. Byvoorbeeld, indien die vraag nie vir datavalidering vra nie, sal geen punte vir datavalidering toegeken word nie.
6. Jou programme moet op só 'n manier gekodeer word dat dit met enige data sal werk en nie net met die voorbeelddata wat verskaf is of enige data-uittreksels wat in die vraestel verskyn nie.
7. Roetines, soos soek, sorteer en seleksie, moet vanuit eerste beginsels ontwikkel word. Jy mag NIE die ingeboude funksies van Delphi vir enige van hierdie roetines gebruik NIE.
8. Alle datastrukture moet deur jou, die programmeerder, verklaar word, tensy die datastrukture verskaf word.
9. Jy moet jou werk gereeld stoor op die disket/CD/DVD/geheuestokkie wat aan jou gegee is, of op die skyfspasie wat vir hierdie eksamensessie aan jou toegeken is.
10. Maak seker dat jou naam as kommentaar verskyn in elke program wat jy kodeer, asook op elke gebeurtenis wat aangedui word.
11. Indien dit vereis word, druk die programmeringskode van al die programme/klasse wat jy voltooi het. Jy sal ná die eksamensessie 'n halfuur tyd vir drukwerk gegee word.
12. Aan die einde van hierdie eksamensessie, moet jy 'n disket/CD/DVD/geheuestokkie inlewer met al jou werk daarop gestoor OF jy moet seker maak dat al jou werk op die skyfspasie gestoor is wat vir hierdie eksamensessie aan jou toegeken is. Maak seker dat al die lêers gelees kan word.
13. Stoor jou werk gereeld.

14. Die lêers wat jy benodig om hierdie vraestel te voltooi, is aan jou gegee op 'n disket/CD/DVD/geheuestokkie of op die skyfspasie wat aan jou toegeken is. Die lêers word in die vorm van wagwoordbeskermde uitvoerbare lêers verskaf.

Doen die volgende:

- Dubbelklik op die uitvoerbare lêer wat met 'n wagwoord beskerm is.
- Klik op die 'Extract'-knoppie.
- Sleutel die volgende wagwoord in: **eMSm&tMR@23**

Nadat dit onttrek ('extracted') is, sal die volgende lys lêers in die lêergids ('folder') **DataAFRSept2023** beskikbaar wees:

Vraag 1:

Vraag1_p.dpr
Vraag1_p.res
Vraag1_u.dfm
Vraag1_u.pas

Vraag 2:

VlugInligting.mdb
VlugInligtingRugsteun.mdb
dbConnection_u.pas
Vraag2_p.dpr
Vraag2_p.res
Vraag2_u.dfm
Vraag2_u.pas

Vraag 3:

Lêergids: Images\Vlae – (7 Vlae)
Lêergids: Images\Vliegtuig – (17 Vliegtuie)

Vliegtuig_Lys.csv
Vraag3_p.dpr
Vraag3_p.res
Vraag3_u.dfm
Vraag3_u.pas
Vraag3KlasDefinisie.pas

Vraag 4:

Lêergids: Destination Textfiles – (5 tekslêers)

Lêergids.png
Vraag4_p.dpr
Vraag4_p.res
Vraag4_u.dfm
Vraag4_u.pas

VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE**SCENARIO:**

Jy wil graag 'n vlieënier word; daarom moet jy begin navorsing doen oor hoe om 'n vlieënier te word en watter tipe vliegtuiglisensie jy moet kry. Jy besluit om 'n program te kodeer wat jou sal help in jou navorsing sowel as studies vir jou vlieënierlisensie.

Doen die volgende:

- Maak die onvolledige program in die **Vraag 1**-lêergids oop.
- Sleutel jou naam as kommentaar in die eerste reël van die **Vraag1_u.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.
- Volg die instruksies hieronder om die kode vir elke afdeling van VRAAG 1, soos beskryf in VRAAG 1.1, VRAAG 1.2 en VRAAG 1.3, te voltooi.

'n Voorbeeld van die grafiese gebruikerskoppelvlak (GGK ('GUI')) word hieronder gegee:

The screenshot displays a graphical user interface (GUI) for a program titled "Vraag 1". The interface is divided into three main sections:

- Vraag 1.1 - Lisensietipe**: This section contains a label "Kies vliegtuiglisensietipe" and three radio buttons labeled "MVL", "PVL", and "KVL". The "MVL" button is selected. Below the radio buttons is a button labeled "V1.1 - Bepaal Koste".
- Vraag 1.2 - Landingsrat**: This section features two circular diagrams representing landing gear. To the right of the diagrams are four buttons labeled "V1.2.1 - Voeg Voorste Wiel By", "V1.2.2 - Lig Landingsrat Op", "V1.2.3 - Sit Landingsrat Neer", and "V1.2.4 - Toets Landingsrat".
- Vraag 1.3 - Hoogtepunt van Daling**: This section contains two columns of input fields. The left column has fields for "Beginhoogte (voet)" (value: 30000), "Eindhoogte (voet)" (value: 5000), and "Helling (°)" (value: 3). The right column has fields for "Beginspoed (knope)" (value: 300), "Eindspoed (knope)" (value: 100), and "+/- Wind (knope)" (value: 10). Below these fields is a button labeled "V1.3 Hoogtepunt van Daling".

1.1 Knoppie [V1.1 – Bepaal koste]

Eerstens moet jy bepaal watter tipe vliegtuiglisensie jy kan bekostig. Die drie hooftipes vlieënierlisensies word in die 'radio group' **rgpQ1_1_Lisensie** gegee.

Wanneer die gebruiker op die **btnV1_1_Koste** knoppie klik, moet die inligting gebaseer op die lisensietipe wat in die 'radio group' gekies is, vertoon word. Jy mag die inligting as eenvoudige stringe bymekaar voeg.

Voorbeeld van skermkote/afvoer:

MVL

Vraag 1.1 - Lisensietipe

Kies vliegtuiglisensietipe

☒ MVL

☐ PVL

☐ KVL

V1.1 - Bepaal Koste

Mikroligte Vlieënier Lisensie = R37 000

PVL

Vraag 1.1 - Lisensietipe

Kies vliegtuiglisensietipe

☐ MVL

☒ PVL

☐ KVL

V1.1 - Bepaal Koste

Privaat Vlieënier Lisensie = R110 451

KVL

Vraag 1.1 - Lisensietipe

Kies vliegtuiglisensietipe

☐ MVL

☐ PVL

☒ KVL

V1.1 - Bepaal Koste

Kommersiele Vlieënier Lisensie = R761 379

(4)

- 1.2 Jy besluit om te begin leer vlieg met die lisensietipe PVL (Privaat Vlieënier Lisensie) met 'n vliegtuig genaamd 'n Cessna 182-RG (intrekbare rat). Op die paneelbord van die Cessna 182-RG is daar drie ligte wat aandui of die landingsrat ingetrek is (in die op-posisie) of verleng en gesluit (in die af-posisie) is.

1.2.1 Knoppie [V1.2.1 – Voeg voorste wiel by]

Jy sal sien wanneer jy die toepassing uitgevoer word dat die koppelvlak net die agterste twee wiele van die Cessna 182-RG vertoon en jy moet die voorste (neus) wiel byvoeg.

Gebruik dinamiese kode om die voorste neuswiel met die volgende eienskappe by te voeg:

- Noem die komponent: `shpVoorsteWiel` (*veranderlike is alreeds vir jou verklaar*)
- Verander die vorm na 'n sirkel
- Stel die posisie van `shpVoorsteWiel` na:
 - `Top = 25`
 - `Left = 35`
- Stel die grootte van `shpVoorsteWiel` na:
 - `Width = 50`
 - `Height = 50`
- Verander die kleur van `shpVoorsteWiel` na `clWhite`

LET WEL: As jy nie die dinamiese kode vir die vraag kan doen nie, voeg 'n 'shape' sonder dinamiese kode by, sodat jy VRAE 1.2.2 – 1.2.4 kan voltooi. Punte vir die dinamiese kode word dan verbeur.

(6)

1.2.2 Knoppie [V1.2.2 – Lig landingsrat op]

In die Cessna 182-RG, wanneer jy die landingrat oplaai, dan word al die ligte op die paneelbord Rooi.

Kodeer die knoppie `btnV1_2_2_Op` om al drie ligte na Rooi te verander.

(1)

1.2.3 Knoppie [V1.2.3 – Sit landingsrat neer]

In die Cessna 182-RG, wanneer jy die landingsrat neer sit, dan word al die ligte op die paneelbord Groen.

Kodeer die knoppie `btnV1_2_3_Af` om al drie ligte na Groen te verander.

(1)

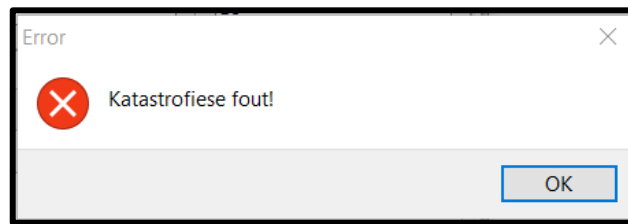
1.2.4 Knoppie [V1.2.4 – Toets landingsrat]

Wanneer enige vliegtuig met intrekbare landingstoerusting land, moet alle landingstoerusting in die geslote posisie wees en die ligte op die paneelbord moet groen wees.

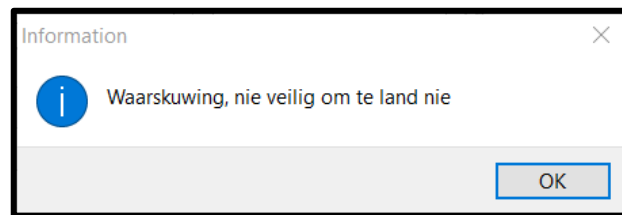
Gegewe kode sal die posisie van die landingrat elke keer kies wanneer die gebruiker op die knoppie klik.

Skryf kode vir knoppie `btnV1_2_4 – Toets` om te toets of dit veilig is om te land. Vertoon 'n boodskap deur 'n 'Message Dialog box' te gebruik om die verskillende landingskondisies voor te stel. *Neem kennis van die tipe boodskappe in elke afvoer/skermkoot.*

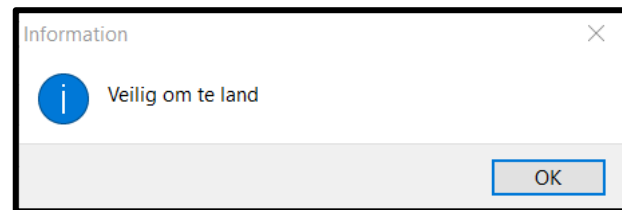
- 3 Rooi Ligte → Katastrofiese fout!



- 1 of 2 Rooi Ligte → Waarskuwing, nie veilig om te land nie



- 3 Groen Ligte → Veilig om te land



(8)

1.3 Knoppie [V1.3 – Hoogtepunt van Daling]

In lugvaat is die Hoogtepunt van Daling 'Top of Descent' die oorgang van 'cruising'-spoed na die landingsfase. Dit is die punt waar die beplande landingsproses begin. Die Hoogtepunt van Daling word bereken deur die hulp van die aanboortstelsel. Jou Cessna 182-RG het nie 'n aanboortstelsel nie, daarom het jy besluit om 'n program te skryf om die Hoogtepunt van Daling te bereken.

3 Konstante waardes is vir jou geskep:

- DEG_TO_RAD (grade na radiale)
- NM_TO_FT (seemyle na voet)
- FT_TO_NM (voet na seemyle)

Die gebruiker sal die volgende inligting in tik:

- Beginhoogte en eindhoogte (gemeet in voet)
- Beginspoed en eindspoed (gemeet in knope)
- Windspoed (gemeet in knope)
- Helling (gemeet in grade)

Vertikale spoedformule:

- $Gemiddelde\ spoed = (Beginspoed \times Eindspoed) \div (2 + Windspoed)$
- $VertikaleSpoed_NM_P_Min = \tan(Helling \times DEG_TO_RAD) \times Gemiddelde\ spoed \div 60$
- $Vertikale\ spoed = VertikaleSpoed_NM_P_Min \times NM_TO_FT$

Afstand formule:

- $\Delta\ hoogte = Beginhoogte - Eindhoogte$
- $\Delta\ Spoed = Beginspoed - Eindspoed$
- $Afstand = (\Delta\ hoogte \times FT_TO_NM) \div \tan(Helling \times DEG_TO_RAD)$
- $Afstand = Afstand + \text{die 'ceiling' funksie van } \Delta\ Spoed \text{ gedeel deur } 10$
- $Afstand = Afstand + \text{die 'ceiling' funksie van } Windspoed \text{ gedeel deur } 10$

Geskatte tydformule:

- $Geskatte\ tyd = (Afstand \div gemiddelde\ spoed) \times 60$

Gebruik die formules hierbo, kodeer die knoppie **btnV1_3_Hoogtepunt_van_Daling** om die volgende inligting in die 'rich edit' **redV1_3** te vertoon.

- Vertikale spoed van afname (gemeet in voet per minuut)
- Afstand na landingsarea (gemeet in seemyle)
- Geskatte tyd wat dit gaan neem om by landingsarea te kom (gemeet in minute)

(20)

Sien voorbeelde van skermskote/afvoer die volgende bladsy:

VOORBEELD 1

Vraag 1.3 - Hoogtepunt van Daling	
Beginhoogte (voet)	Beginspoed (knope)
30000	300
Eindhoogte (voet)	Eindspoed (knope)
5000	100
Helling (°)	+/- Wind (knope)
3	10
V1.3 Hoogtepunt van Daling	
Vertikale Spoed: 1114.53 (fpm) Afstand: 99.51 (nm) Geskatte Tyd: 28.43 (min)	

VOORBEELD 2

Vraag 1.3 - Hoogtepunt van Daling	
Beginhoogte (voet)	Beginspoed (knope)
30000	150
Eindhoogte (voet)	Eindspoed (knope)
5000	40
Helling (°)	+/- Wind (knope)
3	10
V1.3 Hoogtepunt van Daling	
Vertikale Spoed: 557.26 (fpm) Afstand: 90.51 (nm) Geskatte Tyd: 51.72 (min)	

VOORBEELD 3

Vraag 1.3 - Hoogtepunt van Daling	
Beginhoogte (voet)	Beginspoed (knope)
3000	100
Eindhoogte (voet)	Eindspoed (knope)
0	0
Helling (°)	+/- Wind (knope)
6	0
V1.3 Hoogtepunt van Daling	
Vertikale Spoed: 532.19 (fpm) Afstand: 14.70 (nm) Geskatte Tyd: 17.64 (min)	

- Sleutel jou naam en van as kommentaar in die eerste reël van die programlêer.
- Stoor jou program.
- Druk die kode indien dit vereis word.

[40]

VRAAG 2: SQL EN DATABASISPROGRAMMERING

Die bladsye met data aan die einde van die vraestel verskaf inligting oor die ontwerp van die databasis en inhoud van die **VlugInligting.mdb** databasis.

Doen die volgende:

- Kompileer en voer die program in die Vraag 2 lêergids uit. Die program het tans geen funksionaliteit nie.
- Sleutel jou naam as kommentaar in die **Vraag2_u.pas**.
- Volg die instruksies hieronder om die kode vir elke afdeling te voltooi, soos in VRAAG 2.1 en VRAAG 2.2 beskryf word.
- Gebruik SQL-stellings om VRAAG 2.1 te beantwoord en Delphi-kode in VRAAG 2.2.

LET WEL:

- Die 'Herstel databasis'-knoppie word verskaf om die data wat in die databasis voorkom, na die oorspronklike inhoud te herstel.
- Die inhoud van die databasis is met 'n wagwoord beskerm, dit beteken dat jy NIE toegang tot die inhoud van die databasis in Microsoft Access sal kry NIE.
- Kode word voorsien om die G GK('GUI')-komponente met die databasis te koppel. MOENIE enige van die kode wat voorsien word, verander NIE.
- DRIE veranderlikes word as publieke veranderlikes verklaar. Beskrywings vir hierdie veranderlikes word in die tabel hieronder gelys.

Veranderlike	Datatype	Beskrywing
tblVlugte	TADOTable	Verwys na data wat in tblVlugte gestoor is
tblVlieeniers	TADOTable	Verwys na data wat in tblVlieeniers gestoor is
qryInligting	TADOQuery	Query-komponent wat albei tabelle gebruik om inligting te verkry

- Gebruik die ADO-komponente tblVlug en tblVlieenier slegs in **Vraag 2.2**.

Voorbeeld van gebruikerskoppelvlak ('GUI'):

The screenshot shows a Delphi application window titled 'Vraag 2'. It contains two data grids. The top grid, 'Vlugte Tabel', lists flight information with columns: VlugID, Vliegtuig, VliegtuigModel, LisensieVereiste, AantalPassasiers, Bestemming, LughaweNaam, VertrekDatum, and Med. The bottom grid, 'Vlieeniers Tabel', lists pilot information with columns: VlieenierID, Naam, Van, Ouderdom, Epos, and Ge. To the right of the grids are two buttons labeled 'V 2.2.1' and 'V 2.2.2'. At the bottom of the window is a button labeled 'Herstel Databasis'.

- 2.1 In hierdie afdeling mag jy SLEGS **SQL-stellings** gebruik om VRAAG 2.1.1 tot VRAAG 2.1.5 te beantwoord.

Kode word voorsien om die **SQL-stellings** uit te voer en om die resultate van die navrae te vertoon. Die SQL-stellings is onvolledig.

LET WEL: As jy VRAAG 2.2 gekodeer en uitgevoer het, sal jou uitslae vir VRAAG 2.1 anders wees as die voorbeelde van afvoer wat gegee word. Klik asseblief op die **Herstel Databasis** knoppie voordat jy die SQL-stellings uit voer.

2.1.1 Knoppie [V2.1.1]

Skryf SQL-kode om die Naam, Van, Ouderdom en Epos van al die vlieëniers tussen die ouderdom van 36 (ingesluit) en 42 (ingesluit) te vertoon, volgens die ouderdom van oudste tot jongste.

Voorbeeld van afvoer:

Naam	Van	Ouderdom	Epos
Guenna	Louden	42	glouden1x@google.pl
Teador	Egre	42	tegrer@chron.com
Barday	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
Arlene	Persicke	37	apersickeq@prnewswire.com
Hewett	Maase	36	hmaase2m@myspace.com
Luca	Glenfield	36	lglenfield3l@mapquest.com

(4)

2.1.2 Knoppie [V2.1.2]

As die gebruiker wil sien watter vlugte is beskikbaar na hul gekose bestemming, moet hul die bestemming in 'n 'input box' insleutel. Die 'input box'-kode word verskaf en sal die waarde in 'n veranderlike sBestemming stoor.

Skryf SQL-kode om al die velde van die rekords te vertoon wat die bestemming, wat in die 'input box' getik is, bevat.

Voorbeeld van afvoer as die woord 'Alfred' ingetik is:

VlugID	Vliegtuig	VliegtuigModel	LisensieVereiste	AantalPassasiers	Bestemming	LughaweNaam	VertrekDatum	MedeVlieënierNodig	Vluggoede	VlieënierID
1	Cessna	182	PVL	4	Port Alfred	Port Alfred Aerodrome	2023/02/02	False	2200	64
4	Cessna	172	PVL	4	Port Alfred	Port Alfred Aerodrome	2023/03/05	False	2200	118
13	Rans	S4	MVL	2	Port Alfred	Port Alfred Aerodrome	2023/08/21	False	1100	171
41	Piper	Cub	PVL	2	Port Alfred	Port Alfred Aerodrome	2023/04/05	False	1100	171

(4)

2.1.3 Knoppie [V2.1.3]

Jy wil graag weet hoeveel vliegtuie in die maand van September vlieg.

Skryf SQL-kode om die aantal vlugte in September te vertoon. Noem die berekende veld 'Vlugte in September'.

Voorbeeld van afvoer:

Vlugte in September
6

(3)

2.1.4 Knoppie [V2.1.4]

Die lugrederye wil graag al die kostes vir al die vlugte sien gebaseer op die bestemmings.

Skryf 'n SQL-stelling om die Bestemming, Vlieenierkoste, Vlugkoste en Totale Koste vir al die vlugte gebaseer op die bestemmings, te vertoon.

Voorbeeld van afvoer:

Bestemming	Vlieenierkoste	Vlugkoste	Totale Koste
► 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 Knoppie [V2.1.5]

Al die gesertifiseerde KVL (Kommersiële Vlieënierlisensies) vlieëniers word 'n 7% verhoging op die Koste Per Vlugin gegee.

Skryf 'n SQL-stelling om die KVL gesertifiseerde vlieëniers se salarisse met 7% te verhoog.

LET WEL: As jy Knoppie 2.1.4 klik na Knoppie 2.1.5, dan sal die volgende afvoer vertoon word. Slegs George, Graaff Reinet en King Williams Town het geen verandering in vlieënierkoste. Daar is geen KVL-sertifiseerde vlieëniers in daardie dorpe nie.

Bestemming	Vlieënierkoste	Vlugin	Totale Koste
► 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
Gqeberha	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 hierdie afdeling, mag slegs Delphi-programmeringskode gebruik word om VRAAG 2.2.1 en VRAAG 2.2.2 te beantwoord.

Gebruik die globale veranderlikes, tblVlugin en tblVlieënier wat verskaf is.

GEEN punte sal vir SQL-stellings in VRAAG 2.2 toegeken word NIE.

2.2.1 Knoppie [V2.2.1]

Volgens 'n regeringsregulasie mag 'n vlieënier 'n gesertifiseerde lisensie kry as hul ouer as 21 is.

Skryf kode om al die vlieëniers wat jonger as 21 is te verwyder. Vertoon in die 'rich edit' redV2_Afvoer soos in die voorbeeld skermkoot gewys word.

Voorbeeld van afvoer:

Vlieëniers voor verandering: 200
Vlieëniers na die verandering: 193

(3)

2.2.2 Knoppie [V2.2.2]

'n Uiteensetting van die verskillende vliegtuigtipes (lisensievereiste) moet vertoon word. Om afvoer te vereenvoudig, sal die uiteensetting op die bestemmingslughawe gebaseer wees.

Skryf kode om die inligting te verkry gebaseer op die bestemming wat in die 'combo box' cmbV2_2_2_Bestemming gekies is. Bereken en vertoon hoeveel vliegtuigtipes daar volgens die lisensietipes is. 'n Lys van hoeveel medevlieëniers daar is vir die bestemming moet ook vertoon word. Afvoer moet in die 'rich edit' redV2_Afvoer vertoon word.

Voorbeeld van afvoer/skermskoot

East London

KVL: 1
PVL: 3
MVL: 2
Medevlieënier Benodig: 1

Johannesburg

KVL: 8
PVL: 0
MVL: 0
Medevlieënier Benodig: 8

Port Alfred

KVL: 0
PVL: 3
MVL: 1
Medevlieënier Benodig: 0

Cape Town

KVL: 4
PVL: 0
MVL: 0
Medevlieënier Benodig: 4

(6)

- Sleutel jou naam en van as kommentaar in die eerste reël van die programlêer.
- Stoor jou program.
- Druk die kode indien dit vereis word.

[35]

VRAAG 3: OBJEK-GEÏORIENTEERDE PROGRAMMERING

Vliegtuigflitskaarte is 'n uitstekende hulpbron om van verskillende vliegtuie te leer. Jy is hierdeur geïnspireer en het besluit om 'n toepassing te skryf wat hierdie flitskaarte digitaal gaan skep. Deur hierdie kaarte op 'n digitale platform te hê, maak dit moontlik vir jou en ander gebruikers om nie net kennis te verkry nie, maar ook met ander 'n flitskaartspeletjie te speel.

Doen die volgende:

- Kompileer en voer die program in die Vraag 3-lêergids uit. Die program het tans beperkte funksionaliteit.
- Sleutel jou naam as kommentaar in beide die **Vraag3KlasDefinisie_u.pas** en die **Vraag3_u.pas**-lêers in.
- MOENIE gegewe kode verwyder of verander NIE.

Die volgende gebruikerskoppelvlak word vertoon:

Vraag 3

Kies Vliegtuig

Vliegtuignaam
Vervaardiger

Vliegtuigbeskrywing
Vliegtuigbeskrywing

MAKS SPOED (km/h)

MAKS HOOGTE (meters)

AFSTAND (km)

MAKS OPSTYGGEWIG (kg)

VLERKSPAN (meters)

VUURKRAG

Volg die instruksies hieronder om die kode vir elke afdeling van VRAAG 3.1 en VRAAG 3.2 te voltooi.

LET WEL: Vir hierdie vraag mag jy NIE enige addisionele attribute of gebruikergedefinieerde metodes insluit wat nie in die vraag genoem is NIE.

Maak die onvoltooide objekklas **Vraag3KlasDefinisie_u.pas** oop.

- 3.1 Die onvoltooide klas (TVliegtuig) bevat die verklaring van attribute wat die objVliegtuig-objek beskryf.

NAAM VAN ATTRIBUTE	BESKRYWING
fNaam	Stringwaarde wat vliegtuig se naam bevat
fVervaardiger	Stringwaarde wat vervaardiger se naam bevat
fSpoed	'n Reële waarde wat die spoed in knope (knots) bevat
fHoogte	'n Reële waarde wat die hoogte in voet (feet) bevat
fAfstand	'n Reële waarde wat die afstand in myl (miles) bevat
fGewig	'n Reële waarde wat die gewig in pond (pounds) bevat
fVlerkspan	'n Reële waarde wat die vlerkspan in voet (feet) bevat
fVuurkrag	'n Heelgetal tussen 1 – 10
fPrentjieNaam	Stringwaarde wat die prentjie se naam bevat
fLand	Stringwaarde wat die land se naam bevat
fBeskrywing	'Wide' string wat die vliegtuigbeskrywing bevat

- 3.1.1 Skryf kode vir 'n **konstruktor**-metode met die naam **Create** wat die volgende parameters sal ontvang:

Naam

Vervaardiger

Spoed

Hoogte

Afstand

Gewig

Vlerkspan

Vuurkrag

PrentjieNaam

Land

Beskrywing

LET WEL: Hierdie data-times sal van die teksleëer verkry word. Vliegtuig_Lys.csv in die **Vraag3_u.pas** eenheid en sal dan na die konstruktor gestuur word. (5)

Ken hierdie parameterwaardes aan die korrekte attribute toe.

- 3.1.2 Die attribute (Spoed, Hoogte, Afstand, Gewig en Vlerkspan) moet na Suid-Afrikaanse eenhede omgeskakel word. Skryf kode vir 'n **mutator**-metode SetWaardes wat die huidige attribute na hul ooreenstemmende Suid-Afrikaanse maateenheid verander. Die omskakelings is soos volg:

Knope na Kilometers per uur = 1.852

Voet na Meters = 0.3048

Myl na Kilometers = 1.60934

Pond na Kilogram = 0.45359 (5)

3.2 'n Onvolledige eenheid **Vraag3_u.pas** is voorsien.

Dit bevat kode vir die objekklas om toeganklik te wees en verklaar 'n globale objekveranderlike met die naam **objVliegtuig**.

MOENIE gegewe kode verwyder of verander NIE.

Die tekslêer, **Vliegtuig_Lys.csv**, bevat inligting vir 16 vliegtuie soos hieronder vertoon:

```
Vliegtuig_Lys.csv - Notepad
File Edit Format View Help
Vliegtuignaam,Vervaardiger,Maks Spoed,Afstand,Opstyg Gewig,Maks Hoogte,Vlerkspan,Vuurkrag,PrentjieNaam,Vlagnaam,Beskrywing
MAPO-MIG MiG-31,MIKOYAN,1600,1600,101853.444,82000,44.29134,8,Mig31.png,Russia.png,Interceptor aircraft. In service since 1979.
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, t
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
Mirage F1 AZ,DASSAULT,1262,229,35714.844,66000,27.559056,7,F1AZ.png,France.png,"Single-seat ground-attack fighter aircraft. Fitt
A10 Thunderbolt 2,FAIRCHILD-DORNIER,381,250,45999.3963,45000,57.4147,10,A10.png,USA.png,Close support attack and anti armour air
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-capab
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 Am
F-16 Fighting Falcon,GENERAL DYNAMICS,1176,339,42300.04394,58000,32.6771664,8,F16.png,USA.png,"The General Dynamics F-16 Fightin
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, tw
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 America
F-35 Lightning II,LOCKHEED MARTIN,1067,1500,65918.138,50000,36.08924,7,F35.png,USA.png,"The F-35 Lightning II is a single-seat, s
Tornado,PANAVIA,1300,750,44621.5088,50000,45.6364844,7,Tornado.png,UK.png,"The Panavia Tornado is a family of twin-engine, varia
Typhoon,EUROFIGHTER,1147,1600,51808.57,65000,35.925198,8,Typhoon.png,UK.png,"The Eurofighter Typhoon is a European multinational
35 Draken,SAAB,1320,1480,26265.84268,66000,30.9055128,6,Draken.png,Sweden.png,"The Saab 35 Draken ('The Kite' or 'The Dragon') i
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
JAS 39 Gripen,SAAB,1100,430,30864.68,50000,27.559056,8,Gripen.png,Sweden.png,"The Saab JAS 39 Gripen (English: the griffin) is a
```

LET WEL: Die eerste reël van die tekslêer bevat die opskrifte van die verskillende kategorieë van die vliegtuig.

Die prentjies van die vlage kan in die lêergids: **Images\Vlae** gevind word.

Die prentjies van die vliegtuig kan in die lêergids: **Images\Vliegtuig** gevind word.

Volg die instruksies om die oplossing te kodeer.

3.2.1 Combobox [V3.2.1 – cmbV3_KiesVliegtuig]

Wanneer die program uitgevoer word, sal die gebruiker 'n leë flitskaart sien. Die gebruiker sal dan 'n vliegtuig kies wat hulle wil vertoon deur 'n vliegtuig van die 'combo box' te kies.

Kodeer die **OnChange** 'event handler' van die **cmbV3_KiesVliegtuig** 'combo box' om die volgende te doen:

1. Kry die gebruiker se keuse van die 'combo box'. (1)
2. Toets om te sien of die tekslêer bestaan en ken die lêer toe. As die lêer nie bestaan nie, vertoon 'n boodskap en 'exit'. (5)
3. Lus deur die tekslêer totdat die gebruiker se gekose vliegtuig gevind is. (6)
4. As die vliegtuig gevind is:
 - Skryf kode in die lus om die inligting van die tekslêer te verkry en stoor dit in die lokale veranderlikes wat reeds verklaar is. (5)
 - Gebruik die waardes in die veranderlikes om die objek **objVliegtuig** te instansieer (skep). (3)

LET WEL: Slegs een vliegtuig sal in die objek gelaai word. Dit is op die gebruiker se keuse in die 'combo box' gebaseer.

5. Sodra die vliegtuig gevind is moet die lus verlaat word en die objek **objVliegtuig** moet gebruik word om setWaardes metode te roep. (1)
6. Gebruik die objek **objVliegtuig**, om die data in die volgende komponente te laai:

lblV3_Vliegtuignaam
 lblV3_Vervaardiger
 imgV3_Landsvlag
 imgV3_Vliegtuigprent
 lblV3_VliegtuigBeskrywing
 lblV3_MaksSpoed
 lblV3_MaksHoogte
 lblV3_Afstand
 lblV3_MaksOpstygGewig
 lblV3_Vlerkspan
 lblV3_Vuurkrag

(4)

Voorbeeld van 'n flitskaart wanneer 'n vliegtuig gevind is, inligting uit die tekslêer onttrek is, die objek geïnstansieer is en die objek se waardes in die flitskaart gelaai is.

Vraag 3
— □ ×

F-14 Tomcat

F-14 Tomcat

GRUMMAN

Vliegtuigbeskrywing

"The Grumman F-14 Tomcat is an American carrier-capable supersonic, twin-engine, two-seat, twin-tail, variable-sweep wing fighter aircraft. A large and well-equipped fighter designed for air combat against MiG fighters during the Vietnam War."

MAKS SPOED (km/h)	2485
MAKS HOOGTE (meters)	16154
AFSTAND (km)	2575
MAKS OPSTYGGEWIG (kg)	33725
VLERKSPAN (meters)	19.54
VUURKRAG	9

- Sleutel jou naam en van as kommentaar in die eerste reël van die programlêer.
- Stoor jou program.
- Druk die kode indien dit vereis word.

[35]

VRAAG 4: PROBLEEMOPLOSSINGSPROGRAMMERING

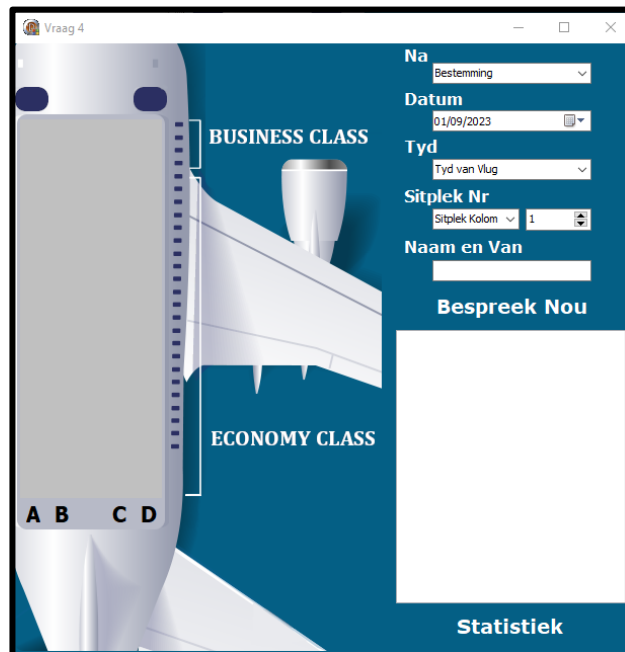
Een van die plaaslike lugrederye het gevra of jy kan help om 'n toepassing te skryf wat die besprekings van hul nuwe vliegtuig, Mitsubishi SpaceJet M100, kan behartig. Die SpaceJet M100 se kajuit het 15 rye van twee sitplekke aan die linkerkant en twee sitplekke aan die regterkant van die gang. Die eerste twee rye is toegeken aan besigheidsklas en die res van die sitplekke is ekonomiese klas.

Doen die volgende:

- Kompileer en voer die program in die Vraag 4-lêergids uit. Die program het tans beperkte funksionaliteit.
- Sleutel jou naam as kommentaar in die eerste reël van die **Vraag4_u.pas**-lêer in.
- MOENIE gegewe kode verwyder of verander NIE.
- Aanvaar dat die gebruiker die korrekte inligting in die toevoerkomponente sal in sleutel. Jy hoef nie enige datavalideringkode vir die toevoerkomponente te skryf nie.

Die volgende is 'n voorbeeld van die koppelvlak:

As die prentjie op jou GGK nie soos die voorbeeld lyk nie, maar seker dat jou rekenaar se Windows Scaling op 'n 100% gestel is.



Vliegtuigbron: Skypark Secure

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

Die gebruikerskoppelvlak maak gebruik van 'n 'String Grid'-komponent om die beskikbaarheid van sitplekke op die vliegtuig te vertoon. Al die data wat bepaal of 'n sitplek beskikbaar is of nie, is in 'n 2D-skikking met die naam `ar2Bespreking`, gestoor. Hierdie 2D-skikking word gebruik om die 'String Grid'-komponent te opdateer en om die kleur van die sitplek te verander.

LET WEL: Alle 'String Grid'-kode word vir jou gegee. Moenie enige 'String Grid'-kode verander of verwyder nie. Jy hoef GEEN 'string grid'-kode te skryf NIE.

Uiteensetting van die data:

Karakterkode wat in die 2D-skikking: ar2Bespreking gestoor is	Beskikbaarheid	Kleur van 'String Grid'-sel
'B'	Bespreekte sitplek	Blou
'A'	Beskikbare sitplek	Wit
'I'	Gang	Grys

Volg die instruksies hieronder uit om die kode vir elke afdeling in VRAAG 4, soos beskryf in VRAAG 4.1, VRAAG 4.2 en VRAAG 4.3, te voltooi.

- 4.1 Die lugredery laat passasiers toe om hul vlugte van Gqeberha te bespreek. Die inligting van die beskikbaarheid van sitplekke vir die verskeie bestemmings is in die globale 2D-skikking gestoor. *Hierdie kode is vir jou gedoen en moenie verwyder of verander word nie:*

- Bloemfontein ar2Bloemfontein
- Kaapstad ar2Kaapstad
- Durban ar2Durban
- Oos-Londen ar2OosLonden
- Johannesburg a2Johannesburg

Skryf kode vir die **'onChange'** gebeurtenis van cmbV4_1_Bestemming om die volgende te doen:

1. Kry die gebruiker se bestemming van die 'combo box'. (1)
2. Dra die data van die bestemming 2D-skikking na die globale 2D-skikking, ar2Bespreking oor. (5)

- 4.2 Die program moet die gebruiker toe laat om 'n sitplek te bespreek. Nadat hul die bestemming gekies het, moet hulle hul inligting in die toevoerkomponente tik.

Skryf kode vir die **'onClick'** gebeurtenis van die paneel, met die naam pnlV4_2_Bespreek, (die paneel word as 'n knoppie gebruik) om die volgende te doen:

1. Kry die sitpleknommer en lus deur die ar2Bespreking-skikking om te bepaal of die sitplek bespreek is of nie.

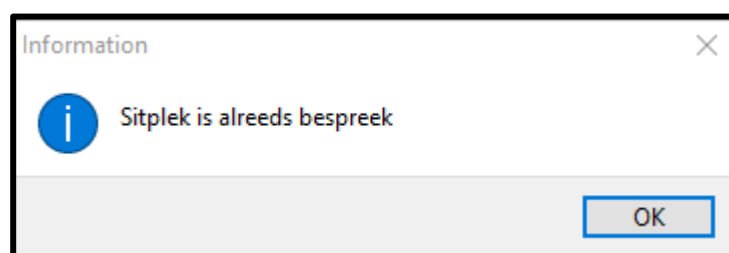
'A' = *Beskikbaar*

'B' = *Bespreek*

'I' = *Gang*

LET WEL: Dat daar slegs vier sitplekkolomme, A..D, is; alhoewel ar2Bespreking 5 kolomme het waarvan die 3de kolom die gang is. Maak seker dat jou kolomme A..D in jou kode met die ar2Bespreking se kolomme ooreenstem.

As die sitplek reeds bespreek is, vertoon 'n boodskap wat aan dui dat die sitplek alreeds bespreek is.



(5)

2. As die sitplek nog nie bespreek is nie, bespreek die sitplek in ar2Bespreking en verkry al die inligting van die verskeie toevoerkomponente.

Die pryse van die verskeie vlugte is in 'n globale skikking, arrPryse, gestoor. Elke bestemming het 'n verskillende prys gebaseer op die bestemming. Besigheidsklas-passasiers betaal 95% meer vir hul vlugte as ekonomiese klas-passasiers.

Verskeie voorbeelde van afvoer van sitplekkaartjie inligting in die 'rich edit' redV4_Afvoer.

The image shows three screenshots of a flight booking interface, each with a blue header and a white body. The first screenshot shows a flight from Bloemfontein to Oos-Londen, departing on 2023/09/01 at 08:00, for a business class passenger named James Wheeler. The second screenshot shows a flight from Oos-Londen to Oos-Londen, departing on 2023/09/01 at 10:00, for a business class passenger named Sarah King. The third screenshot shows a flight from Oos-Londen to Oos-Londen, departing on 2023/09/01 at 10:00, for an economy class passenger named Sarah King. Each screenshot includes a 'Besprek Nou' button and a 'Bespreking Inligting' section with details like Name, Destination, Date, Time, Class, and Price.

Na	Datum	Tyd	Sitplek Nr	Naam en Van	Klas	Koste
Bloemfontein	2023/09/01	Vroegoggend	A 1	James Wheeler	Besigheidsklas	R4 229.55
Oos-Londen	2023/09/01	Aand	B 1	Sarah King	Besigheidsklas	R4 756.05
Oos-Londen	2023/09/01	Aand	B 10	Sarah King	Ekonomiese Klas	R2 439.00

(12)

4.3 Die lugredery wil graag statistieke sien van die verskeie vlugte.

Skryf kode vir die **onClick** gebeurtenis van die paneel met die naam pnIV4_3_Stats (die paneel word gebruik as 'n knoppie) om die volgende te doen:

1. Kry die bestemming.
2. Lus deur ar2Bespreking om te bepaal hoeveel besigheidsklas-passasiers en ekonomiese klas-passasiers daar is.
3. Bepaal die koste vir besigheidsklas asook ekonomiese klas. Besigheidsklas-passasiers betaal 95% meer as ekonomiese klas-passasiers. Die pryse is in 'n globale skikking, arrPryse, gestoor.
4. Vertoon die statistieke van die vlug in die 'rich edit' redV4_Afvoer soos in die voorbeelde skermkote/afvoere op die volgende bladsy.

Hierdie voorbeelde is van die verskeie bestemmings sonder enige addisionele sitplekke wat bespreek is.

Bloemfontein

Statistiek

Kaapstad

Statistiek

Durban

Statistiek

Johannesburg

Statistiek

- Sleutel jou naam en van as kommentaar in die eerste reël van die programlêer.
- Stoor jou program.
- Druk die kode indien dit vereis word.

[40]

GROOTTOTAAL: 150

DATABASISINLIGTING VRAAG 2:

Die ontwerp van die databasistabelle is soos volg:

Tabel: **tblVlugte**

Die tabel bevat inligting van die verskeie vlugte.

	Field Name	Data Type
🔑	VlugID	AutoNumber
	Vliegtuig	Short Text
	VliegtuigModel	Short Text
	LisensieVereiste	Short Text
	AantalPassasiers	Number
	Bestemming	Short Text
	LughaweNaam	Short Text
	VertrekDatum	Date/Time
	MedeVlieënierNodig	Yes/No
	Vlugkoste	Currency
	VlieënierID	Number

Voorbeeld van die eerste twintig rekords in die **tblVlugte**-tabel:

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

Tabel: **tblVlieëniers**

Die tabel bevat inligting van die vlieëniers:

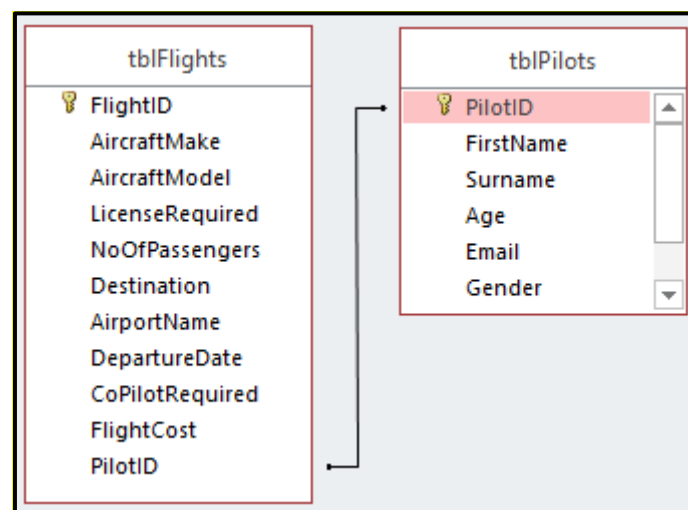
	Field Name	Data Type
🔑	VlieënierID	AutoNumber
	Naam	Short Text
	Van	Short Text
	Ouderdom	Number
	Epos	Short Text
	Geslag	Short Text
	LisensieTipe	Short Text
	VlieënierKostePerVlug	Currency

Voorbeeld van die eerste twintig rekords in die **tblVlieënier**-tabel:

	VlieënierID ▾	Naam ▾	Van ▾	Ouderdom ▾	Epos ▾	Geslag ▾	LisensieTip ▾	VlieënierKo ▾
+	1	Alden	Bamford	34	abamford0@n	Manlik	MVL	R376.00
+	2	Jeralee	Gubbin	67	jgubbin1@goo	Vroulik	KVL	R2 943.00
+	3	Angele	Stamper	64	astamper2@tv	Vroulik	PVL	R1 437.00
+	4	Urbain	Ewin	21	uewin3@ask.c	Manlik	PVL	R569.00
+	5	Garfield	Cerro	72	gcerro4@w3.o	Manlik	MVL	R416.00
+	6	Sheila	Elverstone	41	selverstone5@	Vroulik	KVL	R9 679.00
+	7	Natala	Couchman	30	ncouchman6@	Vroulik	KVL	R5 870.00
+	8	Davis	Sterland	69	dsterland7@n	Vroulik	MVL	R345.00
+	9	Cortney	MacFadin	50	cmacfadin8@c	Vroulik	KVL	R9 978.00
+	10	Shaylah	Rockcliff	47	srockcliff9@ta	Vroulik	PVL	R846.00
+	11	Caryl	Pollins	40	cpollinsa@goc	Vroulik	PVL	R1 412.00
+	12	Floris	Kamien	43	fkamienb@orr	Vroulik	MVL	R314.00
+	13	Viola	Nockalls	62	vnockallsc@nt	Vroulik	KVL	R3 840.00
+	14	Susi	Scampion	19	sscampiond@v	Vroulik	KVL	R3 588.00
+	15	Catie	Curzey	55	ccurzeye@hon	Vroulik	PVL	R921.00
+	16	Cesaro	Pardoe	26	cpardoe@fiz	Manlik	MVL	R424.00
+	17	Arney	Ilyukhov	46	ailyukhovg@b	Manlik	MVL	R317.00
+	18	Fayette	Sirrell	40	fsirrellh@saku	Vroulik	PVL	R802.00
+	19	Joleen	Andover	29	jandoveri@vis	Vroulik	PVL	R1 096.00
+	20	Ariana	Ledgerton	62	aledgertonj@r	Vroulik	KVL	R3 619.00

LET WEL: Konneksiekode is voorsien.

Die volgende een-tot-baie verwantskap bestaan tussen die twee tabelle in die databasis:





**NASIONALE
SENIORSERTIFIKAAT**

GRAAD 12

SEPTEMBER 2023

**INLIGTINGSTEGNOLOGIE V1
NASIENRIGLYN**

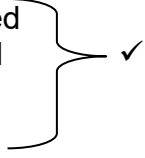
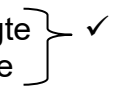
PUNTE: 150

Hierdie nasienriglyn bestaan uit 20 bladsye.

NAAM VAN LEERDER:

TOTAAL VRAAG 1	TOTAAL VRAAG 2	TOTAAL VRAAG 3	TOTAAL VRAAG 4	TOTAAL
/40	/35	/35	/40	/150

VRAAG 1		MAKS PUNTE	PUNTE BEHAAL
1.1	KNOPPIE: [V1.1 – Bepaal koste] Gebruik 'n if / case stelling ✓ Bepaal die lisensietipe uit die radio group ✓ Vertoon die lisensietipe ✓ asook die koste van die lisensie in die Koste-label komponent ✓	4	
1.2.1	KNOPPIE: [V1.2.1 – Voeg voorste wiel by] Skep dinamiese vorm: shpVoorsteWiel ✓ Stel 'parent' van shpFrontWheel na die paneel pnIV1_2_Ligte ✓ Verander die eienskappe van shpVoorsteWiel: Type → Circle ✓ Top → 25 ✓ Left → 35 Width → 50 ✓ Height → 50 Colour → White ✓	6	
1.2.2	KNOPPIE: [V1.2.2 – Lig landingsrat op] Verander die kleur van al drie vorme na Rooi ✓	1	
1.2.3	KNOPPIE: [V1.2.3 – Sit landingsrat neer] Verander die kleur van al drie vorme na Groen ✓	1	
1.2.4	KNOPPIE: [V1.2.4 – Toets landingsrat] Gebruik 'n case of if statement, bepaal die toestand van die ligte en skep die korrekte boodskap met 'n Message DLG As al 3 ligte = Rooi ✓ → MessageDLG ✓ ('Katastrofiese fout') ✓ Else ✓ As al 3 ligte = Groen ✓ → MessageDLG('Veilig om te land') ✓ Else ✓ MessageDLG('Waarskuwing, nie veilig om te land nie') ✓	8	

1.3	KNOPPIE: [V1.3 Hoogtepunt van Daling] Vertikale Spoed Kry BeginSpoed Kry EindSpoed Kry Wind Kry Helling  $\text{GemiddeldeSpoed} = (\text{BeginSpoed} + \text{EindSpoed}) / 2 + \text{Wind} \checkmark \checkmark$ $\text{VertikaleSpoed_NM_P_Min} = (\tan(\text{Helling} * \text{DEG_TO_RAD}) * \text{GemiddeldeSpoed}) / 60 \checkmark \checkmark$ $\text{VertikaleSpoed} := \text{VertikaleSpoed_NM_P_Min} * \text{NM_TO_FT} \checkmark$ Afstand Kry BeginHoogte Kry EindHoogte  $\text{DeltaHoogte} = \text{BeginHoogte} - \text{EindHoogte} \checkmark$ $\text{Afstand} = (\text{DeltaHoogte} * \text{FT_TO_NM}) / \tan(\text{Helling} * \text{DEG_TO_RAD}); \checkmark \checkmark$ $\text{DeltaSpoed} = \text{BeginSpoed} - \text{EindSpoed}; \checkmark$ $\text{Afstand} = \text{Afstand} + \text{ceil}(\text{DeltaSpoed} / 10) \checkmark$ $\text{Afstand} = \text{Afstand} + \text{ceil}(\text{Windspoed} / 10) \checkmark$ Tyd $\text{Geskatte tyd} = (\text{Afstand} / \text{GemiddeldeSpoed}) * 60 \checkmark$ Maak rich edit skoon Voeg by rich edit: Vertikale Spoed: VertikaleSpoed (fpm) – <i>geformateer na 2 desimale</i> $\checkmark \checkmark$ Afstand: Afstand (nm) – <i>geformateer na 2 desimale</i> $\checkmark \checkmark$ Geskatte Tyd: Geskatte tyd (min) – <i>geformateer na 2 desimale</i> $\checkmark \checkmark$	20	
	Vraag 1 Totaal	40	

VRAAG 2		MAKS PUNTE	PUNTE BEHAAL
2.1.1	Knoppie [V2.1.1] 'SELECT Naam, Van, Ouderdom, Epos FROM tblVlieeniers WHERE Ouderdom > 35 AND Ouderdom < 43 ORDER BY Ouderdom DESC' SELECT vier korrekte velde ✓ FROM korrekte tabel ✓ WHERE Ouderdom in die reeks van 36 (insl) en 42 (insl) ✓ ORDER BY korrekte veld DESC ✓	4	
2.1.2	Knoppie [V2.1.2] 'SELECT * FROM tblVlugte WHERE Bestemming LIKE' + QuotedStr('%' + sLine + '%') SELECT * (alle velde) ✓ FROM korrekte tabel ✓ WHERE Bestemming LIKE ✓ QuotedStr('%' + sLine + '%') ✓	4	
2.1.3	Knoppie [V2.1.3] 'SELECT count(*) AS [Vlugte in September] FROM tblVlugte WHERE Month(VertrekDatum) = 9' SELECT count(*) ✓ AS [Vlugte in September] FROM tblVlugte WHERE Month ✓ (VertrekDatum) = 9 ✓	3	
2.1.4	Knoppie [V2.1.4] 'SELECT Bestemming, Format(sum(VlieenierKostePerVlug), "Currency") AS [Vlieenierkoste], Format(sum(Vlugkoste), "Currency") AS [Vlugkoste], Format(sum(VlieenierKostePerVlug) + sum(Vlugkoste), "Currency") AS [Totale Koste] FROM tblVlieeniers, tblVlugte WHERE tblVlieeniers.VlieenierID = tblVlugte.VlieenierID GROUP BY Bestemming' SELECT Bestemming ✓ , Format ✓ (sum ✓ (VlieenierKostePerVlug) ✓ , "Currency" ✓) AS [Vlieenierkoste] ✓ , Format(sum(Vlugkoste), "Currency") AS [Vlugkoste] ✓ , Format((sum(VlieenierKostePerVlug) ✓ + sum(Vlugkoste)) ✓ , "Currency") AS [Totale Koste] FROM albei tabelle ✓ (tblVlieeniers, tblVlugte) WHERE verbinding tussen tabelle ✓ (tblVlieeniers.VlieenierID = tblVlugte.VlieenierID) GROUP BY Bestemming ✓	12	
2.1.5	Knoppie [V2.1.5] 'UPDATE tblVlieeniers SET VlieenierKostePerVlug = VlieenierKostePerVlug * 1.07 WHERE LisensieTipe = "KVL"' UPDATE korrekte tabel ✓ SET VlieenierKostePerVlug = VlieenierKostePerVlug * 1.07 ✓ WHERE LisensieTipe = "KVL" ✓	3	

2.2.1	Knoppie [V2.2.1] if FieldByName('Ouderdom').AsInteger < 21 then ✓ delete ✓ else next; ✓	3	
2.2.2	Knoppie [V2.2.2] begin if Bestemming = sBestemming then ✓ begin if LisensieVereiste = 'KVL' then inc(iKVL) else if LisensieVereiste = 'PVL' then inc(iPVL) else if LisensieVereiste = 'MVL' then inc(iMVL); if MedeVlieenier = True then ✓ inc(imedevlieenier) end ✓ <i>NB: MedeVlieenier moet binne die begin en end wees, anders tel dit op l die medevlieeniers vir al die bestemmings.</i> Next ✓ end Afvoer ('KVL: ' + IntToStr(iKVL) + #13 + 'PVL: ' + IntToStr(iPVL) + #13 + 'MVL: ' + IntToStr(iMVL) + #13 + 'Medevlieenier nodig: ' + IntToStr(imedevlieenier));	6	
	Vraag 2 Totaal	35	

VRAAG 3		MAKS PUNTE	PUNTE BEHAAL
3.1.1	Konstruktor Create Korrekte opskrif en parameters ✓✓ fNaam := sNaam fVervaardiger := sVervaardiger fSpoed := rSpoed fHoogte := rHoogte fAfstand := rAfstand fGewig := rGewig fVlerkspan := rVlerkspan fVuurkrag := iVuurkrag fPrentjieNaam := sPrentjieNaam fLand := sLand fBeskrywing := sBeskrywing Korrekte toekenning van al die attribute ✓✓✓	5	
3.1.2	Mutator Metode – setWaardes Korrekte opskrif (prosedure setWaardes) ✓ Korrekte omskakeling van elke attribuut ✓✓✓✓	5	
3.2.1	OnChange gebeurtenis van cmbV3_KiesVliegtuig 1. Kry die gebruikers se keuse uit die combo box ✓ 2. Toets om te sien of tekslêer bestaan en ken lêer toe. As die lêer nie bestaan nie, vertoon 'n boodskap en exit AssignFile(MyFile, 'Vliegtuig_Lys.csv'); ✓ Try ✓ Reset(MyFile); ✓ except ShowMessage('lêer nie gevind nie'); ✓ Exit; ✓ end; <i>Of alternatief: if not (fileexists(textfile) = true) then</i> 3. Lus deur die tekslêer totdat die vliegtuig gevind is. bFound := False; ✓ while (not eof(MyFile)) ✓ AND (bFound = False) ✓ do if pos(UpperCase(sSearch),UpperCase(sOnline)) ✓ <> 0 ✓ then bFound := True; ✓ end while 4. As die vliegtuig gevind is, dan: Lus en onttrek die inligting van die tekslêer //Vliegtuignaam iPos := pos(',',sOneLine); ✓ sNaam := copy(sOneLine,1,iPos-1); ✓ delete(sOneLine,1,iPos); ✓ //Ander velde ✓✓ Instansieer (skep) die objek objVliegtuig . objVliegtuig := TVliegtuig.Create ✓(sNaam, sVervaardiger, rSpoed, rHoogte, rAfstand, rGewig, rVlerkspan, iVuurkrag, sPrentjieNaam, sLand, sBeskrywing); ✓✓ 5. Roep die setWaarde-metode ✓ 6. Laai die objekdata in die komponente ✓✓✓✓	25	
Vraag 3 Totaal		35	

VRAAG 4		MAKS PUNTE	PUNTE BEHAAL
4.1	Kry die bestemming uit die combo box ✓ Ken die bestemmingskikking aan die ar2Bespreking toe ✓✓✓✓✓	6	
4.2	1. Kry die sitpleknommer ✓✓ Lus deur die skikking en bepaal of die sitplek bespreek is ✓✓ Vertoon 'n boodskap as die sitplek reeds bespreek is ✓ 2. Bespreek die sitplek in ar2Bespreking ✓ Kry al die data ✓ Bepaal klas ✓ Bepaal prys ✓ If Besigheidsklas * 1.95 ✓ Vertoon opskrif Besprekinginligting ✓ in bold ✓ Vertoon besprekingskaartjie-inligting geformateer: 'Naam en Van: ' + #13#9 + sNaamVan + #13 + ✓ 'Bestemming: ' + #13#9 + sBestemming + #13 + 'Datum en Tyd: ' + #13#9 + sDatum + #13 + #9 + sTyd + ' vlug' + #13 + 'Klas: ' + #13#9 + sKlas + #13 + 'Sitpleknommer: ' + #13#9 + cCol ✓ + IntToStr(iRow + 1)✓ + #13 + 'Koste: ' + #13#9 + FloatToStrF(rPrice,ffCurrency✓,10,2);	17	
4.3	1. Kry die bestemming ✓ 2. Lus deur ar2Bespreking en vermeerder die aantal passasiers: Besigheidsklas Lus deur Rye 0 to 1 ✓ Lus deur kolomme 0 to 4 ✓ if ar2Bespreking[Row,Col] = 'B' then ✓ vermeerder(besigheidsklas teller) ✓ en Ekonomiese Klas Lus deur rye 2 tot 14 } ✓ Lus deur kolomme 0 tot 4 } if ar2Bespreking[Row,Col] = 'B' then } ✓ vermeerder(ekonomiese klas teller) } 3. Bepaal koste vir besigheidsklas ✓✓ en ekonomiese klas ✓✓ 4. Vertoon besprekingskaartjie-inligting geformateer 'Passasiers' + #13 + #9 + 'Besigheidsklas: ' + IntToStr(iBusClass) + #13 + ✓ #9 + 'Ekonomiese klas: ' + IntToStr(iEcoClass) + #13 + ✓ #9 + 'Totaal: ' + IntToStr(iBusClass + iEcoClass) + #13#13 + ✓ 'Koste' + #13 + #9 + 'Besigheidsklas: ' + FloatToStrF(rBusPrice,ffCurrency,10,2) + #13 + ✓ #9 + 'Ekonomiese klas: ' + FloatToStrF(rEcoPrice,ffCurrency,10,2) + #13 + ✓ #9 + 'Totale Koste: ' + FloatToStrF(rBusPrice + rEcoPrice,ffCurrency,10,2);✓	17	
Vraag 4 Totaal		40	

VOORBEELDE EN OPLOSSINGS**VRAAG 1**

////////// 40 punte //////////

////////// Vraag 1.1 – 4 punte //////////

```
procedure TfrmQuestion1.btnQ1_1_CostClick(Sender: TObject);
begin
  case rgpQ1_1_License.ItemIndex of
    0 : lblQ1_1_Cost.Caption := 'Microlight Pilot License = R37 000';
    1 : lblQ1_1_Cost.Caption := 'Private Pilot License = R110 451';
    2 : lblQ1_1_Cost.Caption := 'Commercial Pilot License = R761 379';
  end;
end;
```

////////// Vraag 1.2.1 – 6 punte //////////

```
procedure TfrmQuestion1.btnQ1_2_1Click(Sender: TObject);
begin
  shpFrontWheel := TShape.Create(frmQuestion1);
  shpFrontWheel.Parent := pnlQ1_2_Lights;

  with shpFrontWheel do
    begin
      Shape := stCircle;
      Top := 25;
      Left := 35;
      Height := 50;
      Width := 50;
      Brush.Color := clWhite;
    end;
end;
```

////////// Vraag 1.2.2 – 1 punt //////////

```
procedure TfrmQuestion1.btnQ1_2_2_UpClick(Sender: TObject);
begin
  btnQ1_2_1.Click; //Provided code, DO NOT DELETE
  shpFrontWheel.Brush.Color := clRed;
  shpLeftWheel.Brush.Color := clRed;
  shpRightWheel.Brush.Color := clRed;
end;
```

////////// Vraag 1.2.3 – 1 Punt //////////

```
procedure TfrmQuestion1.btnQ1_2_3_DownClick(Sender: TObject);
begin
  btnQ1_2_1.Click; //Provided code, DO NOT DELETE
  shpFrontWheel.Brush.Color := clGreen;
  shpLeftWheel.Brush.Color := clGreen;
  shpRightWheel.Brush.Color := clGreen;
end;
```

////////// Vraag 1.2.4 – 8 punte //////////

```
procedure TfrmQuestion1.btnQ1_2_4_TestClick(Sender: TObject);
begin
  btnQ1_2_1.Click; //Provided code, DO NOT DELETE
  RandomColours; //Provided code, DO NOT DELETE

  if (shpFrontWheel.Brush.Color = clRed) AND
    (shpLeftWheel.Brush.Color = clRed) AND
    (shpRightWheel.Brush.Color = clRed) then
    MessageDLG('Catastrophic failure!',MTErrors,[MBOk],0)
  else
    if (shpFrontWheel.Brush.Color = clGreen) AND
      (shpLeftWheel.Brush.Color = clGreen) AND
      (shpRightWheel.Brush.Color = clGreen) then
      MessageDLG('Safe to land',MTInformation,[MBOk],0)
    else
      MessageDLG('Caution, not safe to land',MTInformation,[MBOk],0);
```

ALTERNATIEWE OPLOSSING

var

iFront, iLeft, iRight, iTotal : Integer;

iFront := 0;

iLeft := 0;

iRight := 0;

iTotal := 0;

if shpFrontWheel.Brush.Color = clGreen then

iFront := 1;

if shpLeftWheel.Brush.Color = clGreen then

iLeft := 1;

if shpRightWheel.Brush.Color = clGreen then

iRight := 1;

iTotal := iFront + iLeft + iRight;

case iTotal of

0 : MessageDLG('Catastrophic failure!',MTErrors,[MBOk],0);

1,2 : MessageDLG('Caution, not safe to land',MTInformation,[MBOk],0);

3 : MessageDLG('Safe to land',MTInformation,[MBOk],0);

end;

end;

////////// Vraag 1.3 - 20 Punte //////////

procedure TfrmQuestion1.Q1_3_TopOfDescentClick(Sender: TObject);

const DEG_TO_RAD = 0.0174532925;

const NM_TO_FT = 6076.11549;

const FT_TO_NM = 1 / NM_TO_FT;

var

rAverageSpeed, rStartSpeed, rEndSpeed, rVerticalSpeed, rWind : Real;

rGlideSlope, rVerticalSpeed_NM_P_Min : Real;

rStartAltitude, rEndAltitude : Real;

rDistance, rDeltaAltitude, rDeltaSpeed : Real;

rEstimatedTime : Real;

begin

//Vertical Speed

rStartSpeed := StrToFloat(edtQ1_3_StartSpeed.Text);

rEndSpeed := StrToFloat(edtQ1_3_EndSpeed.Text);

rWind := StrToFloat(edtQ1_3_Wind.Text);

rGlideSlope := StrToFloat(edtQ1_3_GlideSlope.Text);

rAverageSpeed := (rStartSpeed + rEndSpeed) / 2 + rWind;

rVerticalSpeed_NM_P_Min := (tan(rGlideSlope * DEG_TO_RAD) * rAverageSpeed) / 60;

rVerticalSpeed := rVerticalSpeed_NM_P_Min * NM_TO_FT;

//Distance

rStartAltitude := StrToFloat(edtQ1_3_StartAltitude.Text);

rEndAltitude := StrToFloat(edtQ1_3_EndAltitude.Text);

rDeltaAltitude := rStartAltitude - rEndAltitude;

rDistance := (rDeltaAltitude * FT_TO_NM) / tan(rGlideSlope * DEG_TO_RAD);

rDeltaSpeed := rStartSpeed - rEndSpeed;

rDistance := rDistance + ceil(rDeltaSpeed / 10);

rDistance := rDistance + ceil(rWind / 10);

//Time

rEstimatedTime := (rDistance / rAverageSpeed) * 60;

redQ1_3.Clear;

redQ1_3.Lines.Add('Vertical Speed: ' + FloatToStrF(rVerticalSpeed, ffFixed, 10, 2) + ' (fpm)' +
#13 +

'Distance: ' + FloatToStrF(rDistance, ffFixed, 10, 2) + ' (nm)' + #13 +

'Estimated Time: ' + FloatToStrF(rEstimatedTime, ffFixed, 10, 2) + ' (min)');

end;

VRAAG 2

////////// 35 punte //////////

////////// Vraag 2.1.1 – 4 punte //////////

procedure TfrmQuestion2.btnQuestion2_1_1Click(Sender: TObject);

// Provided code - DO NOT DELETE OR ALTER //

var

sSQL1: String;

begin

/// Enter your code below ///

sSQL1 := 'SELECT FirstName, Surname, Age, Email ' +

'FROM tblPilots ' +

'WHERE Age > 35 AND Age < 43 ' +

'ORDER BY Age DESC';

// Provided code - DO NOT DELETE OR ALTER //

dbCONN.runSQL(sSQL1);

if length(sSQL1) <> 0 then

SetGridColumnWidths(dbgSQL);

end;

////////// Vraag 2.1.2 - 4 punte //////////

procedure TfrmQuestion2.btnQuestion2_1_2Click(Sender: TObject);

// Provided code - DO NOT DELETE OR ALTER //

var

sSQL2: String;

sLine : String;

begin

/// Enter your code below ///

sLine := InputBox('Destination','Enter the destination','');

sSQL2 := 'SELECT * ' +

'FROM tblFlights ' +

'WHERE Destination LIKE ' + QuotedStr('%' + sLine + '%');

// Provided code - DO NOT DELETE OR ALTER //

dbCONN.runSQL(sSQL2);

if length(sSQL2) <> 0 then

SetGridColumnWidths(dbgSQL);

end;

////////// Question 2.1.3 - 3 Marks //////////

procedure TfrmQuestion2.btnQuestion2_1_3Click(Sender: TObject);

// Provided code - DO NOT DELETE OR ALTER //

var

sSQL3 : String;

begin

/// Enter your code below ///

sSQL3 := 'SELECT count(*) AS [Flights in September] ' +

'FROM tblFlights ' +

```
'WHERE Month(DepartureDate) = 9';
```

```
// Provided code - DO NOT DELETE OR ALTER //  
dbCONN.runSQL(sSQL3);  
if length(sSQL3) <> 0 then  
    SetGridColumnWidths(dbgSQL);  
end;
```

```
////////// Vraag 2.1.4 - 12 punte //////////
```

```
procedure TfrmQuestion2.btnQuestion2_1_4Click(Sender: TObject);
```

```
// Provided code - DO NOT DELETE OR ALTER //
```

```
var
```

```
    sSQL4: String;
```

```
begin
```

```
    /// Enter your code below ///
```

```
    sSQL4 := 'SELECT Destination, ' +
```

```
        'Format(sum(PilotCostPerFlight), "Currency") AS [Pilot Cost], ' +
```

```
        'Format(sum(FlightCost), "Currency") AS [Flight Cost], ' +
```

```
        'Format(sum(PilotCostPerFlight) + sum(FlightCost), "Currency") AS [Total Cost] ' +
```

```
        'FROM tblPilots, tblFlights ' +
```

```
        'WHERE tblPilots.PilotID = tblFlights.PilotID ' +
```

```
        'GROUP BY Destination';
```

```
// Provided code - DO NOT DELETE OR ALTER //
```

```
dbCONN.runSQL(sSQL4);
```

```
if length(sSQL4) <> 0 then
```

```
    SetGridColumnWidths(dbgSQL);
```

```
end;
```

```
////////// Vraag 2.1.5 - 3 punte //////////
```

```
procedure TfrmQuestion2.btnQuestion2_1_5Click(Sender: TObject);
```

```
// Provided code - DO NOT DELETE OR ALTER //
```

```
var
```

```
    sSQL5: String;
```

```
begin
```

```
    /// Enter your code below ///
```

```
    sSQL5 := 'UPDATE tblPilots ' +
```

```
        'SET PilotCostPerFlight = PilotCostPerFlight * 1.07 ' +
```

```
        'WHERE LicenseType = "CPL";
```

```
// Provided code - DO NOT DELETE OR ALTER //
```

```
dbCONN.executeSQL(sSQL5,dbgPilots,dbgFlights,dbgSQL);
```

```
if length(sSQL5) <> 0 then
```

```
    SetGridColumnWidths(dbgSQL);
```

```
end;
```

////////// Vraag 2.2.1 - 3 punte //////////

```
procedure TfrmQuestion2.btnQuestion2_2_1Click(Sender: TObject);
begin
  // Provided code - DO NOT DELETE OR ALTER //
  redQ2_Output.Clear;

  with tblPilot do
    begin
      Open;
      redQ2_Output.Lines.Add('Pilots before regulation change: ' + IntToStr(RecordCount));
      First;
      while not (eof) do
        begin
          /// Enter your code below ///
          if FieldByName('Age').AsInteger < 21 then
            Delete
          else
            Next;
          end;
          redQ2_Output.Lines.Add('Pilots after regulation change: ' + IntToStr(RecordCount));
        end;
      end;
    end;
```

////////// Vraag 2.2.2 - 6 punte //////////

```
procedure TfrmQuestion2.btnQuestion2_2_2Click(Sender: TObject);
```

```
// Provided code - DO NOT DELETE OR ALTER //
```

```
var
  sDestination : string;
  iCPL, iPPL, iMPL, iCoPilot : Integer;
```

```
begin
```

```
// Provided code - DO NOT DELETE OR ALTER //
```

```
redQ2_Output.Clear;
iCPL := 0;
iPPL := 0;
iMPL := 0;
iCoPilot := 0;
```

```
with tblFlight do
```

```
begin
  Open;
  sDestination := cmbQ2_2_2_Destination.Text;
  redQ2_Output.SelAttributes.Style := [fsBold];
  redQ2_Output.Lines.Add(sDestination);
  redQ2_Output.Lines.Add('-----');
  First;
```

```
while not (eof) do
  begin
    /// Enter your code below ///
    if FieldByName('Destination').AsString = sDestination then
      begin
        if FieldByName('LicenseRequired').AsString = 'CPL' then
          inc(iCPL)
        else
          if FieldByName('LicenseRequired').AsString = 'PPL' then
            inc(iPPL)
          else
            if FieldByName('LicenseRequired').AsString = 'MPL' then
              inc(iMPL);
            if (FieldByName('CoPilotRequired').AsBoolean = True) then
              inc(iCoPilot);
            end;
          Next;
        end;
      redQ2_Output.Lines.Add('CPL: ' + IntToStr(iCPL) + #13 +
        'PPL: ' + IntToStr(iPPL) + #13 +
        'MPL: ' + IntToStr(iMPL) + #13 +
        'Co-Pilots required: ' + IntToStr(iCoPilot));
    end;
  end;
```

VRAAG 3

////////// 35 punte //////////

////////// Vraag 3.1.1 – 5 punte //////////

```
constructor TAircraft.Create(sName, sManufacturer : String;
                           rSpeed, rHeight, rRange, rWeight : Real;
                           rWingspan : Real;
                           iFirepower : Integer;
                           sImageName, sCountry : String;
                           sDescription : WideString);
```

```
begin
  fName      := sName;
  fManufacturer := sManufacturer;
  fSpeed      := rSpeed;
  fHeight     := rHeight;
  fRange      := rRange;
  fWeight     := rWeight;
  fWingspan   := rWingspan;
  fFirepower  := iFirepower;
  fImageName  := sImageName;
  fCountry    := sCountry;
  fDescription := sDescription;
end;
```

////////// Vraag 3.1.2 – 5 punte //////////

```
procedure TAircraft.setValues;
begin
  fSpeed      := fSpeed * 1.852;      //Knots converted to KM/h
  fHeight     := fHeight * 0.3048;    //Feet converted to Meters
  fRange      := fRange * 1.60934;    //Miles converted to KM
  fWeight     := fWeight * 0.45359;   //Pounds converted to KG
  fWingspan   := fWingspan * 0.3048   //Feet converted to Meters
end;
```

////////// Question 3.2.1 - 20 marks //////////

```
procedure TfrmQuestion3.cmbQ3_SelectAircraftChange(Sender: TObject);
var
  sName, sManufacturer, sImageName, sCountry: String;
  rSpeed, rHeight, rRange, rWeight : Real;
  rWingspan : Real;
  iFirepower : Integer;
  sDescription : WideString;
  MyFile : Textfile;
  sOneLine : String;
  sSearch : String;
  iPos : Integer;
  bFound : Boolean;
```

```
begin
  sSearch := cmbQ3_SelectAircraft.Text;

  AssignFile(MyFile, 'Aircraft_List.csv');
  try
    Reset(MyFile);
  except
    ShowMessage('File not found');
    Exit;
  end;

  bFound := False;
  ReadLn(MyFile, sOneLine); //Skip the heading line in the textfile
  while (not eof(MyFile)) AND (bFound = False) do
    begin
      ReadLn(MyFile, sOneLine);
      if pos(UpperCase(sSearch),UpperCase(sOnline)) <> 0 then
        begin
          //Aircraft Name
          iPos := pos(',',sOneLine);
          sName := copy(sOneLine,1,iPos-1);
          delete(sOneLine,1,iPos);
          //Manufacturer Name
          iPos := pos(',',sOneLine);
          sManufacturer := copy(sOneLine,1,iPos-1);
          delete(sOneLine,1,iPos);
          //Speed
          iPos := pos(',',sOneLine);
          rSpeed := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
          //Range
          iPos := pos(',',sOneLine);
          rRange := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
          //Weight
          iPos := pos(',',sOneLine);
          rWeight := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
          //Height
          iPos := pos(',',sOneLine);
          rHeight := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
          //Wingspan
          iPos := pos(',',sOneLine);
          rWingspan := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
          //Firepower
          iPos := pos(',',sOneLine);
          iFirepower := StrToInt(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
          //Image Name
          iPos := pos(',',sOneLine);
          sImageName := copy(sOneLine,1,iPos-1);
```

```
    delete(sOneLine,1,iPos);
//Country
    iPos := pos(',',sOneLine);
    sCountry := copy(sOneLine,1,iPos-1);
    delete(sOneLine,1,iPos);
//Description
    sDescription := sOneLine;
    objAircraft := TAircraft.Create(sName,sManufacturer,rSpeed,rHeight,rRange,
                                   rWeight,rWingspan,iFirepower,sImageName,sCountry,sDescription);
    bFound := True;
end;
end;

if bFound then
begin
    objAircraft.SetValues;
    lblQ3_AircraftName.Caption := objAircraft.getName;
    lblQ3_Manufacturer.Caption := objAircraft.getManufacturer;
    imgQ3_CountryFlag.Picture.LoadFromFile('Images\Flags\' + objAircraft.getCountry);
    imgQ3_AircraftImage.Picture.LoadFromFile('Images\Aircraft\' +
        objAircraft.getImageName);
    lblQ3_AircraftDescription.Caption := objAircraft.getDescription;
    lblQ3_MaxSpeed.Caption := FloatToStrF(objAircraft.getSpeed,ffFixed,10,0);
    lblQ3_MaxHeight.Caption := FloatToStrF(objAircraft.getHeight,ffFixed,10,0);
    lblQ3_Range.Caption := FloatToStrF(objAircraft.getRange,ffFixed,10,0);
    lblQ3_MaxTakeoffWeight.Caption := FloatToStrF(objAircraft.getWeight,ffFixed,10,0);
    lblQ3_Wingspan.Caption := FloatToStrF(objAircraft.getWingspan,ffFixed,10,2);
    lblQ3_Firepower.Caption := IntToStr(objAircraft.getFirepower);
end;
end;
```

VRAAG 4

////////// 40 punte //////////

////////// Vraag 4.1 – 6 punte //////////

```
procedure TfrmQuestion4.cmbQ4_1_DestinationChange(Sender: TObject);
var
```

```
    iRow, iCol : Integer;
begin
    case cmbQ4_1_Destination.ItemIndex of //Case or If Statement
        0 : ar2Booking := ar2Bloemfontein;
        1 : ar2Booking := ar2CapeTown;
        2 : ar2Booking := ar2Durban;
        3 : ar2Booking := ar2EastLondon;
        4 : ar2Booking := ar2Johannesburg;
    end;
```

```
// //Alternate Solution
// case cmbQ4_Destination.ItemIndex of //Case or If Statement
// 0 : for iRow := 0 to 14 do
//     for iCol := 0 to 4 do
//         ar2Booking[iRow,iCol] := ar2Bloemfontein[iRow,iCol];
// 1 : for iRow := 0 to 14 do
//     for iCol := 0 to 4 do
//         ar2Booking[iRow,iCol] := ar2CapeTown[iRow,iCol];
// 2 : for iRow := 0 to 14 do
//     for iCol := 0 to 4 do
//         ar2Booking[iRow,iCol] := ar2Durban[iRow,iCol];
// 3 : for iRow := 0 to 14 do
//     for iCol := 0 to 4 do
//         ar2Booking[iRow,iCol] := ar2EastLondon[iRow,iCol];
// 4 : for iRow := 0 to 14 do
//     for iCol := 0 to 4 do
//         ar2Booking[iRow,iCol] := ar2Johannesburg[iRow,iCol];
// end;
```

```
//Provide Code - DO NOT DELETE
    PaintColour;
end;
```

////////// Question 4.2 - 17 marks //////////

```
procedure TfrmQuestion4.pnlQ4_2_BookClick(Sender: TObject);
var
```

```
    sDestination, sDate, sNameSur, sTime : String;
    sClass, sLine : String;
    rPrice : Real;
    iRow, iCol : Integer;
    cCol : Char;
begin
```

```

redQ4_Output.Clear;
iRow := sedQ4_RowNumber.Value;

case cmbQ4_ColNumber.ItemIndex of
  0 : begin
    iCol := 0;
    cCol := 'A';
  end;
  1 : begin
    iCol := 1;
    cCol := 'B';
  end;
  2 : begin
    iCol := 3;
    cCol := 'C';
  end;
  3 : begin
    iCol := 4;
    cCol := 'D';
  end;
end;

if ar2Booking[iRow-1,iCol] = 'B' then
  MessageDLG('Seat has already been booked',MTInformation,[MBOK],0)
else
begin
  ar2Booking[iRow-1,iCol] := 'B';
  sDestination := cmbQ4_1_Destination.Text;
  sDate := DateToStr(dtpQ4_Date.Date);
  sTime := cmbQ4_Time.Text;
  sNameSur := edtQ4_NameSur.Text;
  sClass := 'Economy Class';
  rPrice := arrPrice[cmbQ4_1_Destination.ItemIndex];
  case iRow of
    1,2 : begin
      rPrice := rPrice * 1.95;
      sClass := 'Business Class';
    end;
  end;
  sLine := 'Name and Surname: ' + #13#9 + sNameSur + #13 +
    'Destination: ' + #13#9 + sDestination + #13 +
    'Date and Time: ' + #13#9 + sDate + #13 +
    #9 + sTime + ' flight' + #13 +
    'Cabin: ' + #13#9 + sClass + #13 +
    'Seat Number: ' + #13#9 + cCol + IntToStr(iRow + 1) + #13 +
    'Price: ' + #13#9 + FloatToStrF(rPrice,ffCurrency,10,2);
  redQ4_Output.SelAttributes.Style := [fsBold];
  redQ4_Output.Lines.Add('Booking Information' + #13);
  redQ4_Output.Lines.Add(sLine);
end;
PaintColour;
end;

```

////////// Vraag 4.3 – 17 punte //////////

```

procedure TfrmQuestion4.pnlQ4_3_StatsClick(Sender: TObject);
var
  sLine : String;
  iRow, iCol : Integer;
  iBusClass, iEcoClass : Integer;
  rBusPrice, rEcoPrice : Real;
begin
  redQ4_Output.Clear;
  redQ4_Output.Paragraph.TabCount := 1;
  redQ4_Output.Paragraph.Tab[0] := 10;
  iBusClass := 0;
  iEcoClass := 0;
  rBusPrice := 0;
  rEcoPrice := 0;
  for iRow := 0 to 1 do
    for iCol := 0 to 4 do
      begin
        if ar2Booking[iRow,iCol] = 'B' then
          inc(iBusClass);
        end;

        for iRow := 2 to 14 do
          for iCol := 0 to 4 do
            begin
              if ar2Booking[iRow,iCol] = 'B' then
                inc(iEcoClass);
              end;
            rBusPrice := iBusClass * (arrPrice[cmbQ4_1_Destination.ItemIndex] * 1.95);
            rEcoPrice := iEcoClass * (arrPrice[cmbQ4_1_Destination.ItemIndex]);
            redQ4_Output.SelAttributes.Style := [fsBold];

            sLine := 'Passengers' + #13 +
              #9 + 'Business Class: ' + IntToStr(iBusClass) + #13 +
              #9 + 'Economy Class: ' + IntToStr(iEcoClass) + #13 +
              #9 + 'Total: ' + IntToStr(iBusClass + iEcoClass) + #13#13 +
              'Cost' + #13 +
              #9 + 'Business Class: ' + FloatToStrF(rBusPrice,ffCurrency,10,2) + #13 +
              #9 + 'Economy Class: ' + FloatToStrF(rEcoPrice,ffCurrency,10,2) + #13 +
              #9 + 'Total Cost: ' + FloatToStrF(rBusPrice + rEcoPrice,ffCurrency,10,2);

            redQ4_Output.Lines.Add('Statistics of Flight' + #13);
            redQ4_Output.Lines.Add(sLine);
          end;
        end;
      end;
    end;
  end;

```

TOTAAL: 150