INLIGTINGS-TEGNOLOGIE

(Praktiese vraestel (V1) – Hersiening)

Graad 12

Hersienings-ondersteunings-pakket

Hierdie pakket bestaan uit die hersieningsmateriaal vir Gr 12 IT ter voorbereiding van die WKOD-proef en die NSS-eindeksamen.

GESKEP DEUR DIE IT-ONDERWYSERS VAN DIE WKOD

Western Cape Government

Hierdie pakket vorm die hersieningsmateriaal vir Gr 12 IT vir inhoud wat gedoen is volgens die WKOD-ATP's..

Die hersieningsondersteuningspakket sal bestaan uit kognitief gebalanseerde vrae volgens die praktiese vraestelle. Leiding word gegee om die vraag duidelik te verstaan en waarna gesoek moet word om elke vraag te antwoord.

Praktiese vraestel -Vraestel 1:
AFDELING A: Algemene programmeringsvaardighede
AFDELING B: Databasisprogrammering
AFDELING C: Objekgeoriënteerde programmering
AFDELING D: Probleemoplossing

Ontwerpelemente vir hierdie Hersienings ondersteuningspakket sal bestaan uit:

a. Aanduidings oor die inhoud van die vraag. (Weereens sal ons na die DBE-handboekbladsye verwys, ens. Dit is 'n gratis bron waartoe leerders toegang het)
b. Vrae wat:
   - konsepte / inhoud wat waarskynlik in die proef- en finale eksamenvraestelle gevra sal word, dek
   - in die taal en struktuur van die finale NSS-vraestelle sal wees
   - 'vars' vrae sal wees, nie vorige NSS vrae nie
   - in balans is tussen ongeveer 30:40:30 persent laer orde: middelorde: hoër orde kognitiewe vrae
   - 'n memorandum / nasienriglyn sal hê
c. Die vrae wat gevra word, sal:
   - leiding aan die leerder gee soos waaraan aandag te skenk wanneer die vraag to beantwoord word
   - waar beskikbaar, stap-vir-stap leiding gee om die vraag te beantwoord.
   - waar beskikbaar, aantekeninge by 'n tutor vir bemiddeling van die inhoud.
   - Skakels na video's oor die beantwoording van die onderskeie vrae en subvrae

Antwoorde, Memorandum en Merkstate:
Die volgende URL bevat die memorandum van die vrae:
https://wcedeportal.co.za/eresource/143876

Uitleg van die praktiese vraestel:

Die vraestel bestaan uit VIER afdelings:

AFDELING	BESKRYWING
A	**Algemene programmeringsvaardighede (± 40 punte)** Vrae wat die fundamentele programmerings- en probleemoplossingsvaardighede toets.
B	**Databasisprogrammering (± 40 punte)** Vrae wat die manipulasie van 'n relasionele databasis met behulp van SQL toets en manipulasie met behulp van data-bewuste komponente sonder SQL
C	**Objekgeoriënteerde programmering (± 40 punte)** Vrae rakende die voltooiing van die ontwerp van 'n klas. Die skepping van 'n objek van daardie klas en die gebruik daarvan
D	**Probleemoplossing (± 30 punte)** Oop vrae waar die vaardigheid om al die programmeringskonstrukte, tegnieke, algoritmes en vaardighede wat bestudeer is, te toets.

INSTRUKSIES EN INLIGTING

1. Hierdie vraestel is in VIER afdelings verdeel. Jy moet AL VIER afdelings beantwoord.
2. Maak seker dat jy die vrae volgens die spesifikasies wat in elke vraag gegee word, beantwoord. Punte sal slegs volgens die voorgeskrewe vereistes toegeken word.
3. Beantwoord in elke vraag slegs wat gevra is. Byvoorbeeld, indien die vraag nie datavalidering vereis nie, sal geen punte vir datavalidering toegeken word nie.
4. Jou programme moet so gekodeer word dat dit met enige data sal werk en nie net met die voorbeelddata wat voorsien is of enige data-uittreksels wat in die vraestel verskyn nie.
5. **Roetines soos soek, sorteer en seleksie** moet uit **eerste beginsels** ontwikkel word. Jy mag nie die ingeboude funksies van 'n programmeringstaal vir enige van hierdie roetines gebruik nie.
6. Datastrukture wat nie voorsien is, moet deur jou, die programmeerder, verklaar word.
7. Maak seker dat jou **Naam en Van** as kommentaar verskyn in <u>elke</u> eenheid wat jy kodeer.
8. Die lêers wat jy benodig om hierdie vraestel te voltooi, is beskikbaar by die volgende skakel: https://wcedeportal.co.za/eresource/143876

<u>Vraag 1</u>	<u>Vraag 2</u>	<u>Vraag 3</u>	<u>Vraag 4</u>
Question1_P.dpr Question1_P.res Question1_U.dcu Question1_U.dfm Question1_U.pas	DB.mdb DBBackup.mdb Question2_P.dpr Question2_P.res Question2_U.dcu Question2_U.dfm Question2_U.pas TDBConnection_U.dcu TDBConnection_U.pas	PaintJob_U.dcu PaintJob_U.pas Question3_P.dpr Question3_P.res Question3_U.dcu Question3_U.dfm Question3_U.pas	MajorDams.txt Question4_p.dpr Question4_p.res Question4_u.dcu Question4_u.dfm Question4_u.pas

Verander die naam van die DelphiData-lêergids na jou **<u>Van en Naam</u>**.

AFDELING A: ALGEMENE PROGRAMMERINGSVAARDIGHEDE

Basiese vaardighede wat hoofsaaklik in graad 10 en 11 ontwikkel is.
Word ook beskou as maklike punt in ’n praktiese eksamenvraestel.

Hierdie vaardighede en konsepte is:

- Verander die eienskappe (*properties*) van ’n komponent in die *object inspector* en deur kode.
- Verklaar veranderlikes en kies die geskikte data tipe daarvoor.
- Verstaan wanneer ’n globale veranderlike benodig word in teenstelling met ’n lokale like.
- Verkry toevoer vanaf verskeie komponente.
- Skakel data tipe om en ken waardes toe aan veranderlikes.
- Basiese wiskundige bewerkings.
- Gebruik van ingeboude funksies en prosedure.
- Verstaan hoe keusestrukture werk
    - Boolese toets voorwaarde.
    - Waar die kode geplaas moet word wanneer die toets waar of vals evalueer word.
- Verstaan hoe lusstrukture werk
    - Watter lus om te gebruik en hoekom?
    - Lusbeheer – inisialiseer, verander en eindig (ICT)
- Werk met 1D skikkings en tekslêers
- Algoritmes wat in die CAPS dokument genoem word: omruil van waardes, sorteer, bepaal die grootste waarde, priemgetalle, faktore, ens.
- Basiese probleemoplossingsvaardighede.

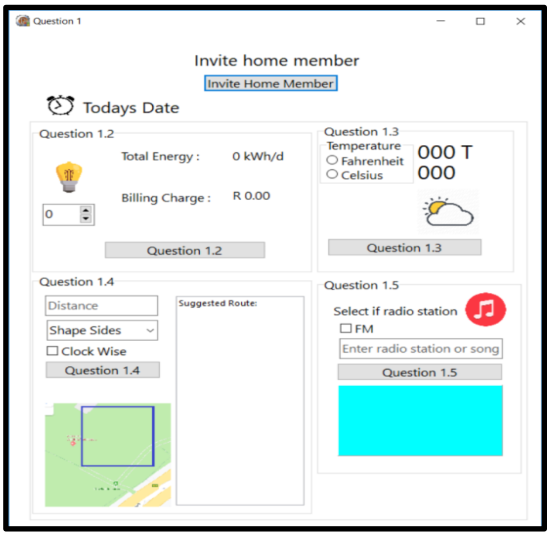
VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE

SCENARIO:

'n Slim huis (*smart home*) is 'n woning waar internet verbinde toestelle gebruik word om toestelle en stelsels soos beligting en verwarming op afstand te monitor en te bestuur. 'n Onderneming wil 'n koppelvlak ontwikkel om 'n slim huis te bestuur en benodig ontwikkelaars om met die basiese ontwerp te help.

Doen die volgende:

- Maak die onvoltooide program, in die **Question1** lêergids oop.
- Voeg jou eksamen nommer as ‘n kommentaar in die eerste lyn van die **Question1_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** te voltooi, soos dit in vrae **VRAAG 1.1** to **VRAAG 1.5** beskryf word.

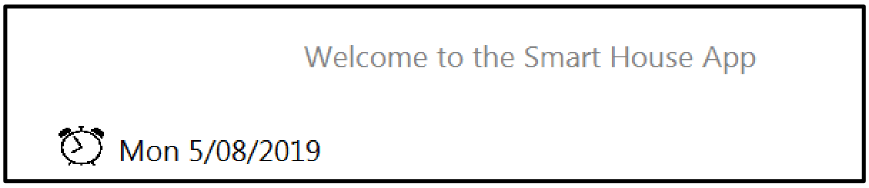


1.1 **KNOPPIE - [Invite Home Member]**

Skryf kode om die volgende te doen:

- Vertoon die boodskap "**Welkom in die Smart House-app**" op **lblMember**.
- Verander die kleur van die teks van **lblMember** na grys.
- Verander die datum van die *label* **lblDate** na vandag se datum.
- Die formaat van die datum moet "**ddd d/mm/yyyy**" wees.
- Verander **btnMembers** sodat dit nie op die vorm vertoon word nie.

VOORBEELD VAN AFVOER:



(5)

Voorafkennis	
Konsep	**Waar om in die DBE handboek te hersien**
Verander die eienskappe van verskeie komponente Lees die stelsel datum en formateer 'n datum	Gr 10 – Unit 2.3 bladsy 24, 25 en bladsy 27 r 11 – Unit 4.3 bladsy 90

Vraag 1.1	
Vraag	**Belangrike konsepte om te doen**
Vertoon die boodskap "**Welcome to die Smart House App**" op **lblMember**.	Ken die welkom-boodskap toe aan die *label* se *caption*
Verander die kleur van die teks van **lblMember** na grys.	Verander die *font* se kleur en nie die *label* se kleur nie.
Verander die datum van die *label* **lblDate** na vandag se datum. Die formaat van die datum moet "**dddd/mm/yyyy**" wees.	Ken die korrekte geformateerde datum to aan die label. Oorweeg die funksies Now() en FormatDateTime
Verander **btnMembers** sodat dit nie op die vorm vertoon word nie.	Verander die *visible* eienskap van die *button* na *true* of *false*

1.2 **KNOPPIE - [Question 1.2]**

1.2.1 Die kragverbruik van 'n slim huis is 'n algemene kwessie.

Skryf kode om die volgende te doen:

- Onttrek die aantal ure uit die **spnLights**.
- Bereken die hoeveelheid **kilowatt** wat in die **maand** gebruik is.
  Kilowatt per maand word soos volg bereken:
  Kilowatt per maand = 0.06 * ***ure*** * 30
- Vertoon die **kilowatts** geformateer met **TWEE** desimale plekke op die **lblWatts** in die volgende formaat:
  <berekende kilowatts> kWh/m

(4)

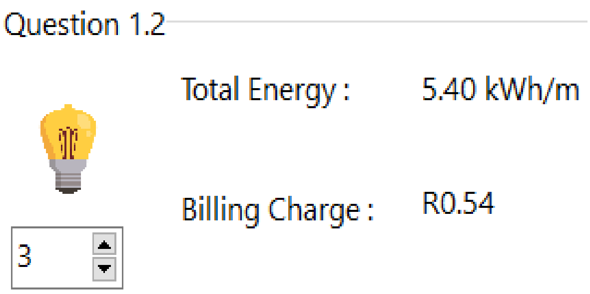
1.2.2 ’n Kragonderneming hef 10 sent per kWh, oftewel R 0.10.

Skryf kode om die volgende te doen:

- Bereken die bedrag wat gehef word deur die kilowatts per maand te vermenigvuldig met 0.10.
- Vertoon die resultaat op **lblAmount** in **GELDEENHEID** (*currency*) formaat.

(2)

VOORBEELD VAN AFVOER:



Voorafkennis	
Konsep	**Waar om in die DBE handboek te hersien**
Verkry die toevoer uit die *spin edit* en ken dit toe aan ’n veranderlike met die korrekte data tipe	Gr 10 – Unit 3.5 bladsy 53
Kodeer verskeie formules	Gr 10 – Unit 4.1 bladsy 62
Formatering van desimale getalle: aantal plekke na die punt of randformaat (currency)	Gr 10 – Unit 4.2 bladsy 69

Vraag 1.2.1	
Vraag	Belangrike konsepte om te doen
Onttrek die aantal ure uit **spnLights**.	Ken die *value* eienskap van die “spin edit” komponent to aan ’n veranderlike
Bereken die aantal kilowatt wat in die maand gebruik is. Kilowatt per maand word soos volg bereken: Kilowatt per maand = 0.06 * ure * 30	Verklaar die korrekte tipe veranderlike vir die kilowatt per maand. Gebruik die ure wat uit die *spin edit* verkry is om die formule te kodeer
Vertoon die kilowatts geformateer tot 2 plekke na die desimale punt op lblWatts in die volgende formaat: *<Berekende kilowatts> kWh/m*	Voeg die eenheid - kW/m by die korrek geformateerde reële getal en ken dit toe aan die *label*

Vraag 1.2.2	
Vraag	Belangrike konsepte om te doen
Bereken die bedrag wat gehef word deur die kilowatt per maand met 0.10 te vermenigvuldig	Verklaar die korrekte tipe veranderlike vir die bedrag en implementeer die berekening
Vertoon die resultaat op **lblAmount** in GELDEENHEID formaat.	Formateer die bedrag as *currency* en ken dit to aan die *label* se *caption*

1.3 **KNOPPIE - [Question 1.3]**

In die weerverslag panel op die Smart Home koppelvlak kan die gebruiker kies tussen die temparatuur eenhede Celsius en Fahrenheit.

Skryf kode vir die volgende:

- Kontroleer dat die gebruiker 'n eenheid vir die temperatuur gekies het en vertoon 'n geskikte boodskap indien niks gekies is nie.
- Genereer **TWEE** lukrake waardes vir die minimum en die maksimum temperature in **Celsius** vir die dag se weervoorspelling. Gebruik die boonste en onderste grense soos aangedui in die onderstaande tabel.

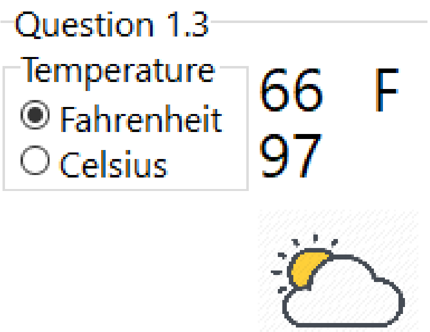
Min Temperatuur			**Maks Temperatuur**	
Onderste grens	Boonste grens		Onderste grens	Boonste grens
18	22		32	38

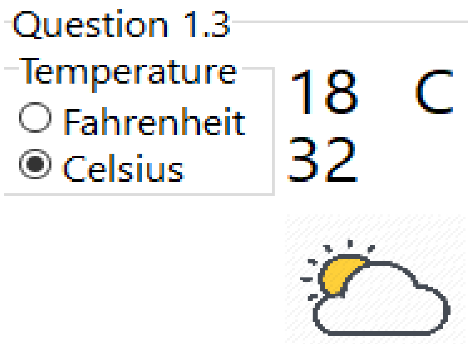
- Indien die gebruiker Fahrenheit gekies het, word die volgende formule gebruik om die temperatuur om te skakel na die minimum en maksimum waardes in Fahrenheit:

  Fahrenheit = (**Celsius** × 9/5) + 32
- Vertoon die minimum en maksimum waardes op **lblMin** en **lblMax**. Beide hierdie waardes moet **AF AFGEROND** word na die naaste heelgetal.
- Vertoon die ooreenstemmende eenheid op **lblDegrees.**

(9)

VOORBEELD VAN AFVOER:





Nota: Omdat lukrake getalle gegenereer word, sal die afvoer telkens verskil.

Voorafkennis	
Konsep	**Waar om in die DBE handboek te hersien**
Kyk of die gebruiker die radioknoppie selekteer het. Gebruik die *showMessage* komponent vir terugvoer.	Gr 10 – Unit 5.2 bladsy 98
Maak gebruik van die korrekte seleksiestruktuur om die vereistes te kodeer volgens die radioknoppie gekies.	Gr 10 – Unit 5.6 bladsy 123
Genereer lukrake nommers tussen die gegewe grense.	Gr 10 - Unit 4.3 bladsy 74
Implementeer die formule met klem op bewerkingsvoorkeur	Gr 10 – Unit 4.1 bladsy 62
Vertoon die waardes op die *labels*	Gr 10 – Unit 2.3 bladsy 24

Vraag 1.3	
Vraag	**Belangrike konsepte om te doen**
Kontroleer dat die gebruiker ’n eenheid vir die temperatuur gekies het en vertoon ’n geskikte boodskap indien niks geselekteer is nie.	Die radioknoppie toets is ook die kondisie van die *if-else of case* struktuur. Plaas die boodskap in die korrekte gedeelte van die seleksiestruktuur.
Genereer TWEE lukrake waardes vir die minimum en die maksimum temperature in **Celsius** vir die dag se weervoorspelling. Gebruik die boonste en onderste grense in die tabel.	Maak gebruik van **Random** of **RandomRange** om die lukrake getalle te genereer in die gegewe reeks. Ken die waardes toe aan twee heelgetal veranderlikes.
Indien die gebruiker Fahrenheit gekies het, word die volgende formule gebruik om die temperatuur om te skakel na die minimum en maksimum waardes in Fahrenheit: **Fahrenheit** = (**Celsius** × 9/5) + 32 Vertoon die minimum en maksimum waardes op **lblMin** en **lblMax**. Beide hierdie waardes moet **AFGEROND** word na die naaste heelgetal	Bereken die Fahrenheit waardes en ken dit aan-aan twee heelgetal veranderlikes. Maak gebruik van die **Floor-**funksie om af af te rond, doen die omskakeling en ken die waarde toe aan die ooreenstemmende *label*. Ken die waardes toe aan die *captions* van die *labels* gebaseer op die radioknoppie geselekteer
Vertoon die ooreenstemmende eenheid op **lblDegrees**.	Ken die korrekte eenheid toe aan die *label caption*. Koppel die eenheid aan die keuse van die radioknoppie bv. as Celsius gekies was moet die C-simbool vertoon word.

1.4 **KNOPPIE - [Question 1.4]**

Die ontwikkelaars wil gebruikers aanmoedig om meer aktief te wees en besluit om aanwysings vir ’n drafroete op die Smart Home-koppelvlak by te voeg.

Skryf kode vir die volgende:

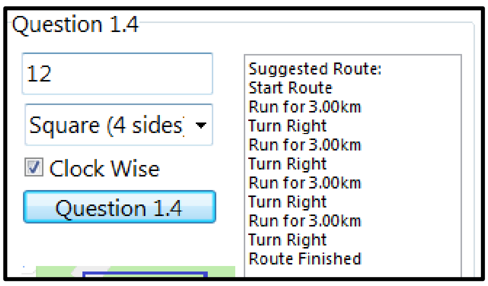
- Lees die aantal sye vir die vorm uit **cmbShape**.
- Bepaal, deur die gebruik van die **chkClock**-komponent, of die gebruiker kloksgewys of antikloksgewys wil hardloop.
- Lees die roete se totale afstand in vanaf **edtDistance**.
- Die afstand voor 'n draai word bereken deur die volgende formule te gebruik:

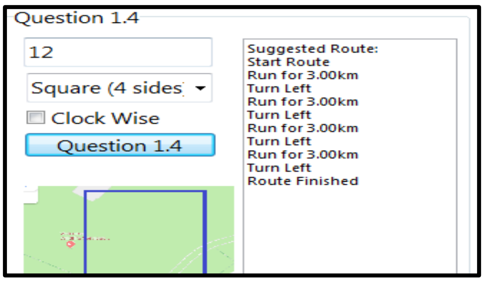
  Afstand voor draai = totale afstand / aantal sye
- Vertoon die aanwysings van die roete op die **redMap**-komponent in die volgende formaat:

  Start Route
  Run for <distance> km
  Turn Left (Anti Clockwise) / Turn Right (Clockwise)
  ...
  Route Finished

(13)

VOORBEELD VAN AFVOER:



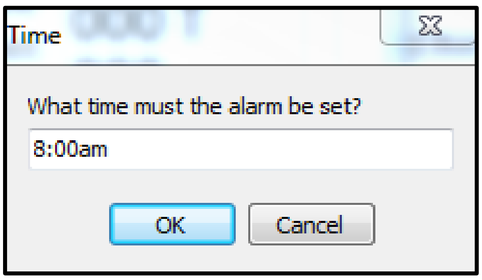


Voorafkennis	
Konsep	**Waar om in die DBE handboek te hersien**
Onttrek data van ’n *combo box* en stoor dit in ’n geskikte veranderlike. Toets of ’n *checkbox* gemerk is of nie. Gebruik hierdie toets in ’n seleksiestruktuur om die rigting van die draai toe te ken. Lees die waarde in die *edit* , skakel dit om en ken dit toe aan die afstand veranderlike. Implementeer die gegewe formule Skep ’n lus vir die vooraf bepaalde aantal kere Vertoon data in ’n *rich edit*	Gr 10 – Unit 7.1 bladsy 155 Gr 10 – Unit 5.3 bladsy 105 Gr 10 - Unit 5.4 bladsy 111 Gr 10 – Unit 3.5 bladsy 49 Gr 10 – Unit 4.1 bladsy 62 Gr 10 – Unit 7.3 bladsy 162 Gr 11 - Unit 5.2 bladsy 109

Vraag 1.4	
Vraag	**Belangrike konsepte om te doen**
Onttrek die aantal sye vir die vorm uit die **cmbShape**.	Gebruik *Combobox*, die *ItemIndex* en ’n *case*-stelling om die korrekte aantal sye toe te ken aan ’n veranderlike.
Bepaal, deur die gebruik van die **chkClock**-komponent, of die gebruiker kloksgewys of antikloksgewys wil hardloop	Toets of die *checkbox* afgemerk is en ken Right of Left toe aan die rigting veranderlike. Maak gebruik van ’n seleksiestruktuur.
Lees die roete se totale afstand in vanaf **edtDistance**	Onttrek gebruikerstoevoer uit die *edit*, skakel dit om en ken dit toe aan ’n geskikte veranderlike.
Die afstand voor 'n draai word bereken deur die volgende formule te gebruik: **Afstand voor draai = totale afstand / aantal sye**	Implementeer die gegewe formule.
Vertoon die aanwysings van die roete op die **redMap**-komponent in die volgende formaat: Start Route Run for <distance> km Turn Left (Anti Clockwise)/Turn Right (Clockwise) ... Route Finished	Bestudeer die gegewe formaat. Let op dat gedeelte van die inligting herhaal word en ’n geskikte lusstruktuur dus gebruik moet word. Identifiseer wat binne en wat buite die lus geplaas moet word, watter veranderlike die lus moet beheer en watter deel is vaste teks (*hardcoded*) is.

1.5 **KNOPPIE - [Question 1.5]**

Die ontwikkelaars wil 'n alarm by die Smart Home-koppelvlak voeg.
Skryf kode om ‘n naam vir die alarm te skep:

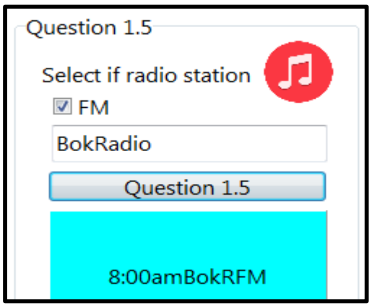


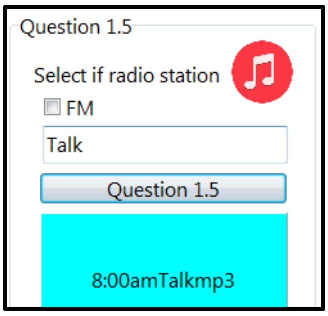
- Verkry die naam van die radiostasie of die liedjie vanaf **edtMusic**.
- Toets dat die gebruiker wel 'n waarde ingevoer het en vertoon 'n toepaslike foutboodskap indien daar niks ingevoer is nie.
- Vra die gebruiker om 'n tyd in te voer deur gebruik te maak van ‘n invoerdialoog (*input box*). Die verstekwaarde (*default value*) van die invoerdialoog moet op 8:00am gestel wees.
- Kontroleer, met behulp van die **chkFm**-komponent, of 'n radiostasie of liedjie gekies is.
- Die formaat van die alarmnaam indien **chkFm** geselekteer is, is as volg:

  <alarm tyd><eerste vier letters van die radio stasie>FM

- Die formaat van die alarmnaam indien **chkFm** NIE geselekteer is NIE, is as volg:
  <alarm tyd><eerste vier letters van die liedjie>mp3
- Genereer die alarmnaam en vertoon dit op die **pnlInfo**-komponent. (7)

VOORBEELD VAN AFVOER:





Voorkennis	
Konsep	**Waar om in die DBE handboek te hersien**
Onttrek data uit ’n *edit* komponent.	Gr 10 - Unit 3.5 bladsy 49
Doen validasie op toevoer uit *edit*.	Gr 10 - Unit 6.2 bladsy 137
Verkry toevoer deur gebruik te maak van ’n toevoerdialoog komponent.	Gr 10 - Unit 7.5 bladsy 173
Toets of ’n *checkbox* gemerk is of nie.	Gr 10 - Unit 5.3 bladsy 105
Gebruik die toets in ’n seleksiestruktuur om die afvoer te formateer.	Gr 10 - Unit 5.4 bladsy 111
Vertoon data op ’n *panel* komponent soortgelyk aan ’n *label*.	Gr 10 - Unit 2.3 bladsy 24
Onttrek karakters uit ’n karakterstring.	Gr 11 - Unit 4.1 bladsy 70

Vraag 1.5	
Vraag	**Belangrike konsepte om te doen**
Verkry die naam van die radiostasie of die liedjie vanaf **edtMusic**.	Lees *edit*, ken toe aan ’n veranderlike
Toets dat die gebruiker wel 'n waarde ingevoer het en vertoon 'n toepaslike foutboodskap indien daar niks ingevoer is nie.	Toets vir leë karakterstring, vertoon boodskap. Gebruik ’n if-else struktuur.
Vra die gebruiker om 'n tyd in te voer deur gebruik te maak van ‘n invoerdialoog (*input box*). Die verstekwaarde (*default value*) van die invoerdialoog moet op 8:00am gestel wees.	Maak gebruik van invoerdialoog om die tyd te verkry, stel die verstekwaarde soos aangedui in die vraag. Ken die tyd toe aan ’n veranderlike.
Kontroleer, met behulp van die **chkFm**-komponent, of 'n radiostasie of liedjie gekies is. Die formaat van die alarmnaam indien **chkFm** geselekteer is, is as volg: <alarm tyd><eerste vier letters van die radio stasie>FM Die formaat van die alarmnaam indien **chkFm** NIE geselekteer is NIE, is as volg: <alarm tyd><eerste vier letters van die liedjie>mp3	Toets of die *checkbox* afgemerk is en voeg die afvoerteks saam volgens die gegewe formaat in ’n if-else struktuur. Gebruik die **Copy**-funksie om die eerste 4 karakters te onttrek. Ken die afvoer toe aan ’n veranderlike.
Genereer die alarmnaam en vertoon dit op die **pnlInfo**-komponent.	Ken afvoer veranderlike toe aan die *caption* van die *panel*

TOTAAL AFDELING A: 40

AFDELING B: DATABASISPROGRAMMERING

Databasis vaardighede ontwikkel in graad 11 en 12.

SQL in graad 12 (Vraag 2.1) en ADO-komponente in graad 11(Vraag 2.2)

Hierdie vaardighede en konsepte is:

- In die eksamen word die konneksie na die databasis gegee in teenstelling met die PAT waar die leerder hierdie vaardigheid moet demonstreer.
- In beide maniere van databasis verwerking kan jy gevra word om die volgende te demonstreer:
  - Databasis manipulasie prosedures soos verander, byvoeg en skrap.
  - Navraag (*Querying*) prosedures:
    - Onttrek rekords wat aan ’n sekere stel kriteria voldoen,
    - Vertoon geselekteerde kolomme
    - Sorteer die inligting
    - Onttrek data uit twee tabelle
    - Genereer nuwe inligting: berekende velde
    - Inligting wat saamgevoegde (aggregate) funksies gebruik
- In 2.2 word basiese Delphi kodering vaardighede soortgelyk aan wat gevra word in Vraag 1, van jou verwag. Spesifiek geneste lusse om data uit twee tabelle te onttrek.

VRAAG 2: DATABASISPROGRAMMERING

Hierdie afdeling bestaan uit twee vrae. Die volgende belangrike aantekeninge is op albei vrae van toepassing:

- Jy mag **NIE** op enige manier die gegewe data verander of daaraan byvoeg nie.
- Goeie programmeringstegnieke moet toegepas word by die kodering van jou oplossings.
- **GEEN punte word vir *hard coding* toegeken nie**. Gebruik beheerstrukture en veranderlikes waar nodig.

Die databasis, **DB.mdb**, bevat inligting van bestuurders en hul eksotiese en skaars motors. Die databasis bevat twee tabelle, naamlik **tblDrivers** en **tblCars**.

Die **tblDrivers**-tabel is saamgestel met die volgende velde:

VELD	DATATIPE	BESKRYWING
DriverNo (PK)	Text	Unieke nommer vir elke bestuurder.
DriverName	Text	Bestuurder se noemnaam.
DriverSurname	Text	Bestuurder se van.
DriverTel	Text	Die bestuurder se telefoonnommer.
DriverDoB	Date/Time	Geboortedatum van die bestuurder.

Voorbeeld van die data in die **tblDrivers** tabel:

DriverNo	DriverName	DriverSurname	DriverTel	DriverDoB
D101	Darsie	Semper	+17 43 349 808	1962/07/14
D102	Simonne	Aarons	+15 80 130 295	2004/06/16
D103	Merwyn	Lille	+19 82 227 171	1967/09/08
D104	Norri	Garlick	+10 54 951 640	1991/05/31
D105	Tess	Barnewille	+10 60 229 780	1995/12/26
D106	Ewart	Scouller	+12 86 390 741	1987/12/27
D107	Harcourt	Nashe	+11 38 771 593	2002/08/30
D108	Kennan	Ebbitt	+16 29 352 293	1978/12/09

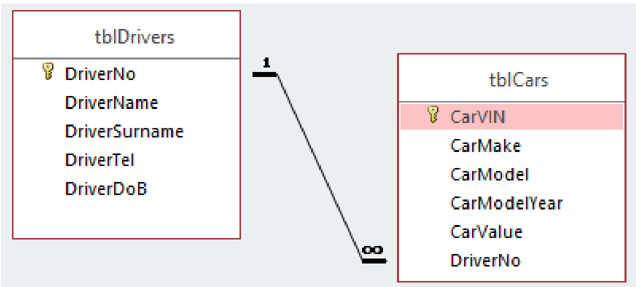
Die **tblCars**-tabel is saamgestel met die volgende velde:

VELD	DATATIPE	BESKRYWING
CarVIN (PK)	Text	Die motor se unieke nommer.
CarMake	Text	Die fabrikaat van die motor.
CarModel	Text	Die model van die motor.
CarModelYear	Number	Die jaar waarin die motor vervaardig is.
CarValue	Currency	Die huidige waarde van die motor.
DriverNo (FK)	Text	'n Verwysing na die bestuurder se unieke nommer.

Voorbeeld van die data in die **tblCars**-tabel:

CarVIN	CarMake	CarModel	CarModelYear	CarValue	DriverNo
137FA84301E175026	Saturn	S-Series	1997	R4 364 501.00	D126
19UUA65525A102408	Ford	Expedition EL	2007	R204 245.00	D125
19UUA66267A951657	Ford	Transit Connect	2012	R4 689 870.00	D139
19UUA662X6A190551	Subaru	Forester	2012	R4 458 195.00	D125
19UUA75537A059383	Cadillac	Fleetwood	1993	R4 784 280.00	D121
19UUA9F58CA593144	Ferrari	F430 Spider	2006	R1 757 154.00	D123
19UYA41673A238857	Ford	Aspire	1994	R3 728 543.00	D121
19UYA42793A209125	Volvo	V40	2000	R1 116 123.00	D134

Die volgende een-tot-meer-verhouding met *referential integrity* bestaan tussen die twee tabelle in die databasis:



NOTA:

- Die verbindingskode is verskaf.
- Die databasis is beskerm met 'n wagwoord - daarom sal jy nie direk toegang tot die databasis kry nie.
- Wanneer die knoppie **btnDBRestore** gekliek word, sal die data in die databasis herstel word na die oorspronklike data

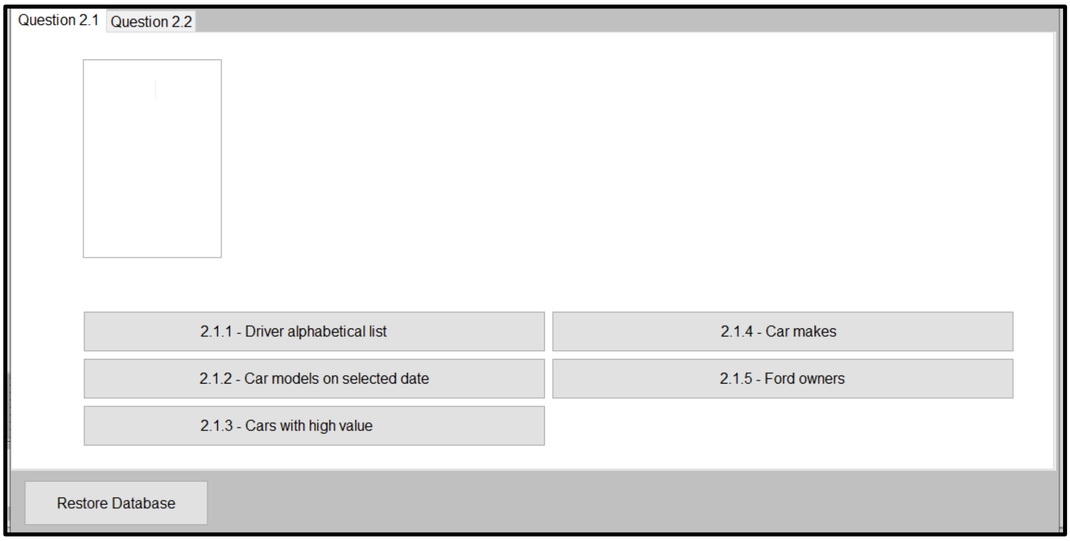
Doen die volgende:

- Maak die program in die Question2-lêergids oop en voer dit uit. Die program het tans beperkte funksionaliteit.
- Voltooi die kode vir elke vraag soos beskryf in VRAAG 2.1 en VRAAG 2.2

Die program het 'n grafiese-koppelvlak met twee oortjies (*tab-sheets*) genaamd **Question 2.1** en **Question 2.2**.

VRAAG 2.1 STRUCTURED QUERY LANGUAGE (SQL)

Selekteer *tab sheet* **Question 2.1**, wat die volgende koppelvlak vertoon:



Voltooi die SQL-kode om aan die vereistes te voldoen wat gespesifiseer is in **VRAAG 2.1.1** tot **VRAAG 2.1.5**.

BASIESE STAP-VIR-STAP GIDS OOR HOE OM SQL VRAE TE BENADER

1. *Identifiseer die sleutelwoorde (soos: "vertoon", "sorteer", "rangskik", "voeg by", "opdateer", "verwyder/skrap" ens.) om vas te stel wat bereik met word.*

2. *Breek die vraag op in die verskeie dele van die SQL-stelling, bv.*
   a. *bepaal wat gekies (**SELECT**) moet word,*
   b. *bepaal die tabel(le) waaruit (**FROM**) die inligting gehaal moet word,*
   c. *watter voorwaardes moet aan voldoen word (**WHERE**)/ hoeveel detail is nodig/ watter rekords moet spesifiek vertoon word/ waaraan moet die rekord voldoen om ingesluit te word*

   .
3. *Word iets bereken of formateer?*
   a. *Indien ja, identifiseer die berekende velde en identifiseer die velde wat formateer moet word*
   b. *kyk of daar aliasse is en gebruik die **AS** klousule*
   c. *as dit 'n berekende veld is, herhaal die berekening in die **WHERE** klousule want die alias kan nie in die WHERE klousule gebruik word nie*
   d. *moenie vergeet om die **GROUP BY** in te sluit as daar 'n saamgevoegde (aggregate) funksie gebruik word nie, bv.*
      SELECT COUNT (*) FROM tblCars GROUP BY CarModel;
   e. *kyk ook of groepering gebaseer is op voorwaardes/kriteria. **HAVING** sal dan nodig wees.*
4. *Vertoon slegs velde wat gevra word.*
5. *Bou die SQL stelling stukkie vir stukkie op en verstel soos jy aangaan.*
6. *Moenie dit oop los nie! Maak seker jy kodeer iets. Geen kode=geen punte.*

2.1.1 **KNOPPIE [2.1.1 – Driver alphabetical list]**

Vertoon AL die besonderhede van die bestuurders in die **tblDrivers**-tabel, alfabeties gesorteer volgens die van van die bestuurder.

Voorbeeld afvoer van eerste vier rekords:

DriverNo	DriverName	DriverSurname	DriverTel	DriverDoB
D102	Simonne	Aarons	+15 80 130 295	2004/06/16
D141	Zora	Abelevitz	+18 50 871 480	2001/04/06
D138	Cort	Bantick	+12 99 508 640	1942/01/11
D105	Tess	Barnewille	+10 60 229 780	1995/12/26

(4)

VRAAG 2.1.1 (Gr 12 DBE Handboek Unit 4.1 bladsy 111)

1. *Begin deur die sleutelwoorde te soek en dit op te breek in take wat jy moet doen.*
   a. *Dit help regtig om die woorde te onderstreep en die verskillende dele te nommer totdat dit later vanselfsprekend word. Byvoorbeeld:*
   b. *(1) Vertoon ALL die besonderhede van die bestuurders in die (2) **tblDrivers** tabel, (3) alfabeties gesorteer volgens die van van die bestuurder.*
2. *Nou het jy vasgestel dat daar 3 sleuteldele is in die stelling en kan jy begin om dit op te bou:*
   a. *Deel 1: SELECT **
   b. *Deel 2: FROM tblDrivers*
   c. *Deel 3: ORDER BY DriverSurname ASC*
3. *Voeg al hierdie dele saam en kontroleer jou antwoord. Wanneer die stellings meer ingewikkeld raak sal jy regstellings en aanpassings maak.*

2.1.2 **KNOPPIE [2.1.2 – Car models on selected date]**

Vertoon die fabrikaat (*make*) en model van al die motors wat in 1959 vervaardig is.

Voorbeeld van afvoer:

CarMake	CarModel
BMW	600
Austin	Mini

(3)

VRAAG 2.1.2 (Gr 12 DBE Handboek Unit 4.2 bladsy 113)

1. *Benader hierdie vraag net soos jy by die vorige een gedoen het. Kom ons begin deur die dele te identifiseer en dan kyk jy of jy die res op jou eie kan doen.*
2. *Die dele sal dan wees: (1)Vertoon die fabrikaat (make) en model van al die (2) motors wat in 1959 vervaardig is.*
   ***Wenk:** Die tabelnaam word nie in die vraag gegee nie, moenie daarvan vergeet nie!*

2.1.3 **KNOPPIE [2.1.3 - Cars with high value]**

Vertoon die fabrikaat, model en waarde van die motors met 'n waarde van R 2 000 000 of meer. Formateer die motor se waarde as geldeenheid en vertoon die veldnaam as **CurrentValue**.

Voorbeeld afvoer van die eerste vier rekords

CarMake	CarModel	CurrentValue
GMC	Sierra 2500	R2 524 597.00
Dodge	Ram Van B250	R2 327 265.00
GMC	Yukon XL	R3 025 337.00
Chevrolet	Corvette	R2 347 406.00

(6)

VRAAG 2.1.3 (Gr 12 DBE Handboek Unit 4.3 bladsy 129)

1. *Hierdie een is 'n bietjie meer uitdagend, dus help ons jou bietjie meer hier.*
2. *Begin, soos altyd, om die sleutelwoorde en dele te identifiseer:*
   a. *(1)Vertoon die fabrikaat, model en waarde van die motors met 'n (2)waarde van R 2 000 000 of meer. . (3) Formateer die motor se waarde as geldeenheid en (4) vertoon die veldnaam as **CurrentValue**.*
   b. *Daar is 4 duidelike dele in hierdie vraag. Kom ons skryf dit neer...*
   c. *Deel 1: SELECT CarMake, CarModel*
      Deel 2: WHERE CarValue >= 2000000
      Deel 3: Format(CurrentValue,"currency")
      Deel 4: AS CurrentValue
3. *Bogenoemde dele is nie in die regte volgorde nie en daar is nog foute, maar kom ons sit dit saam en kyk wat ons het:*
   a. *SELECT CarMake, CarModel, Format(CurrentValue,"currency") AS CurrentValue FROM tblCars WHERE CarValue >=2000000*
4. *Goeie begin, maar daar is nog klein foutjies? Kan jy hulle raaksien?*
   ***Wenk**: Kyk na die name van die velde*

2.1.4 **KNOPPIE [2.1.4 - Car makes]**

Vertoon 'n lys van fabrikate (**CarMake**) van die motors, asook die aantal kere wat dit in die **tblCars**-tabel voorkom.

Voorbeeld van afvoer:

CarMake	Amount
Acura	14
Aston Martin	3
Audi	33
Austin	2

(3)

VRAAG 2.1.4 (Gr12 DBE Handboek Unit 4.4 bladsy 133-136)

Kan jy hierdie een op jou eie doen? Wenk: 'n sleutelaspek is die "aantal kere". Onthou om na die wenke aan die begin van Vraag 2 te gaan kyk as jy nog raad nodig het.

2.1.5 **KNOPPIE [2.1.5 - Ford owners]**

Vertoon die naam en van van die bestuurder en die model van die motor van al die bestuurders wat 'n Ford-motor besit.

VOORBEELD AFVOER van eerste vier rekords:

DriverName	DriverSurname	CarModel
Sharline	Bodycomb	F250
Norri	Garlick	Crown Victoria
Ira	Castagneto	Explorer
Ira	Castagneto	F250

(5)

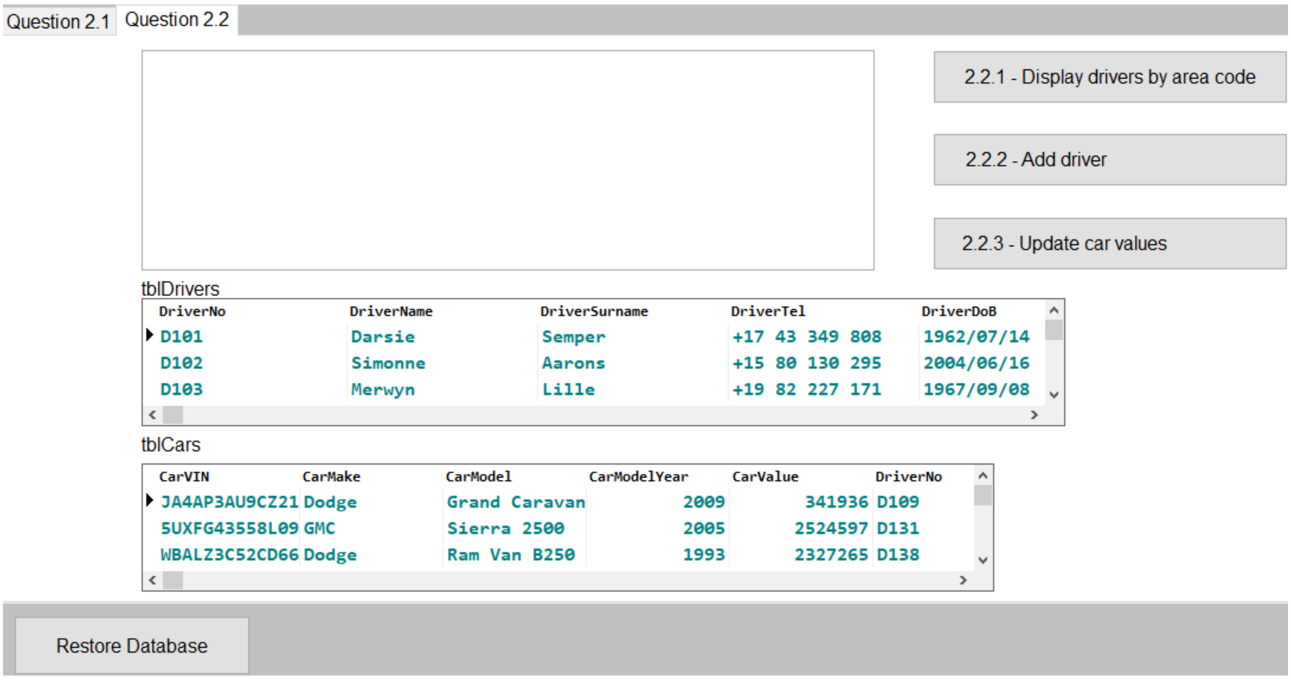
VRAAG 2.1.5 Gr 12 DBE Handboek Unit 4.6 bladsy 144

1. *Daar is min instruksies in hierdie vraag, maar daar is tog verskeie klein deeltjies.*
2. *Weereens gaan jy eers die dele identifiseer en hulle in die korrekte volgorde plaas.*
3. *Onthou dat die velde uit twee aparte tabelle kom.*
   a. *Jy kan dit doen deur "FROM tblDrivers d, tblCars c"*
   b. *en dan verwys jy na die spesifieke veld deur die letter vooraan te voeg "d.DriverNo"of "tblDrivers.DriverNo"*
   c. *onthou dat die skakel tussen die twee tabelle die primêre sleutel van die een tabel en sy ooreenstemmende vreemde (foreign) sleutel in die ander tabel is.*

[21]

VRAAG 2.2: DATABASIS MANIPULASIE (GEEN SQL)

Kies die oortjie (*tab sheet*) **Question 2.2**, wat die volgende gebruikerskoppelvlak vertoon:



NOTA:

- GEEN punte word toegeken vir die gebruik van SQL-stellings vir **VRAAG 2.2** nie.
- Die name van die tabelle wat in jou kode gebruik moet word, moet **tblDrivers** en **tblCars** wees, wat **TADOTable**-objekte is wat aan die databasis gekoppel is.

Voltooi die kode om aan die vereistes te voldoen soos gespesifiseer in **VRAAG 2.2.1** tot **VRAAG 2.2.3**.

GLOBALE WENKE OOR HOE OM DIE GEEN-SQL VRAAG TE BENADER

1. *Onthou dat jy die volgende stukkies kode nodig gaan hê <u>om deur die rekords van die tabel te lees</u> – of jy nou vir 'n rekord soek, rekords opdateer en of jy nou 'n rekord wil verwyder:*
   a. *tbl.First voor die while loop*
   b. *while NOT tbl.EOF Do*
   c. *tbl.Next in die while loop*

2. *Wanneer jy <u>'n nuwe rekord moet byvoeg</u> gaan jy die volgende stukkies kode benodig:*
   a. *tbl.Last*
   b. *tbl.Insert OF tbl.Append (dit hang af van jou voorkeur)*
   c. *tbl.Post*

2.2.1 **KNOPPIE [2.2.1 - Area code]**

Skryf kode om die **naam**, **van** en **geboortedatum** te toon van al die ***bestuurders*** wie se **telefoonnommer** met **+15** begin.

Die data moet in die **redQ2**-komponent in netjiese kolomme vertoon word, soos in die voorbeeld van uitvoer.

NOTA: Kode vir die opskrifte word voorsien.

Voorbeeld van afvoer vir die eerste vier lyne van afvoer:

Name	Surname	Date of Birth
Simonne	Aarons	2004/06/16
Sabra	Mumbeson	2014/11/07
Web	Maven	1955/03/23
Maddy	McReynold	1964/04/17

(7)

Vraag 2.2.1 (Gr 11 DBE Handboek Eenheid 8.5 bladsy 193)

Ontleding Vraag 2.2.1

***Wenk** : Dit help om die sleutelwoorde te onderstreep of uit te lig (highlight) soos hierbo geïllustreer.*

Vra jouself die volgende af:

- ***Vraag :** Wat moet ek vertoon?*
- ***Antwoord :** Die* naam, van en geboortedatum. *Al drie hierdie sleutelwoorde is velde in die tblDriver tabel.*
- ***Vraag**: Wanneer moet ek die velde vertoon?*
- ***Antwoord:** As die telefoonnommer veld in tblDriver met +15 bv. begin*
- ***Vraag :** Watter metode gaan ek gebruik om vas te stel **of** die telefoonnommer met **+15** begin?*
- ***Antwoord:** Maak van 'n ingeboude funksie gebruik om **die eerste 3 karakters** van die telefoonnommer te onttrek. Evalueer en toets of die telefoonnommer met +15 begin.*

Hoe om met Vraag 2.2.1 te begin:

- *Onthou om die **opskrif** soos hierbo gewys word in **die redQ2 komponent** te vertoon.*
- *Gebruik 'n **lus**struktuur om **elke drywer in die tblDriver** te evalueer en in die lus 'n **if**-stellings om te toets of die nommer met **+15** begin.*
- *Vertoon al die **toepaslike inligting** van die **drywer** van wie die telefoonnommer met +15 begin. Vertoon in die if-stelling **OF** voeg 'n afvoerstring saam wat aan die einde van die lus vertoon kan word.*
- ***Onthou** die algemene riglyne wanneer jy deur elke rekord in die tabel lees/soek.*

2.2.2 **KNOPPIE [2.2.2 - Add driver]**

Skryf kode om 'n rekord by te voeg by die **tblDrivers**-tabel.

Die data van die bestuurder wat bygevoeg word, word hieronder gegee:

Driver no:	D150
Driver name:	Jule
Driver surname:	Orris
Driver telephone:	+51 87 580 3783
Driver date of birth:	1972/11/25

Voorbeeld van die laaste paar rekords in **tblDrivers** nadat die rekord vir Jule bygevoeg is:

driverNo	driverName	driverSurname	driverTel	driverDOB
D146	Sharline	Bodycomb	+11 95 974 204	2009/12/07
D147	Halli	Huggens	+18 61 562 337	1991/09/28
D148	Ailsun	Chrstine	+15 12 259 324	2006/07/09
D149	Carny	Gilbard	+16 76 979 991	2011/12/12
D150	Jule	Orris	+51 87 580 373	1972/11/25

Vraag 2.2.2 (Gr 11 DBE Handboek Unit 8.4 bladsy 182)

Ontleding Vraag 2.2.2: *Vra jouself die volgende vrae af:*

- ***Vraag :****Wat moet die kode doen in hierdie vraag?*
- ***Antwoord : Die kode moet byvoeg by die tblDriver tabel.***
- ***Vraag****: Watter kodering moet ek doen by by te voeg by die tabel?*
- ***Antwoord: Ek gebruik 'n byvoeg algoritme om tot die tabel by te voeg.***

Hoe om met Vraag 2.2.2 te begin:

- ***Onthou*** *die algemene riglyne oor hoe data by die databasis bygevoeg word.*
- ***Wenk:*** *As data bygevoeg word, is die data tipe belangrik bv. as die veld 'n teksveld is, moet die kode* ***tbl[FieldName]:= ""****wees; as die veld 'n getal moet wees sal die kode* ***tbl[FieldName]:= 8*** *wees;*

2.2.3 **KNOPPIE [2.2.3 - Update value]**

Skryf kode om die **waarde van al die motors** te **verhoog** volgens die onderstaande tabel:

Jaar van vervaardiging van model (CarModelYear)	% waarvolgens waarde verhoog moet word
Voor 2010	15%
2010 en later	10%

Voorbeeld van die laaste paar rekords in **tblCars** nadat die waardes verander is:

CarVIN	CarMake	CarModel	CarModelYear	CarValue	DriverNo
1GTN1UEH1EZ11	Kia	Spectra	2003	3523156.1	D142
19UUA662X6A19	Subaru	Forester	2012	4904014.5	D125
5J6TF1H39BL27	Kia	Rio	2007	1961477.95	D114

(9)
[19]

Vraag 2.2.3 (Gr 11 DBE Handboek Unit 8.5 bladsy 197)

Ontleding Vraag 2.2.3 : *Vra jouself die volgende vrae af:*

- ***Vraag :*** *Wat gaan met die data gebeur as ek die waarde van die motors verhoog?*
- ***Antwoord: Die data gaan verander. WENK: In watter staat moet die databasis wees om veranderinge te kan maak?***
- ***Vraag****: Watter veld moet toeneem?*
- ***Antwoord: CarValue***
- ***Vraag:*** *Lys die* ***twee*** *voorwaardes waarop die data moet verander:*
- ***Antwoord****:*
    - *As die jaar van vervaardiging van die model voor 2010 is*
    - *As die jaar van vervaardiging van die model 2010 of later is*
- ***Vraag :*** *Hoe sal die data verhoog word volgens die voorwaarde?*
- ***Antwoord: CarValue = CarValue + (CarValue * % waarde waarmee verhoog word).***

Hoe om met Vraag 2.2.3 te begin:

- *Onthou dat die waarde by elke rekord in die tabel verhoog moet word.*
- *Afhangend van die jaar van vervaardiging sal die CarValue met 10% of 15% verhoog word.*
- ***Onthou*** *die algemene riglyne wanneer jy elke rekord in die tabel moet opdateer/verander.*

TOTAAL AFDELING B: 40

AFDELING C: OBJEK-GEORIËNTEERDE PROGRAMMERING

Metodes (functions , procedures) en OOP vaardighede ontwikkel in graad 11 en 12.
Voltooiing van die klasdefinisie (Vraag 3.1) en instansiëring en gebruik van 'n objek (Vraag 3.2)
Hierdie vaardighede en konsepte is:

- Die struktuur van 'n klasdefinisie.
- Die klas as 'n data tipe, sy eienskappe (attribute/velde) en wat dit kan doen (metodes)
- Verskil tussen private en publieke afdelings.
  - Velde in die private gedeelte verklaar is versteek
  - Metodes in die publieke gedeelte is sigbaar en vorm die koppelvlak na die klas.
- Metode *headers*, die sleutelwoorde *functions* en *procedures*, *parameters* en *return type*
- Die gebruik van die gereserveerde woord *Result* om die waarde van 'n funksie terug te stuur.
- Die doel van die konstruktor, toString metode, *getters*, *setters* en hulp (*auxiliary*) metodes.
- Gebruik die konstruktor metode om 'n objek te skep.
- Die punt.notasie wat toegepas word wanneer 'n objekmetode geroep word.
- Deurstuur van parameters: die aantal parameters, data tipe en volgorde.
- Die verskil tussen die roep van 'n metode binne-in die klas en van buite die klas.

VRAAG 3: OBJEK-GEORIËNTEERDE PROGRAMMERING

SCENARIO
'n Onderneming wat daarin spesialiseer om mariene-onderdele vir bote en skepe te verf benodig sagteware om die prys van verfwerk wat hulle moet voltooi, te bereken.

Die onvolledige projek **Question3_P** word in jou **Question3**-eksamenlêergids voorsien. Dit bevat:

- Die onvolledige hoofvorm-eenheid genaamd **Question3_U.pas**
- Die onvolledige objekklas genaamd **PaintJob_U.pas**

Die program het tans geen funksionaliteit nie.
Voltooi die kode vir elke afdeling van VRAAG 3 soos gespesifiseer in **VRAAG 3.1** en **VRAAG 3.2** hieronder.

3.1 'n Onvolledige klas genoem **PaintJob_U.pas** is voorsien. Voltooi die kode in die gegewe klas genaamd **TPaintJob** soos beskryf in **VRAAG 3.1.1** tot **VRAAG 3.1.4**.

Die volgende eienskappe (attribute) word vir die **TPaintJob**-klas voorsien:

EIENSKAPPE (ATTRIBUTE)	BESKRYWING
fCustomer	Volle naam (voornaam en van) van die kliënt wat die verfwerk wil laat doen.
fPaintType	Tipe verf wat benodig word.
fSurfaceArea	Grootte van die oppervlak wat geverf moet word in m^2.
fCode	Unieke kode vir die werk wat gedoen word.

NOTA: 'n Voltooide **toString** metode is aan jou voorsien in die **TPaintJob** klas.

3.1.1 Skryf kode vir 'n konstruktor-metode wat die volle naam van die kliënt, die tipe verf wat gebruik gaan word en die oppervlakte wat geverf moet word as parameters sal ontvang. Ken hierdie waardes aan die onderskeie attribute toe.

(4)

VRAAG 3.1.1 (Gr 12 DBE Handboek Unit 4.1 bladsy 111)

- *Die konstruktor-metode is 'n spesiale metode wat slegs een keer gebruik word met die uitsluitlike doel om die instansie van die klas in die geheue te skep (instansieer). Spesifieke waardes (uit die Form-klas) word as parameters deurgestuur na die attribute van die klas.*
- *Wees versigtig met die data tipes van jou parameters: hulle moet dieselfde wees as die van die attributes waaraan hulle toegeken moet word. Ons benoem die veranderlikes volgens die standaard Delphi konvensie, d.w.s. s-voorvoegsel vir karakterstring (string); r-voorvoegsel vir desimale getalle (real/double); i-voorvoegsel vir heelgetalle (integer)*
- *Die konstruktor word in die publieke gedeelte verklaar omdat ons dit van BUITE die klas roep.*

3.1.2 Skryf kode vir 'n metode genaamd **calculateCost** wat 'n **heelgetal waarde** (Integer) as parameter ontvang en **die koste van die verfwerk** terugstuur.

- Die heelgetal waarde wat ontvang word is die aantal lae wat geverf moet word.
- Vermenigvuldig die aantal lae met die oppervlakte om die totale oppervlak wat geverf moet word, te bereken.
- Die totale berekende oppervlakte moet vermenigvuldig word met die koste van die tipe verf per vierkante meter (soos in die tabel hieronder getoon).

TIPE VERF	KOSTE PER VIERKANTE METER (m^2)
Vinyl	R1254.60
Polyurethane	R748.35
CoalTar	R322.62

Bereken die koste van die verfwerk en stuur dit terug as resultaat.

(8)

VRAAG 3.1.2 (Gr 12 DBE Handboek Unit 2.2 bladsy 73) calculateCost-metode

'n Metode wat 'n waarde moet teruggee kan net 'n function wees.

- *Functions MOET verklaar word met 'n return tipe na die naam van die function*
  function .... :string;
  function ..... :integer; ens;
  Kosteberekeninge lewer gewoonlik desimale getalle as resultaat, dus sal die return tipe ..... wees?
- *As 'n funksie parameters ontvang, word die parameters, gevolg deur die return tipe, in hakies verklaar, bv.*
  function calcDays(dFrom,dTo:TDateTime):integer;
  function getGender(sID:string): string;
  In die vraag hierbo verwys die heelgetal parameter na die aantal lae wat geverf gaan word.
- *Onthou om CTRL+SHIFT+C te klik met jou muis langs die verklaring van die metode om die metode se raamwerk te skep.*
- *Functions stuur resultate terug d.m.v. die gereserveerde veranderlike Result. Dit wat aan die Result veranderlike toegeken word, word teruggestuur na die metode van waar die funksie geroep word. Maak seker jy sluit Result:=...... in, in jou function, EN dat die korrekte datatipe daarmee assosieer word.*
  In ons vraag, word die basiese koste (Result) bereken met die formule :
  koste = oppervlak x aantalLae
  Ons moet eers 'n seleksie struktuur gebruik om te toets watter tipe verf gebruik gaan word, want elk van die 3 tipes verf het elkeen sy eie koste/m^2 (soos in die tabel aangedui).
  Dus: geneste if-else-if of 'n case stelling kan gebruik word.

3.1.3 Skryf kode vir 'n metode genaamd **generateCode** wat 'n **string** stoor in die **fCode**-attribuut wat saamgestel is in die volgende formaat:

%%$$**

Die beskrywing van die verskillende afdelings in die formaat van die kode is soos volg:

- %% stel die eerste **TWEE** letters van die verftipe in **KLEINLETTERS** voor.
- $$ stel die **EERSTE** en **LAASTE** letters van die kliënt se die volle naam voor.
- ** stel **TWEE** lukrake (*random*) getalle van 0 tot 9 voor (albei ingesluit).

Voorbeeld van 'n gegenereerde kode met die volgende attribuutwaardes:

TIPE VERF:	Polyurethane
VOLLE NAAM VAN KLIËNT:	Floyd Khanyiso
GEGENEREERDE KODE:	poFo54

Voorbeeld van 'n gegenereerde kode met die volgende attribuutwaardes:

TIPE VERF:	Coaltar
VOLLE NAAM VAN KLIËNT:	Barry Davids
GEGENEREERDE KODE:	coBs20

Nota: Jou afvoer kan verskil van die voorbeelde hierbo aangesien lukrake getalle gegenereer word.

(8)

fCode is 'n attribute van die klas. Met hierdie PROCEDURE, moet 'n waarde geskep/gegenereer word vir hierdie attribute. Eers nadat hierdie PROCEDURE uitgevoer het, kan die FUNCTION getCode (in die volgende vraag) geroep word in die vorm-klas om fCode terug te stuur. Hoekom is generateCode 'n procedure en nie 'n function nie? Daar is twee redes, fCode is sigbaar in hierdie metode, hulle is beide in dieselfde klas/.pas lêer en daar word nêrens in die vraag aangedui dat 'n waarde teruggestuur moet word nie.

Ons maak hier gebruik van ons kennis van STRING FUNKSIES (wat ons in gr10 gedoen het) om die kode te skep:

Eerste TWEE:

- *kopieer die eerste TWEE karakters van fPaintType (copy(fPaintType,1,2)*
- *skakel dit om na KLEINLETTERS (onthou Lower() word vir 'n enkele karakter gebruik en LowerCase() vir karakterstringe) +*

Volgende TWEE:

- *vollenaam[1] vir die eerste karakter PLUS*
- *vollenaam[lengte van vollenaam] om die laaste karakter te neem +*

Laaste TWEE:

- *twee lukrake getalle. Wees versigtig: om 0 EN 9 in te sluit, gebruik Random(10) OF RandomRange(0,10). Maak seker dat die Math unit ingesluit is by Uses indien jy RandomRange gebruik*
- *Moenie vergeet om om die IntToStr-funksie te gebruik om die lukrake getalle by te voeg by die eerste gedeeltes nie.*

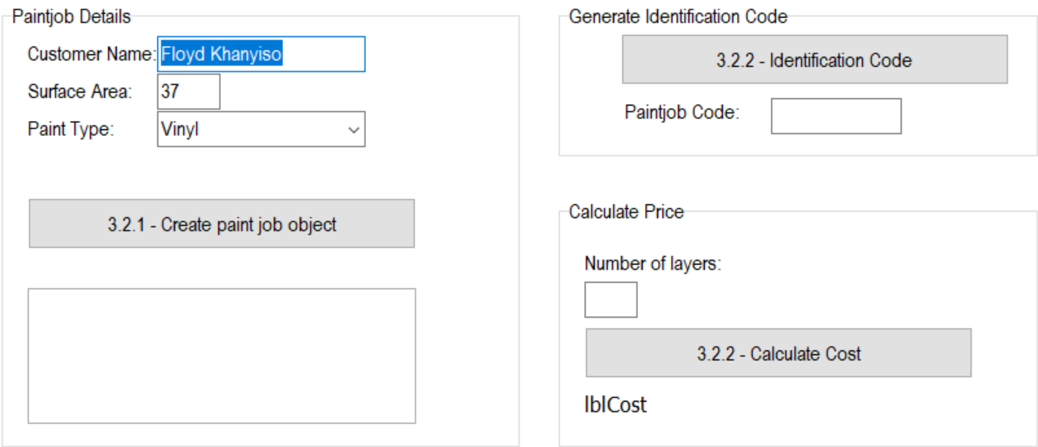
ONTHOU OOK: Hierdie is 'n PROCEDURE, daar sal dus geen resultaat of return tipe wees nie.

3.1.4 Skryf kode vir 'n metode genaamd **getCode** wat die unieke kode van die verftaak wat in die **fCode**-attribuut (gegenereer in **VRAAG 3.1.3**) as 'n **string**, sal terugstuur.

(2)

Weereens: 'n metode wat 'n waarde moet teruggee kan slegs 'n function wees. Standaard accessor(getter) metode vir die attribute fCode.

'n Onvolledige vormklas **Question3_U** is aan jou voorsien met die volgende grafiese gebruikerskoppelvlak:



3.2 Voltooi die vormklas deur die volgende te doen:

Gr 12 DBE Handboek Hoofstuk 2 bladsy 70-74)

Neem kennis van die volgende 2 belangrike dinge voordat jy begin om te kodeer:

- *Die naam van die unit sonder die uitbreiding waarin die objekklas definieer word moet ingesluit word in die Uses gedeelte heel boaan die Form-klas. Dit word gewoonlik gegee in die eksamen, maar wees bewus daarvan sodat jy die konneksie met die objekklas kan verstaan..*
  Uses Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ComCtrls,PaintJob_U,Math;
- *Dan is dit ook baie belangrik om te kyk waar en hoe die objek verklaar word. Dit word gewoonlik in hierdie formaat gegee: objName : className;*
  var objPaintJob: TPaintJob;

In die eerste vraag word die gebruikerstoevoer gelees. Jy mag die veranderlikes verklaar met die korrekte tipes en gebruik daarvoor of jy kan kies om nie veranderlikes te gebruik nie. Gebruik veranderlikes as jy onseker is.

3.2.1 **KNOPPIE [3.2.1 – Create paint job object]**

Die gebruiker moet die volle naam van die kliënt, die oppervlak van die area wat geverf moet word en die tipe verf wat benodig word, intik in die komponente wat voorsien word.

Die volgende toevoer data is voorsien in die ontwerp van die grafiese gebruikerskoppelvlak om die program te toets:

- Kliëntnaam: Floyd Khanyiso
- Oppervlak: 37
- Tipe verf: Vinyl

'n Globale **TPaintJob** veranderlike genaamd **objPaintJob** is gegee.

Skryf kode om die volgende te doen:

- Gebruik die gegewens wat deur die gebruiker ingevoer is om die **objPaintJob**-objek te instansieer.

Om die objek te instansieer is dit belangrik om die volgende reg te doen:

- *objName aan die linkerkant van die := teken,*
- *die className.create met*
- *die parameter lysie (die regte aantal parameters) met die korrekte datatipes, in die korrekte volgorde soos in die definisie van die konstruktor gespesifiseer) bv.:*
  ObjName = className.create(input variable1, input variable2,...)

- Gebruik die **toString**-metode om die besonderhede van die objek in die *rich edit* afvoerarea **redQ3** te vertoon.

(7)

Die toString metode is 'n funksie wat 'n afvoerstring skep in 'n spesifieke formaat.
Onthou om jou objekveranderlike te gebruik om die Form-klas in staat te stel (bou 'n brug na die objekklas) om toegang te verkry tot die metodes (funksies/prosedures) van die objekklas.
Omdat die toString funksie 'n afvoerstring skep, moet dit geroep word in 'n afvoerstelling om die gewenste afvoer (die attribute van die objek) te kan vertoon.

3.2.2 **KNOPPIE [3.2.2 - Identification code]**

Skryf kode om die volgende te doen:

- Roep die **generateCode**-metode om 'n unieke kode vir die verftaak te genereer..

- *Die hulp metode (auxiliary), die funksie **generateCode** is in Vraag 3.1.3 geskep.*
- *Dit het geen parameters en die resultaat is 'n string data tipe omdat dit 'n kode skep.*
- *Die **generateCode** funksie moet geroep word om die unieke kode vir die verftaak te skep.*
- *Moenie vergeet daarom met die korrekte objeknaam te roep nie. Dit sal verseker dat toegang na die funksie in die objekklas verkry word.*

- Vertoon die saamgestelde identifikasiekode in die redigeerblokkie (edit box) **edtCode** met behulp van die toepaslike metode.

- *Onthou om die kode toe te ken aan die regte eienskap (property) van die TEdit edtCode:*
  (i.e. edtCode.text:= ...)

VOORBEELD VAN AFVOER:

Voorbeeld van afvoer as die toetsdata wat as toevoer waardes in **VRAAG 3.2.1** verskaf is, gebruik is:

(3)

3.2.3 **KNOPPIE [3.2.3 – Calculate Cost]**

Die gebruiker moet die aantal lae geverf moet word selekteer in die combobox **cmbLayers**.

NB! Dit is 'n TEdit en NIE 'n TComboBox komponent NIE.

Kode is verskaf wat 'n konstante veranderlike met die naam iMaxLayers verklaar, wat die maksimum aantal lae is waarmee 'n verftaak geverf kan word. Dit is tans op die waarde van 7 ingestel.

Skryf kode om die volgende te doen:

- Onttrek die aantal lae wat gekies is uit die komponent **edtLayers**.

Lees die aantal lae verf benodig uit die invoerkomponent (TEdit) for Vraag 3.2.3. Maak seker jy gebruik die regte eienskap (property) m uit 'n TEdit te lees (.text)

– Bepaal of die onttrekte aantal lae bygevoeg kan word sonder dat die maksimum aantal lae wat gebruik kan word oorskry word.

Maak seker dat die maksimum aantal lae verf (die konstante wat net onder die objekveranderlike verklaar is) nie oorskry word nie. Jy gaan 'n seleksiestruktuur (if-stelling) nodig hê om dit te doen..

– Indien die aantal lae nie die toegelate maksimum oorskry nie, roep die toepaslike metode om die koste van die verftaak te bereken.

Jy gaan nou dus die hulp metode (auxiliary method), die funksie CalculateCost, wat jy in die objekklas in vraag 3.1.2 geskep het, gebruik om die totale koste te bereken.
Maak weereens seker jy gebruik die objekveranderlike om toegang te verkry tot die metodes in die objekklas (bv. objNaam.metodeNaam).

– Vertoon die totale prys in die **lblCost**-komponent, geformateer tot **TWEE** desimale plekke.

Maak seker dat die data tipe van die totale koste omgeskakel word na die regte afvoer datatipe om die totale prys te vertoon. Kyk mooi na die omskakeling data jy die regte eenheid (currency) gebruik in die Format of FloatToStrF funksie.

- Indien die aantal lae die maksimum oorskry, vertoon 'n geskikte foutboodskap op die **lblCost**-komponent.

Weereens moet jy seker maak dat die korrekte eienskap van die TLabel komponent gebruik word om die afvoer te vertoon.

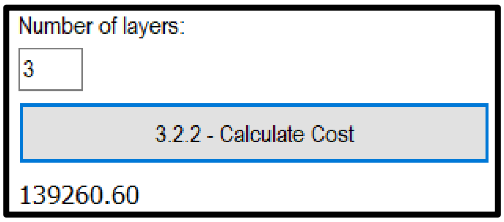
Vir elk van die toets gevalle druk **KNOPPIE [3.2.1 Create paint job object]** dan **KNOPPIE [3.2.3 – Calculate Cost]**

'n Baie belangrike fase in die ontwikkeling van programmatuur is om te toets. Dit moet met groot sorg gedoen word.

VOORBEELD 1: Toetsdata:

OPPERVLAKTE:	37
TIPE VERF:	Vinyl
AANTAL LAE:	3

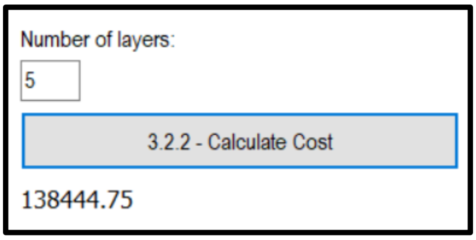
Voorbeeld van afvoer op **lblCost** gebaseer op die toetsdata hierbo:



VOORBEELD 2: Toetsdata:

OPPERVLAKTE:	37
TIPE VERF:	Polyurethane
AANTAL LAE:	5

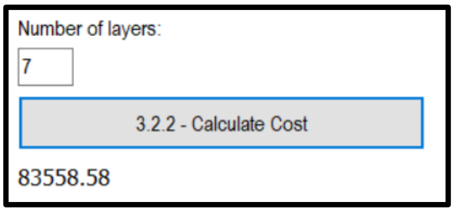
Voorbeeld van afvoer op **lblCost** gebaseer op die toetsdata hierbo:



VOORBEELD 3: Toetsdata:

OPPERVLAKTE:	37
TIPE VERF:	CoalTar
AANTAL LAE:	7

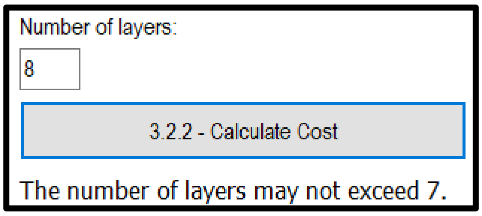
Voorbeeld van afvoer op **lblCost** gebaseer op die toetsdata hierbo:



VOORBEELD 4: Toetsdata:

OPPERVLAKTE:	37
TIPE VERF:	Vinyl
AANTAL LAE:	8

Voorbeeld van afvoer op **lblCost** gebaseer op die toetsdata hierbo:



(8)

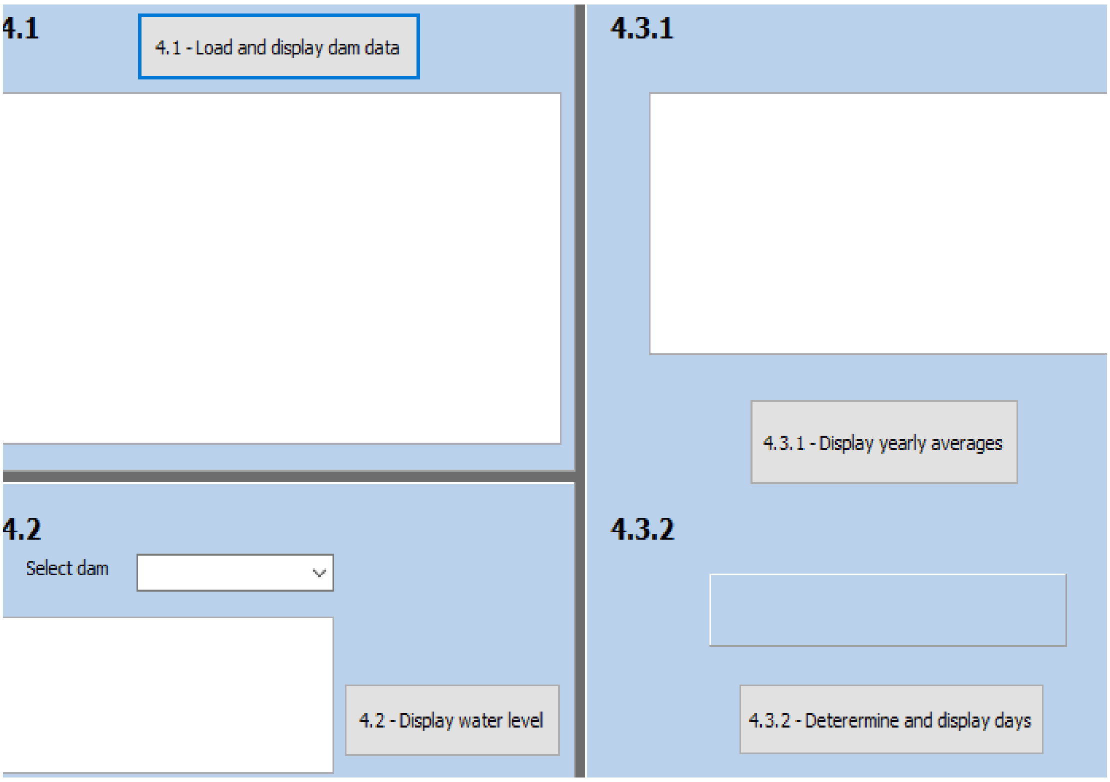
TOTAAL AFDELING C: 40

AFDELING D: PROBLEEMOPLOSSING

SCENARIO:
Die stad Kaapstad gebruik 'n stelsel genaamd Wes-Kaapse Watervoorsieningsstelsel (WCWSS) om die watervlakke van die damme wat die stad van water voorsien, te monitor.
Daar is ses hoofdamme en agt kleiner damme wat water aan die stad Kaapstad lewer. Hierdie vraag fokus op die ses hoofdamme.

VRAAG 4: PROBLEEMOPLOSSING

- Kompileer en voer die program in die **Question 4**-lêergids uit.
- Die program het op die oomblik geen funksionaliteit nie.
- Die volgende gebruikerskoppelvlak word vertoon:



Die program bevat die verklarings van drie skikkings genaamd **arrDams**, **arrDamLevels** en **arrCapacities**. Al drie die skikkings is verklaar met 'n maksimum grootte van 6 elemente:

- **arrDams** is verklaar vir die **SES HOOFDAMME** wat Kaapstad van water voorsien.
- **arrDamLevels** is verklaar vir die huidige watervlak van elke dam wat as persentasie gestoor word.
- **arrCapacities** is verklaar vir die individuele waterbergingsvermoë van elke dam in **MILJOENE KUBIEKE METER**.

Die volgende twee skikkings word met 'n maksimum grootte van 5 elemente verklaar:

- **arrYears** bevat die afgelope vyf jaar (2015 - 2019).
- **arrAverages** bevat die gemiddelde persentasie van die watervlakke vir elk van die hoofdamme soos aangedui aan die einde van Julie van elke jaar.

```
var
  arrYears: array [1..5] of string = ('2019', '2018', '2017', '2016', '2015');
  arrAverages: array [1..5] of real = (53.1, 25.1, 42.4, 53.5, 0);
```

NOTA:
arrTempDams, **arrTempDamLevels** en **arrTempCapacities** is drie konstante skikkings wat met waardes gevul is en moet SLEGS gebruik word om **VRAAG 4.2** en **VRAAG 4.3** te beantwoord as u kode vir **arrDams**, **arrDamLevels** en **arrCapacities** in **VRAAG 4.1** nie suksesvol was nie.

'n Tekslêer met die naam **MajorDams.txt** is verskaf en bevat 6 reëls met data. Elke reël bevat die naam van 'n dam, die dam se volle kapasiteit en die persentasie watervlak van die dam. Die inligting in elke reël word in die volgende formaat gestoor:

<dam name>;<full capacity>,<water level>

Voorbeeld van die inhoud van die tekslêer:

```
Berg River;130.010,89.5
Steenbras Lower;33.517,53.2
Steenbras Upper;31.767,80.8
Dieewaterskloof;480.188,50.5
Voelvlei Dam;164.095,64.0
Wemmershoek;58.644,51.4
```

Verduideliking van die eerste twee reëls data in die **MajorDams.txt** -tekslêer:

- Die volle kapasiteit van die Berg River dam is 130.010 miljoen kubieke meter en die watervlak is 89.5%.
- Die volle kapasiteit van Steenbras Lower dam is 33.517 miljoen kubieke meter en die watervlak is 53.2%.

Voltooi die kode vir die vereistes beskryf in **VRAAG 4.1** tot **VRAAG 4.3**.

4.1 **KNOPPIE [4.1 - Load and display dam data]**

Skryf kode vir die volgende:

- Toets om vas te stel of die tekslêer **MajorDams.txt** toeganklik is. Indien nie, vertoon 'n geskikte boodskap wat aandui dat die tekslêer nie beskikbaar is nie en verlaat die prosedure.
- Maak gebruik van die tekslêer **MajorDams.txt** om die skikkings **arrDams**, **arrDamCapacities** en **arrDamLevels** met die toepaslike data te vul.
- Vertoon die inhoud van **arrDams**, **arrDamCapacities** en **arrDamLevels** met toepaslike opskrifte in die komponent **redQ4_1**, soos getoon in die voorbeeld van die afvoer.
- Bepaal die totale kapasiteit van die hoofdamme.
- Vertoon die berekende totale kapasiteit op die **redQ4_1**-komponent soos aangedui in die voorbeeld van die afvoer.

VOORBEELD VAN AFVOER:

Name of Dam	Full capacity	Water level %
Berg River	130.01	89.5
Steenbras Lower	33.517	53.2
Steenbras Upper	31.767	80.8
Theewaterskloof	480.188	50.5
Voelvlei Dam	164.095	64
Wemmershoek	58.644	51.4
Total capacity	898.221	

(14)

Voorafkennis	
Konsep	**Waar om in die DBE handboek te hersien**
While lus – word gebruik met tekslêers – teller	Gr 10 – Unit 7.7 bladsy 182 Gr 11 – Unit 5.2 bladsy 106
For lus – word gebruik met skikkings	Gr 10 – Unit 7.3 bladsy 162 Gr 11 – Unit 3.1 bladsy 44
Tekslêers – toets of die tekslêer bestaan – lees van 'n tekslêer	Gr 11 – Hoofstuk 5 bladsy 101 Gr 11 – Unit 5.5 bladsy 106
Skikkings – stel 'n waar vir die indeks van die skikking – teller (om in lus te gebruik) – gaan deur (*loop*) parallelle skikkings afvoer bereken: som van die elemente van skikking – Twee-dimensionele skikkings *Word nie in hierdie vraag gevra nie, maar daar sal parallelle of twee-dimensionele skikkings gevra word. Jy moet al twee ken.*	Gr 11 – Hoofstuk 3 bladsy 41 Gr 11 – Unit 3.1 bladsy 42 Gr 11 – Unit 3.3 bladsy 64 Gr 12 – Hoofstuk 3 bladsy 81
String manipulasie funksies pos, copy en delete	Gr 11 – Hoofstuk 4 Gr 11 Unit 4.2 bladsy 86

Vraag 4.1	
Vraag	Belangrike konsepte om te doen
Toets om vas te stel of die tekslêer MajorDams.txt toeganklik is. Indien nie, vertoon 'n geskikte boodskap wat aandui dat die tekslêer nie beskikbaar is nie en verlaat die prosedure.	– Toets of MajorDams.txt toeganklik is dit kan op 2 maniere gedoen word – Vertoon boodskap as die lêer nie bestaan nie – - ontsnap uit prosedure
Maak gebruik van die tekslêer MajorDams.txt om die skikkings arrDams, arrDamCapacities en arrDamLevels met die toepaslike data te vul.	– Skep teller en stel na 0 voor die lus – Gaan deur (*loop*) die tekslêer; While NOT EOF – inkrementeer teller in die lus (Inc) – lees lyn uit tekslêer – Onttrek data uit die lyn – Ken data toe aan die korrekte skikkings

Vertoon die inhoud van arrDams, arrDamCapacities en arrDamLevels met toepaslike opskrifte in die komponent redQ4_1, soos getoon in die voorbeeld van die afvoer.	– Maak redQ4_1 skoon (*Clear*) – Vertoon opskrif – gaan deur (*loop*) deur skikking (of 1..6) – Vertoon arrDams + *tab* + arrDamCapacities + *tab* + arrDamLevels – - met dieselfde lus indeks
Bepaal die totale kapasiteit van die hoofdamme.	– skep 'n heelgetal veranderlike en stel dit na 0 – gaan deur (*loop*) deur arrDamCapacities vermeerder veranderlike met arrDamCapacities by die lus indeks
Vertoon die berekende totale kapasiteit op die redQ4_1-komponent soos aangedui in die voorbeeld van die afvoer.	– Vertoon veranderlike op redQ4_1

4.2 **KNOPPIE [4.2 - Display water level]**

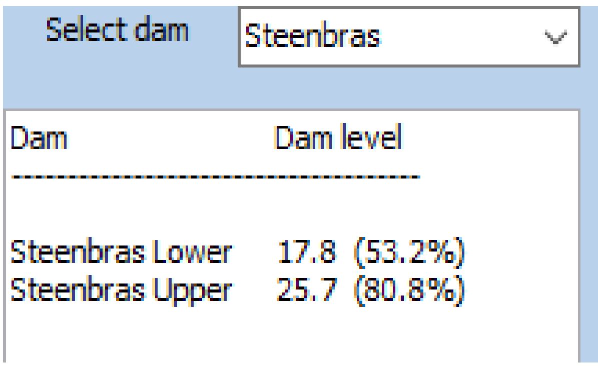
Die gebruiker moet 'n dam selekteer uit die **cmbDams**-kombinasielys (*combo box*).
Die doel van die program is om die inhoud van die geselekteerde dam in **miljoene kubieke meter** te bereken en vertoon.

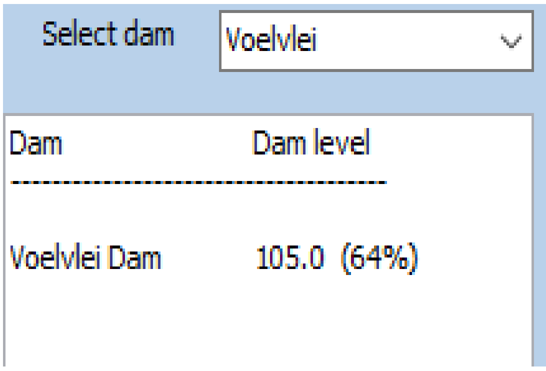
NOTA:
Die Steenbras-dam bestaan uit twee sub-damme, naamlik Steenbras Lower en Steenbras Upper.

Kode is voorsien om die opskrifte te vertoon soos aangedui in die voorbeelde van afvoer onderaan die bladsy.
Skryf kode vir die volgende:

- Bereken die inhoud van die dam(me) wat deur die gebruiker uit die **cmbDams**-komponent geselekteer is.
- Gebruik die waardes wat in **arrDamLevels** en **arrCapacities** gestoor is om die inhoud van die dam(me) in **miljoene kubieke meter** te bereken.
- Vertoon die persentasie van die watervlak(ke) tussen hakies (sien voorbeelde van afvoer).
- As 'n dam uit sub-damme bestaan, moet die inhoud van al die sub-damme bereken en vertoon word (sien voorbeelde van afvoer).
- Vertoon die afvoer wat bereken is op die **redQ4_2**-komponent soos aangedui in die voorbeeld van die afvoer hieronder.

VOORBEELD VAN AFVOER:





(8)

Voorafkennis	
Konsep	**Waar om in die DBE handboek te hersien**
for lus – met skikking	Gr 10 – Unit 7.3 bladsy 162 Gr 11 – Unit 3.1 bladsy 44
If	Gr 10 – Hoofstuk 5 bladsy 87
Combobox	Gr 10 – Unit 7.1 bladsy 155
Skikkings – gaan **deur** (loop) parallelle skikkings – afvoer	Gr 11 – Hoofstuk 3 bladsy 41 Gr 11 – Unit 3.3 bladsy 64
String manipulasie – **pos**	Gr 11 – Hoofstuk 4 Gr 11 Unit 4.2 bladsy 86

Vraag 4.2	
Vraag	**Belangrike konsepte om te doen**
Bereken die inhoud van die dam(me) wat deur die gebruiker uit die cmbDams-komponent geselekteer is.	– skep veranderlikes om damLevel en damCapacities te stoor – stel veranderlikes na 0 – verkry die geselekteerde *item* in cmbDams – gaan deur (*loop*) deur arrDams – as die cmbDams en arrDams gelyk is inkrementeer damLevel met arrDamLevels by die indeks van die lus – inkrementeer damCapacities met arrCapacities by die indeks van die lus
Gebruik die waardes wat in arrDamLevels en arrCapacities gestoor is om die inhoud van die dam(me) in miljoene kubieke meter te bereken.	– Berekening: (damLevel by gestoorde indeks / 100) * damCapacities by gestoorde indeks
Vertoon die persentasie van die watervlak(ke) tussen hakies (sien voorbeelde van afvoer).	- formateer jou afvoer korrek
As 'n dam uit sub-damme bestaan, moet die inhoud van al die sub-damme bereken en vertoon word (sien voorbeelde van afvoer).	– verander jou kode om die if in die lus te verander om die pos-metode of enige ander metode te gebruik om al die sub-damme vas te stel
Vertoon die afvoer wat bereken is op die redQ4_2-komponent soos aangedui in die voorbeeld van die afvoer hieronder.	– formateer jou afvoer – onthou die hakies ***Wenk** –kyk na die afvoer. As dit een desimale plek na die punt het, kan jy dit afrond na een desimale plek.*

4.3.1 **KNOPPIE [4.3.1 - Vertoon yearly averages]**

Die gemiddelde persentasie van die vlak van die damme word vir vyf jaar gehou.

Die jare en die gemiddelde watervlakpersentasies word in die **arrYears** en **arrAverages** skikkings gestoor.

Die data in die **arrAverages**-skikking is verkeerd vasgelê:

- Die gemiddelde persentasie vir 2019 bevat die data vir 2018.
- 2018 bevat die data vir 2017.
- 2017 bevat die data vir 2016.
- 2016 bevat die data vir 2015.
- Die data vir 2015 is tans 0%.

Kode is voorsien om opskrifte te vertoon soos aangedui op die voorbeelde van afvoer.

Die gemiddelde damvlakpersentasie vir 2019 is 59.8%.

Skryf kode vir die volgende:

- Pas die **arrAverages** aan sodat die korrekte watervlakpersentasies vir die korrekte ooreenstemmende jare geberg word:
  - Die persentasie watervlak vir 2019 moet in die eerste element van die skikking geberg word.
  - Die res van die waardes moet korrek in elemente 2 tot 5 geberg word.
- Vertoon die jare en gemiddelde watervlakpersentasies op die **redQ4_3_1** afvoer komponent.

VOORBEELD VAN AFVOER

Year	Water level%
2019	59.8
2018	53.1
2017	25.1
2016	42.4
2015	53.5

(4)

Voorafkennis	
Konsep	**Waar om in die DBE handboek te hersien**
Skikkings	Gr 11 – Hoofstuk 3
- stel 'n waarde na die skikkings indeks	Gr 11 – Unit 3.1 bladsy 42
- teller (om vir die lus te gebruik)	
- gaan deur (*loop*) deur skikkings	Gr 11 – Unit 3.1 Bladsy 50

Vraag 4.3.1	
Vraag	**Belangrike konsepte om te doen**
Pas die arrAverages aan sodat die korrekte watervlakpersentasies vir die korrekte ooreenstemmende jare geberg word: ○ Die persentasie watervlak vir 2019 moet in die eerste element van die skikking geberg word. ○ Die res van die waardes moet korrek in elemente 2 tot 5 geberg word.	– gaan deur (*loop*) deur die array, begin by die laaste index tot die 2de indeks – vervang arrAverages by indeks I – tot die vorige indeks (I – 1) – stel arrAverages by indeks 1 na 59.8
Vertoon die jare en gemiddelde watervlakpersentasies op die redQ4_3_1 afvoer komponent.	– Vertoon die opskrif – - Vertoon die skikking

4.3.2 **KNOPPIE [4.3.2 - Determine en display days]**

Die aantal dae wat water aan die stad Kaapstad voorsien kan word, moet bereken word volgens die **TOTALE** watervlakpersentasie soos aangedui in die konstante **rTOT** (59.8%) wat voorsien is.

Die stad se daaglikse watergebruik is 0.38% van die **TOTALE** watervlakpersentasie. 'n Lokale konstante, genaamd **rDaily**, word voorsien en bevat die waarde vir die stad se daaglikse watergebruik (0.38%).

NOTA: Die laaste 10% van die dam se water is nie geskik vir menslike gebruik nie.

Skryf kode vir die volgende:

- Bereken die totale aantal dae waarvoor die stad Kaapstad van water voorsien kan word as die totale watervlakpersentasie konstant bly, afgerond tot die laagste heelgetal.
- Vertoon die berekende aantal dae met 'n geskikte boodskap op **pnlDays**.

VOORBEELD VAN AFVOER: Water is available for 131 days.

(4)

Voorafkennis	
Konsep	**Waar om in die DBE handboek te hersien**
Wiskundige probleme	Gr 10 – Hoofstuk 4 bladsy 61
Wiskundige funksie- Floor	Gr 10 – Unit 4.2 bladsy 72

Vraag 4.3.2	
Vraag	**Belangrike konsepte om te doen**
Bereken die totale aantal dae waarvoor die stad Kaapstad van water voorsien kan word as die totale watervlakpersentasie konstant bly, afgerond tot die laagste heelgetal.	– trek eers die 10% van rTot af – Berekening: rTot / rDaily – - rond “af” – Floor
Vertoon die berekende aantal dae met 'n geskikte boodskap op pnlDays.	– Vertoon die aantal dae wat oor is – - onthou om die karakterstring op te bou

TOTAAL AFDELING D: [30]
GROOTTOTAAL: 150