



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NASIONALE
SENIOR SERTIFIKAAT**

GRAAD 12

INLIGTINGSTEGNOLOGIE V1

NOVEMBER 2022

PUNTE: 150

TYD: 3 uur

Hierdie vraestel bestaan uit 24 bladsye en 2 bladsye met data.

INSTRUKSIES EN INLIGTING

1. Hierdie vraestel is in VIER afdelings ingedeel. Kandidate moet AL die vrae in AL VIER afdelings 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 Delphi as programmeringstaal is. Die Delphi-programmeringstaal moet gebruik word om die vrae te beantwoord.
4. Maak seker dat jy die vrae beantwoord volgens die spesifikasies wat in elke vraag gegee word. Punte sal volgens die voorgeskrewe 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 die Delphi-programmeringstaal 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 eksamennommer 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. Jou eksamennommer moet op al die drukstukke verskyn. 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. 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 wagwoordbeskernde uitvoerbare lêers verskaf.

Doen die volgende:

- Dubbelklik op die volgende uitvoerbare lêer wat met 'n wagwoord beskerm is: **DataAFRNov2022.exe**
- Klik op die 'Extract'-knoppie.
- Sleutel die volgende wagwoord in: **#TIME4RRR&\$**

Nadat dit onttrek ('extracted') is, sal die volgende lys lêers in die lêergids ('folder') **DataAFRNov2022** beskikbaar wees:

Vraag 1:

Vraag1_P.dpr
Vraag1_P.dproj
Vraag1_P.res
Vraag1_U.dfm
Vraag1_U.pas

Vraag 2:

ConnectDB_U.dcu
InsamelingDB - Copy.mdb
InsamelingDB.mdb
Vraag2_P.dpr
Vraag2_P.dproj
Vraag2_P.res
Vraag2_U.dfm
Vraag2_U.pas

Vraag 3:

logo.jpeg
SonkragAanleg_U.pas
Vraag3_P.dpr
Vraag3_P.dproj
Vraag3_P.res
Vraag3_U.dfm
Vraag3_U.pas

Vraag 4:

rrr.jpeg
Vraag4_P.dpr
Vraag4_P.dproj
Vraag4_P.res
Vraag4_U.dfm
Vraag4_U.pas

AFDELING A**VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

Doen die volgende:

- Maak die onvolledige program in die **Vraag 1**-lêergids oop.
- Sleutel jou eksamennommer 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.

Voorbeeld van die grafiese gebruikerskoppelvlak (GGK ('GUI')):

The screenshot shows a graphical user interface for a programming exercise. The window title is "Vraag1 - Algemene programmeringsbeginsels". It contains five distinct sections for different questions:

- Vraag 1.1:** Contains a button labeled "Paneel" and a text box labeled "1.1 - Eienskappe".
- Vraag 1.2:** Contains a label "Selekteer maand:" followed by a dropdown menu, and a button labeled "1.2 - Verlofmaand".
- Vraag 1.3:** Contains three input fields labeled "A:", "B:", and "C:", each with the number "0" entered. Below them is a button labeled "1.3 - Bereken".
- Vraag 1.4:** Contains a button labeled "1.4 - Punte" and a large empty text area below it.
- Vraag 1.5:** Contains two input fields labeled "Reuk" and "keur", a button labeled "1.5 - Anagram", and a large empty text area below the button.

- Voltooi die kode vir elke afdeling van VRAAG 1, soos in VRAAG 1.1 tot VRAAG 1.5 wat volg, beskryf is.

1.1 Knoppie [1.1 - Eienskappe]

Skryf kode om die eienskappe van paneel **pnIV1_1** soos volg te verander:

- Stel die kleur op geel ('yellow').
- Stel die skrif ('font') op skuinsgedruk ('italics').
- Verander die opskrif ('caption') na 'Ek hou van programmering!'.

Voorbeeld van afvoer:



(3)

1.2 Knoppie [1.2 - Verlofmaand]

'n Werknemer selekteer die maand waarin hy/sy verlof wil neem. Die maatskappy is van Junie tot Augustus gesluit en verlof kan nie gedurende hierdie tydperk geneem word nie.

Die kombinasieblokkie ('combo box') **cmbV1_2** is met die name van die maande van Januarie tot Desember gevul.

Die gebruiker moet 'n maand uit kombinasieblokkie **cmbV1_2** selekteer.

Skryf kode om die volgende te doen:

- Verkry die maand wat uit kombinasieblokkie **cmbV1_2** geselekteer is.
- As die maatskappy in die geselekteerde maand oop is, vertoon 'n boodskap op die etiket ('label') **lbIV1_2** deur hierdie formaat te gebruik:

"Jou verlof in " <maand> " is toegestaan."

- As die maatskappy gedurende die geselekteerde maand gesluit is, doen die volgende:
 - Vertoon 'n boodskap op die etiket **lbIV1_2** deur hierdie formaat te gebruik:

"Maatskappy gesluit, selekteer 'n ander maand."
 - Maak die seleksie van die maand uit kombinasieblokkie **cmbV1_2** skoon.

Voorbeeld van afvoer as April geselekteer is:



Voorbeeld van afvoer as Julie geselekteer is.



(7)

1.3 Knoppie [1.3 - Bereken]

Die resultaat van die volgende formule word benodig:

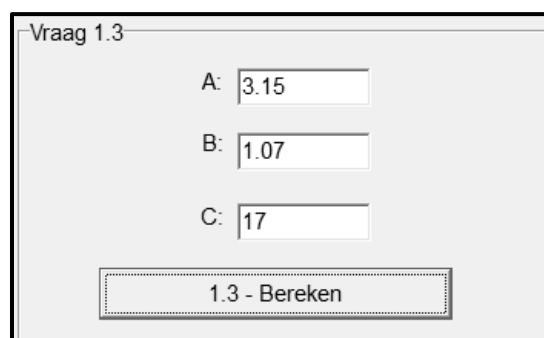
$$C = \sqrt{A^5 + \pi B^2}$$

Die gebruiker moet die waardes vir A en B in die redigeerblokkies ('edit boxes') wat voorsien is, insleutel.

Skryf kode om die volgende te doen:

- Verkry die waarde van A uit redigeerblokkie **edtV1_3_1** en die waarde van B uit redigeerblokkie **edtV1_3_2**.
- Gebruik geskikte wiskundige funksies om die waarde van C te bereken, met gebruik van die formule wat voorsien is.
- Vertoon die afgekapte ('truncated') waarde van C in redigeerblokkie **edtV1_3_3**.

Voorbeeld met die waardes van 3.15 (A) en 1.07 (B) as toevoer en die resultaat (C) as afvoer:



(8)

1.4 Knoppie [1.4 - Punte]

Die aantal vakke waarvoor leerders ingeskryf het, wissel. Die punte wat 'n leerder vir elke vak behaal het, moet ingesleutel word en die gemiddelde punt moet bereken word volgens die getal punte wat ingesleutel is.

Skryf kode om die volgende te doen:

- Laat die gebruiker toe om die punte wat 'n leerder per vak behaal het, in te sleutel, beginnende by vak nommer 1.
- Gebruik 'n voorwaardelike lus en 'n toevoerblokkie ('input box') om die punte in te sleutel. Die lus moet getermineer word as 'n punt van -1 ingesleutel word.
- Soos wat punte ingesleutel word, moet 'n genommerde lys met vakke en punte in die **redV1_4**-afvoerarea vertoon word, soos getoon in die voorbeeld wat volg.
- Nadat die lus vir toevoer getermineer is, bereken die gemiddelde punt wat die leerder behaal het. Vertoon die gemiddelde punt in die **redV1_4**-afvoerarea, geformateer tot TWEE desimale plekke.

Voorbeeld van toevoer van punte:

The image shows three separate dialog boxes, each titled 'Leerderpunte'. Each dialog box contains a label 'Sleutel punt in vir vak: [number]' and a text input field. Below the input field are 'OK' and 'Cancel' buttons.

- The first dialog box is for 'vak: 1' and the input field contains the number '75'.
- The second dialog box is for 'vak: 2' and the input field contains the number '68'.
- The third dialog box is for 'vak: 3' and the input field contains the number '-1'.

Voorbeelde van afvoer:

Punte vir twee vakke en die gemiddeld:

The image shows a rectangular output window containing the following text:

```
Vak 1: 75  
Vak 2: 68  
Gemiddelde punt: 71.50
```

Punte vir agt vakke en die gemiddeld:

Vak 1: 81
Vak 2: 55
Vak 3: 92
Vak 4: 56
Vak 5: 49
Vak 6: 80
Vak 7: 72
Vak 8: 73
Gemiddelde punt: 69.75

(11)

1.5 Knoppie [1.5 - Anagram]

Twee woorde moet geanalyseer word om te bepaal of die woorde 'n perfekte anagram vorm of nie. 'n Perfekte anagram is wanneer 'n ander woord gevorm word deur die letters van die oorspronklike woord te herrangskik. Byvoorbeeld, die woord kat is gevorm uit die woord tak, keur uit reuk en sweet uit weste.

Kode is voorsien om twee woorde uit redigeerblokkies ('edit boxes') **edtV1_5_1** en **edtV1_5_2** te verkry en na kleinletters ('lower case') om te skakel.

Skryf kode om te bepaal of die twee woorde wat uit die redigeerblokkies verkry is, 'n anagram vorm of nie en vertoon 'n gepaste boodskap in memo **memV1_5**.

Voorbeelde van toevoer en afvoer:

Vraag 1.5

Keur Reuk

1.5 - Anagram

Die woorde vorm 'n anagram.

Vraag 1.5

Vader Radar

1.5 - Anagram

Die woorde vorm nie 'n anagram nie.

(11)

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

TOTAAL AFDELING A: 40

AFDELING B**VRAAG 2: SQL EN DATABASISPROGRAMMERING**

Die Collect-a-Can-herwinningsinisiatief het 'n databasis ontwikkel met die naam **InsamelingDB.mdb**, wat inligting oor die insameling van blikkies van kliënte vir die jaar 2020 tot 2022 bevat.

'n Toepassing word benodig wat die **InsamelingDB.mdb**-databasis sal gebruik om die data en betalings aan kliënte wat aan die kontant-vir-blikkies-inisiatief deelneem, te bestuur.

Die databasis bevat twee tabelle met die name **tblKliente** en **tblBlikInsameling**.

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

Doen die volgende:

- Maak die onvolledige projeklêer met die naam **Vraag2_P.dpr** in die **Vraag 2**-lêergids oop.
- Sleutel jou eksamennummer as kommentaar in die eerste reël van die **Vraag2_U.pas**-eenheidlêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie. Die inhoud van die tabelle word vertoon soos wat hieronder op die seleksie van oortjieblad ('tab sheet') **Vraag 2.2 - Delphi-kode** getoon word.

KlientID	KlientNaam	KlientVan	Adres	Stad
ABI10	Prashant	Govender	Bergstraat 72	Kimberley
BUS06	Busi	Nkosi	Donaldweg 65	Welkom
CHR08	Chris	Ferreira	Richmondstraat 188	Potchefstroom
DAM07	Damian	Coetzer	Kaaprylaan 12	Durban

InsamelID	InsamelDatum	GetalBlikkies	Betaal	KlientID
C001	2020/01/19	412	True	WIL12
C002	2020/03/21	300	False	GER01
C003	2020/03/23	599	True	WIL12
C004	2020/03/25	514	True	WIL12

Vraag 2.2.1

Voeg in

Vraag 2.2.2

Jaar:

2020 2021 2022

Persentasie

Herwin databasis Sluit Af

- Volg die instruksies hieronder om die kode vir elke afdeling wat in VRAAG 2.1 en VRAAG 2.2 beskryf word, te voltooi.
- Gebruik SQL-stellings om VRAAG 2.1 te beantwoord en Delphi-kode om VRAAG 2.2 te beantwoord.

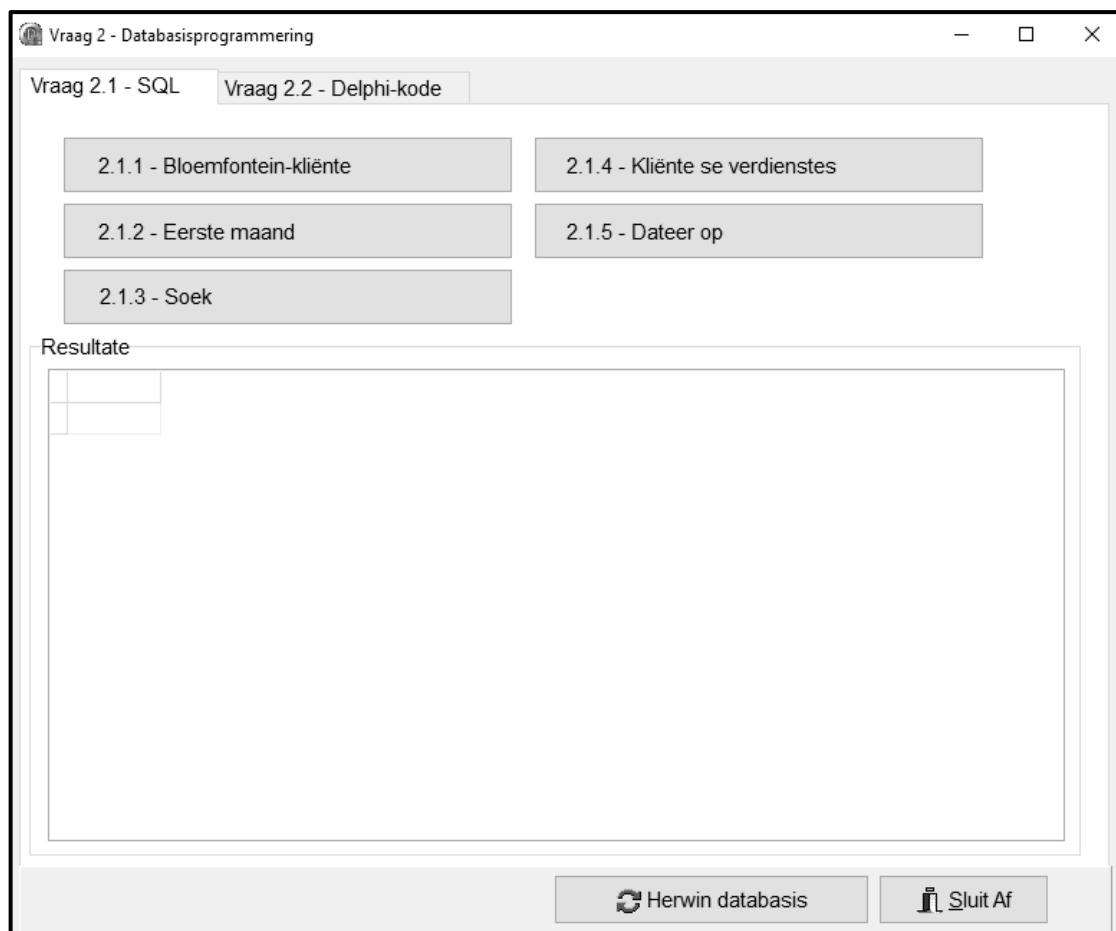
LET WEL:

- Die 'Herstel databasis'-knoppie ('Restore database') word voorsien om die data wat in die databasis voorkom, na die oorspronklike inhoud te herstel.
- Die inhoud van die databasis is met 'n wagwoord beskerm, m.a.w. jy sal NIE in staat wees om toegang tot die inhoud van die databasis te verkry deur Microsoft Access te gebruik NIE.
- Kode word voorsien om die GGK('GUI')-komponente met die databasis te koppel. MOENIE enige van die kode wat voorsien word, verander NIE.
- TWEE veranderlikes word as publieke veranderlikes verklaar, soos in die tabel hieronder beskryf word.

Veranderlike	Datatype	Beskrywing
tblKliente	TADOTable	Verwys na die tabel tblKliente
tblBlikInsameling	TADOTable	Verwys na die tabel tblBlikInsameling

2.1 Oortjieblad ('Tab sheet') [Vraag 2.1 - SQL]

Voorbeeld van die gebruikerskoppelvlak (GGK) vir VRAAG 2.1:



LET WEL:

- Gebruik SLEGS SQL-kode om VRAAG 2.1.1 tot VRAAG 2.1.5 te beantwoord.
- Kode om die SQL-stellings uit te voer en die resultate van die navraag te vertoon, is voorsien. Die SQL-stellings wat aan die **sSQL1**, **sSQL2**, **sSQL3**, **sSQL4** en **sSQL5** toegeken is, is onvolledig.

Voltooi die SQL-stellings om die take wat in VRAAG 2.1.1 tot VRAAG 2.1.5 hieronder beskryf word, uit te voer.

2.1.1 Knoppie [2.1.1 - Bloemfontein-kliënte]

Vertoon al die besonderhede van kliënte wat in Bloemfontein bly uit die **tblKliente**-tabel, volgens die **KlientVan**-veld gesorteer.

Voorbeeld van afvoer:

KlientID	KlientNaam	KlientVan	Adres	Stad
JAC05	Jacob	Human	Humanstraat 8	Bloemfontein
PIE12	Piet	Mogorosi	Stormweg 5	Bloemfontein
RHO09	Rhoda	Somers	Marilynstraat 14	Bloemfontein
GER01	Gert	Vermeulen	Dawnstraat 55	Bloemfontein

(4)

2.1.2 Knoppie [2.1.2 - Eerste maand]

Vertoon die **InsamelID**, **InsamelDatum** en **GetalBlikkies** van insamelings wat in Januarie gedoen is.

Voorbeeld van afvoer van die eerste vyf rekords:

InsamelID	InsamelDatum	GetalBlikkies
C001	2020/01/19	412
C031	2021/01/05	1000
C032	2021/01/10	450
C033	2021/01/16	2150
C034	2021/01/28	800

(3)

2.1.3 Knoppie [2.1.3 - Soek]

Kode is voorsien om 'n letter in te sleutel deur 'n toevoerblokkie ('input box') te gebruik. Die letter word in 'n veranderlike met die naam **sLetter** gestoor. Vertoon AL die besonderhede van kliënte wie se **KlientID** begin met die letter wat ingesleutel is.

Voorbeeld van afvoer as die letter J ingesleutel is:

KlientID	KlientNaam	KlientVan	Adres	Stad
JOH03	Johan	Weston	Michellinstraat 43	Vanderbijlpark
JAC05	Jacob	Human	Humanstraat 8	Bloemfontein

(4)

2.1.4 Knoppie [2.1.4 - Kliënte se verdienstes]

Kliënte sal R8,00 ontvang vir elke kilogram blikkies wat ingesamel is. Die maatskappy gebruik die formule: 1 kilogram = 76 blikkies.

Bereken en vertoon die totale bedrag wat elke kliënt sal ontvang vir die blikkies wat hulle ingesamel het, na geldeenheid geformateer. Vertoon die **KlientNaam**-veld en die totale bedrag wat ontvang sal word deur die veldnaam **Totale Bedrag** te gebruik.

Voorbeeld van afvoer van die eerste vyf rekords:

KlientNaam	Totale Bedrag
Busi	R905.26
Chris	R1 173.47
Damian	R465.58
Gert	R661.68
Henry	R400.11

(8)

2.1.5 Knoppie [2.1.5 - Dateer op]

Die besonderhede van 'n spesifieke insameling blikkies is verkeerd ingesleutel.

Verander die besonderhede van rekord **C003** in die **tblBlikInsameling**-tabel soos volg:

- GetalBlikkies: 250
- 'n Betaling is nie gemaak nie

Kode is voorsien om 'n boodskap te vertoon wat aandui dat 'n rekord in die databasis verander is.

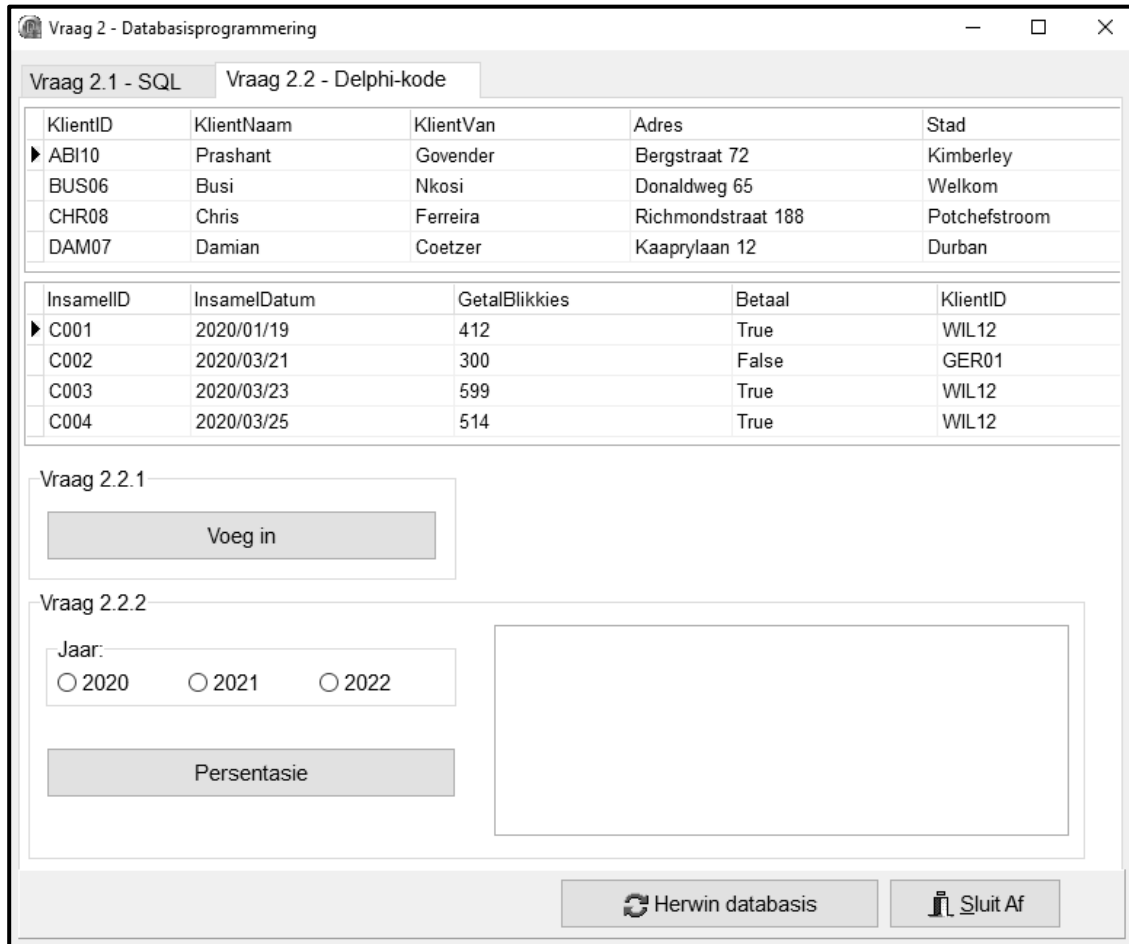
(4)

2.2 Oortjieblad ('Tab sheet') [Vraag 2.2 - Delphi-kode]

LET WEL:

- Gebruik SLEGS Delphi-programmeringskode om VRAAG 2.2.1 en VRAAG 2.2.2 te beantwoord.
- GEEN punte sal vir SQL-stellings in VRAAG 2.2 toegeken word nie.

Voorbeeld van grafiese gebruikerskoppelvlak (GGK ('GUI')) vir VRAAG 2.2:



2.2.1 **Knoppie [2.2.1 - Voeg in]**

Skryf kode om 'n nuwe rekord by die **tblKliente**-tabel te voeg. Die besonderhede van die kliënt is soos volg:

- Kliënt se ID: **CHA01**
- Kliënt se naam: **Charles**
- Kliënt se van: **du Boit**
- Adres: **Van Wouw Straat 24**
- Stad: **Kaapstad**

Voorbeeld van die eerste vier rekords van die **tblKliente**-tabel wat toon dat die rekord van die nuwe kliënt suksesvol in die tabel bygevoeg is:

KlientID	KlientNaam	KlientVan	Adres	Stad
CHA01	Charles	du Boit	Van Wouwstraat 24	Kaapstad
CHR08	Chris	Ferreira	Richmondstraat 188	Potchefstroom
DAM07	Damian	Coetzer	Kaaprylaan 12	Durban
GER01	Gert	Vermeulen	Dawnstraat 55	Bloemfontein

(4)

2.2.2 Knoppie [2.2.2 - Persentasie]

Die maatskappy wil die getal blikkies wat deur 'n spesifieke kliënt in 'n spesifieke jaar ingesamel is, bereken as 'n persentasie van die totale getal blikkies wat in daardie spesifieke jaar deur die maatskappy ingesamel is.

Die gebruiker moet die volgende doen:

- Selekteer 'n kliënt uit die DBGrid deur op die rekord te klik.
- Selekteer 'n jaar uit die radiogroep ('radio group') **rgpV2_2_2**.

Kode is voorsien om die jaar wat uit die radiogroep **rgpV2_2_2** geselekteer is, te verkry.

Gebruik die **redV2_2_2**-afvoerarea om die inligting wat hieronder genoem word, te vertoon.

Skryf kode om die volgende te doen:

- Vertoon die naam en van van die kliënt wat geselekteer is.
- Bepaal en vertoon die totale getal blikkies wat vir die geselekteerde jaar deur die kliënt ingesamel is.
- Bepaal en vertoon die totale getal blikkies wat vir die geselekteerde jaar deur die Collect-a-Can-maatskappy ingesamel is.
- Bereken watter persentasie van die totale getal blikkies wat in die geselekteerde jaar ingesamel is, deur die kliënt ingesamel is. Vertoon die persentasie geformateer tot twee desimale plekke.

Voorbeeld van afvoer as die kliënterekord met KlientID **BUS06** en die jaar **2022** geselekteer is:

Busi Nkosi	
Kliënt in 2022 ingesamel:	2000
Maatskappy in 2022 ingesamel:	21660
Persentasie deur kliënt ingesamel:	9.23%

Voorbeeld van afvoer as die kliënterekord met KlientID **CHR08** en die jaar **2021** geselekteer is:

Chris Ferreira	
Kliënt in 2021 ingesamel:	5268
Maatskappy in 2021 ingesamel:	32687
Persentasie deur kliënt ingesamel:	16.10%

(13)

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

TOTAAL AFDELING B: 40

AFDELING C**VRAAG 3: OBJEK-GEÛRIENTEERDE PROGRAMMERING**

Die sektor vir volhoubare energie wil rekord hou van die sonkragaanlegte in die land en hulle kapasiteit om die krag wat hulle opwek, te vermeerder.

Doen die volgende:

- Maak die onvolledige program in die **Vraag 3**-lêergids oop.
- Maak die onvolledige objekklas **SonkragAanleg_U.pas** oop.
- Sleutel jou eksamennommer as kommentaar in die eerste reël van beide die **Vraag3_U.pas**-lêer en die **SonkragAanleg_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans beperkte funksionaliteit.

Voorbeeld van grafiese gebruikerskoppelvlak (GGK):

Vraag 3 - Objek-geïntegreerde programmering

Sonkragaanlegtoepassing

Vraag 3.2.1

Aanlegkode:

Getal panele:

Krag per paneel:

3.2.1 - Instansieer objek

Vraag 3.2.2

Panele om by te voeg:

3.2.2 - Vermeerder panele

Vraag 3.2.3

Selekteer 'n seisoen:

3.2.3 - Dateer seisoen op

Vraag 3.2.4

3.2.4 - Bereken

Herstel

- Voltooi die kode soos in VRAAG 3.1 en VRAAG 3.2 wat volg, gespesifiseer is.

LET WEL: Jy mag NIE enige addisionele attribute of gebruikersgedefinieerde metodes byvoeg NIE, tensy dit eksplisiet in die vraag aangedui word.

- 3.1 Die onvolledige objekklas (**TSonkragAanleg**) wat voorsien is, bevat die verklaring van vier attribute wat 'n **SonkragAanleg**-objek beskryf.

Die attribute vir 'n **SonkragAanleg**-objek is soos volg verklaar:

Attribute	Beskrywing
fAanlegKode	'n Unieke kode vir die sonkragaanleg
fGetalPanele	Die getal panele wat by die aanleg geïnstalleer is
fKragPerPaneel	Die maksimum krag per paneel in kilowatt-uur (kWh)
fSeisoen	Die seisoen waarin krag opgewek gaan word – Somer, Herfs, Winter, Lente

Kode is voorsien vir die volgende toegangsmethodes ('accessor methods'):

- **getAanlegKode** wat die fAanlegKode-attribuu terugstuur
- **getGetalPanele** wat die fGetalPanele-attribuu terugstuur
- **getSeisoen** wat die fSeisoen-attribuu terugstuur

Voltooi die kode in die objekklas soos in VRAAG 3.1.1 tot VRAAG 3.1.5 hieronder beskryf word.

- 3.1.1 Skryf kode vir 'n **konstruktor**-metode wat die kode van die aanleg, die getal panele en die krag per paneel as parameters sal ontvang. Ken die parameterwaardes aan die onderskeie attribute toe. Ken die verstekwaarde 'Somer' aan die seisoen-attribuu toe. (5)
- 3.1.2 Skryf die kode vir 'n metode met die naam **inkGetalPanele** wat 'n waarde as parameter ontvang en die **fGetalPanele**-attribuu met die waarde wat ontvang is, te vermeerder. (4)
- 3.1.3 Skryf kode vir 'n wysigingsmetode ('mutator method') met die naam **setSeisoen** wat 'n waarde as parameter ontvang en die **fSeisoen**-attribuu stel na die waarde wat ontvang is. (3)
- 3.1.4 Skryf 'n metode met die naam **berekenKapasiteit** wat die inligting wat volg, gebruik om die kragopwekkingskapasiteit van die panele te bereken wat in die huidige seisoen geïnstalleer is. Stuur die resultaat as 'n reële waarde terug.

Die seisoen bepaal die ure sonlig per dag wat gebruik kan word om sonkrag te genereer, soos in die volgende tabel getoon:

SEISOEN	URE SONLIG PER DAG
Somer	10
Herfs	8
Winter	6
Lente	8

Die formule om die opwekkingskapasiteit (GC) van die sonkragaanleg te bereken, is soos volg:

$$GC = \text{GetalPanele} \times \text{KragPerPaneel} \times \text{UrePerDag} \quad (8)$$

3.1.5 Skryf 'n **toString**-metode om 'n string met al die attribute van die objek in die volgende formaat terug te stuur:

```
Aanlegkode: <AanlegKode>
Getal panele: <GetalPanele>
Krag per paneel: <KragPerPaneel>
Seisoen: <Seisoen>
```

(3)

3.2 'n Onvolledige program is in die **Vraag 3**-lêergids voorsien. Die program bevat kode vir die objekklas om toeganklik te wees en verklaar 'n objekveranderlike met die naam **objAanleg**.

Skryf kode om die take wat in VRAAG 3.2.1 tot VRAAG 3.2.4 hieronder beskryf word, uit te voer.

3.2.1 Knoppie [3.2.1 - Instansieer objek]

Skryf kode om die volgende te doen:

- Verkry die aanlegkode uit die redigeerblokkie **edtV3_2_1_Kode**, die getal panele uit die spin-redigeerder ('spin edit') **sedV3_2_1** en die krag per paneel uit die redigeerblokkie **edtV3_2_1_Krag**.
- Gebruik die inligting om 'n nuwe **SonkragAanleg**-objek te instansieer.
- Gebruik die **toString**-metode om die inligting van die **SonkragAanleg**-objek in die 'rich edit' **redV3** te vertoon.

Voorbeeld van toevoer en afvoer:

(6)

3.2.2 Knoppie [3.2.2 - Vermeerder panele]

Soos wat die behoefte aan meer krag ontstaan, kan panele bygevoeg word om die aanleg se kragopwekkingskapasiteit te vergroot.

Skryf kode om die volgende te doen:

- Verkry die waarde van die getal panele wat bygevoeg moet word uit die spin-redigeerder ('spin edit') **sedV3_2_2**.
- Roep die **inkGetalPanele**-metode met die waarde uit die spin-redigeerder as argument.
- Roep die relevante objekmetodes om die volgende in die 'rich edit' **redV3** te vertoon:
 - Die aanlegkode
 - Die getal panele nadat die waarde van die attribuut vermeerder is

Voorbeeld van toevoer en afvoer:

The image shows two windows. The top window, titled "Vraag 3.2.2", has a label "Panele om by te voeg:" and a spin box containing the number "3". Below the spin box is a button labeled "3.2.2 - Vermeerder panele". The bottom window displays the output of the operation: "Aanlegkode: CSE01" and "Getal panele: 18".

(4)

3.2.3 Knoppie [3.2.3 - Dateer seisoen op]

Die getal ure vir die opwekking van krag verskil volgens seisoen.

Die gebruiker moet die seisoen uit die kombinasieblokkie ('combo box') **cmbV3_2_3** selekteer.

Skryf kode om die volgende te doen:

- Verkry die geselekteerde seisoen uit die kombinasieblokkie **cmbV3_2_3**.
- Roep die **setSeisoen**-metode deur die waarde van die kombinasieblokkie as argument te gebruik.
- Gebruik die **toString**-metode om die inligting van die opgedateerde **Aanleg**-objek in die 'rich edit' **redV3** te vertoon.

Voorbeeld van toevoer en afvoer:

Vraag 3.2.3

Selekteer 'n seisoen: Winter ▾

3.2.3 – Dateer seisoen op

Aanlegkode: CSE01
 Getal panele: 18
 Krag per paneel: 0.532
 Seisoen: Winter

(3)

3.2.4 Knoppie [3.2.4 - Bereken]

Die maksimum kragkapasiteit wat deur die panele opgewek kan word, moet bereken word.

Skryf kode om die volgende in die 'rich edit' **redV3** te vertoon:

- Die seisoen as deel van die afvoerstring.
- Die resultaat van die **berekenKapasiteit**-metode na 'n string omgeskakel. Die eenheid 'kW' moet ook vertoon word.

Voorbeeld van afvoer:

Die maksimum opwekkingskapasiteit per dag in die
 Winter:
 57.456 kW

(4)

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

TOTAAL AFDELING C: 40

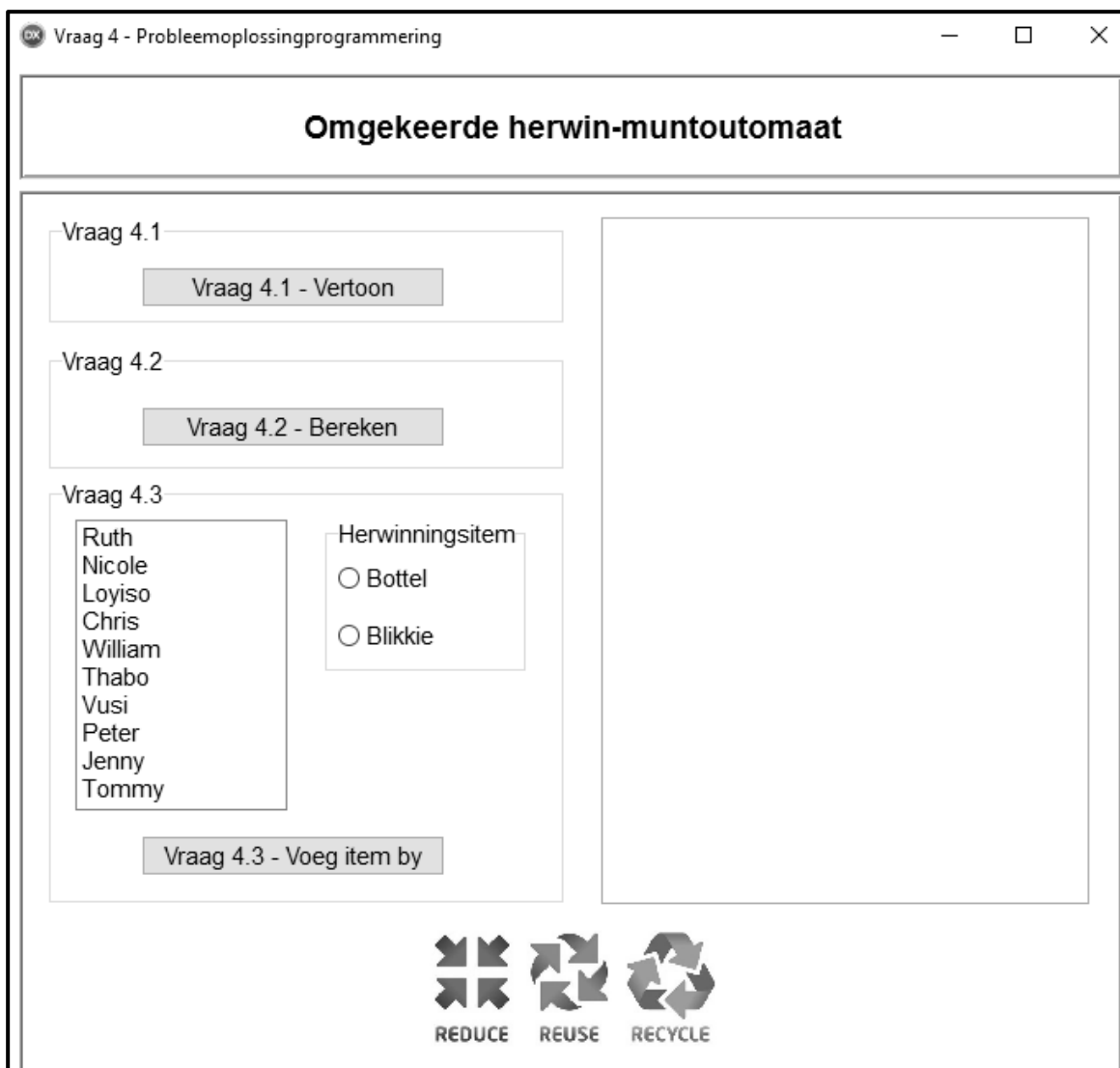
AFDELING D**VRAAG 4: PROBLEEMOPLOSSINGPROGRAMMERING**

Muntoutomate kan gebruik word om items soos lekkergoed, skyfies, koeldrank, ens. uit te gee. 'n Omgekeerde muntoutomaat kan vir herwinningsdoeleindes gebruik word. Jou skool wil omgekeerde muntoutomate ('reversed vending machines') op die skoolgrond gebruik om bottels en blikkies te herwin. 'n Leerder sal 'n aantal bottels of blikkies in die omgekeerde muntoutomaat deponeer/ingooi en sal dan vergoed word. Een van die omgekeerde muntoutomate sal in 'n klas met 10 leerders getoets word.

Doen die volgende:

- Maak die onvolledige program in die **Vraag 4-lêergids** oop.
- Sleutel jou eksamennummer as kommentaar in die eerste reël van die **Vraag4_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.

Voorbeeld van grafiese gebruikerskoppelvlak (GGK):



Die volgende skikkings is in die program voorsien:

- 'n Skikking **arrName** wat die name van tien leerders in 'n klas bevat:

```
arrName: array [1 .. 10] of String = (
    'Ruth', 'Nicole', 'Loyiso', 'Chris', 'William',
    'Thabo', 'Vusi', 'Peter', 'Jenny', 'Tommy');
```

- 'n Tweedimensionele skikking, **arrOutomaat**, wat die tipe items wat deur die leerders herwin is, stoor. Die skikking is gedeeltelik gevul en laat toe dat tot 15 items deur elk van die 10 leerders in die klas herwin kan word:

```
arrOutomaat: array [1 .. 10, 1 .. 15] of String =
    (('K', '', '', '', '', '', '', '', '', '', '', '', '', '', ''),
    ('B', 'B', 'B', 'K', 'K', 'K', 'B', 'B', 'B', 'K', 'K', 'K', 'K',
    'K', ''),
    ('', '', '', '', '', '', '', '', '', '', '', '', '', '', ''),
    ('K', 'K', '', '', '', '', '', '', '', '', '', '', '', '', ''),
    ('B', 'B', 'K', 'K', 'B', 'B', 'K', 'K', 'K', 'K', 'B', 'K', 'K',
    'B', ''),
    ('K', 'K', 'B', '', '', '', '', '', '', '', '', '', '', '', ''),
    ('K', 'B', '', '', '', '', '', '', '', '', '', '', '', '', ''),
    ('B', 'B', '', '', '', '', '', '', '', '', '', '', '', '', ''),
    ('K', '', '', '', '', '', '', '', '', '', '', '', '', '', ''),
    ('B', 'K', '', '', '', '', '', '', '', '', '', '', '', '', ''));
```

Voltooi die kode vir elke afdeling van VRAAG 4, soos in VRAAG 4.1 tot VRAAG 4.3 hieronder beskryf word.

4.1 Knoppie [4.1 - Vertoon]

Die name van leerders is in die skikking **arrName** voorsien. 'n Ooreenstemmende lys met items wat vir elke leerder herwin is, is in die skikking **arrOutomaat** voorsien.

Die items in die skikking **arrOutomaat** word voorgestel deur die letters 'B' of 'K' te gebruik, waar 'B' 'n bottel en 'K' 'n blikkie verteenwoordig.

Kode is voorsien om die opskrif te vertoon.

Skryf kode om die inhoud van skikkings **arrName** en **arrOutomaat** in die volgende formaat te vertoon:

```
<Naam><tab><herwinde item><herwinde item><herwinde item>...
```

Voorbeeld van afvoer:

Name	Herwinde items
Ruth	K
Nicole	BBBKBBBKBBBK
Loyiso	
Chris	KK
William	BBKBBBKBBBKBB
Thabo	KKB
Vusi	KB
Peter	BB
Jenny	K
Tommy	BK

(5)

4.2 Knoppie [4.2 - Bereken]

Leerders wat die muntoutomaat gebruik om items te herwin, sal soos volg vir elke item vergoed word:

ITEM WAT HERWIN IS	BEDRAG BETAAL
Bottel (B)	R2.15
Blikkie (K)	R0.75

Skryf kode om die volgende in die 'rich edit' **redV4** te vertoon:

- Die naam van elke leerder en die totale bedrag wat elke leerder gaan ontvang vir die bottel- en blikkie-items wat die leerder in die muntoutomaat gesit het.
- Die naam (name) van die leerder(s) wat die grootste bedrag gaan ontvang.

Voorbeeld van afvoer:

Name	Totale bedrag uitbetaal
Ruth	R0.75
Nicole	R18.90
Loyiso	R0.00
Chris	R1.50
William	R18.90
Thabo	R3.65
Vusi	R2.90
Peter	R4.30
Jenny	R0.75
Tommy	R2.90
Grootste uitbetaling(s):	
Nicole	R18.90
William	R18.90

(14)

4.3 Knoppie [4.3 - Voeg item by]

Die gebruiker moet die volgende doen as 'n leerder 'n item wil byvoeg om herwin te word:

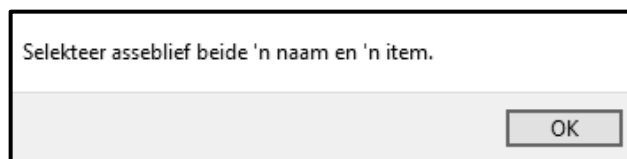
- Selekteer die naam van die leerder uit die lysblokkie ('list box') **IstV4**.
- Selekteer die tipe item wat vir herwinning uit die radiogroep **rgpV4** bygevoeg moet word.

Die geselekteerde bottel- of blikkie-item moet dan by die volgende beskikbare leë ruimte in die skikking **arrOutomaat** gevoeg word, by die indeks wat met die naam van die geselekteerde leerder ooreenstem. 'n Geskikte boodskap moet vertoon word as die muntoutomaat-ruimte vir die gespesifiseerde leerder vol is.

Skryf kode om die volgende te doen:

- Verkry die naam van die leerder uit die lysblokkie **IstV4** en die item wat herwin moet word uit die radiogroep **rgpV4**.
- Vertoon 'n geskikte boodskap as nóg die naam nóg die item geselekteer is.
- As 'n oop ruimte in die skikking vir die spesifieke leerder beskikbaar is:
 - Voeg die geselekteerde herwinde item (B of K) by die skikking **arrOutomaat** by die korrekte indeks vir die leerder wat geselekteer is.
 - Vertoon die inhoud van die skikking **arrName** en die opgedateerde skikking **arrOutomaat** nadat elke item bygevoeg is.
- As daar geen spasie in die skikking vir die gespesifiseerde leerder beskikbaar is nie, vertoon 'n geskikte boodskap wat aandui dat die muntoutomaat vol is.

Voorbeeld van afvoer as 'n naam of item nie geselekteer is nie:



Voorbeeld van afvoer as die naam 'Nicole' en die item 'Bottel' geselekteer is:



Name	Herwinde items
Ruth	K
Nicole	BBBBKKBBBBKKKKKB
Loyiso	
Chris	KK
William	BBKKBBBBKKKBKKB
Thabo	KKB
Vusi	KB
Peter	BB
Jenny	K
Tommy	BK

Voorbeeld van afvoer as 'n poging aangewend is om nog 'n item vir die naam 'Nicole' by te voeg:



(11)

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

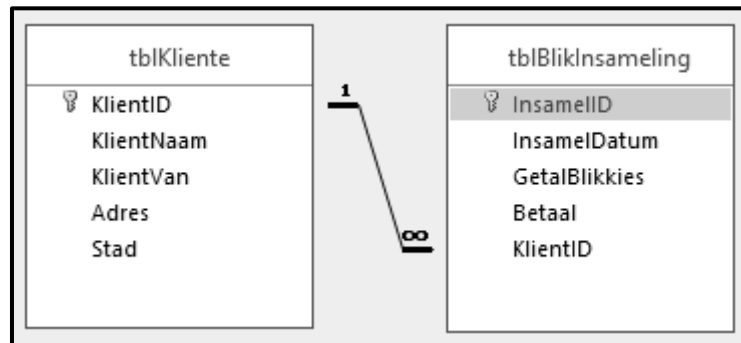
TOTAAL AFDELING D: 30
GROOTTOTAAL: 150

INLIGTINGSTEGNOLOGIE V1

DATABASIS-INLIGTING VRAAG 2:

Die databasis **InsamelingDB** bestaan uit tabel **tbKliente** en **tbBlikInsameling**.

Die volgende een-tot-baie-verwantskap met referensiële integriteit ('referential integrity') bestaan tussen die twee tabelle in die databasis:



Die ontwerp van die databasistabelle is soos volg:

Tabel: **tbKliente**

Hierdie tabel bevat besonderhede van die kliënte.

Veldnaam	Datatype	Beskrywing
KlientID	Text (5)	Unieke ID vir die kliënt
KlientNaam	Text (15)	Die naam van die kliënt
KlientVan	Text (15)	Die van van die kliënt
Adres	Text (20)	Die adres van die kliënt wat vir die oplaai van blikkies gebruik word
Stad	Text (15)	Die stad waar die kliënt woon

Voorbeeld van die rekords in die **tbKliente**-tabel:

KlientID	KlientNaam	KlientVan	Adres	Stad
ABI10	Prashant	Govender	Bergstraat 72	Kimberley
BUS06	Busi	Nkosi	Donaldweg 65	Welkom
CHR08	Chris	Ferreira	Richmondstraat 188	Potchefstroom
DAM07	Damian	Coetzer	Kaaprylaan 12	Durban
GER01	Gert	Vermeulen	Dawnstraat 55	Bloemfontein
HEN11	Henry	Marques	Kingsleyrylaan 1	Kaapstad
JAC05	Jacob	Human	Humanstraat 8	Bloemfontein
JOH03	Johan	Weston	Michellinstraat 43	Vanderbijlpark
PHI04	Phillip	Brown	Parkweg 11	Sasolburg
PIE12	Piet	Mogorosi	Stormweg 5	Bloemfontein
RHO09	Rhoda	Somers	Marilynstraat 14	Bloemfontein
WIL12	Willem	de Wit	Arrowstraat 2	Sasolburg

Tabel: **tblBlikInsameling**

Hierdie tabel bevat inligting van die al die insamelings.

Veldnaam	Datatype	Beskrywing
InsamelID	Text (5)	Unieke kode vir elke insameling
InsamelDatum	Date/Time	Datum van die insameling
GetalBlikkies	Number	Heelgetal-waarde wat die getal blikkies wat ingesamel is, aandui
Betaal	Boolean	'n Boolese waarde wat aandui dat die kliënt betaling ontvang het vir die blikkies wat ingesamel is
KlientID	Text (5)	Die ID van die kliënt wat die blikkies ingesamel het

Voorbeeld van die eerste tien rekords in die **tblBlikInsameling**-tabel:

InsamelID	InsamelDatum	GetalBlikkies	Betaal	KlientID
C001	2020/01/19	412	<input checked="" type="checkbox"/>	WIL12
C002	2020/03/21	300	<input type="checkbox"/>	GER01
C003	2020/03/23	599	<input checked="" type="checkbox"/>	WIL12
C004	2020/03/25	514	<input checked="" type="checkbox"/>	WIL12
C005	2020/03/26	1200	<input type="checkbox"/>	CHR08
C006	2020/03/30	480	<input type="checkbox"/>	DAM07
C007	2020/04/02	511	<input checked="" type="checkbox"/>	GER01
C008	2020/04/15	200	<input type="checkbox"/>	BUS06
C009	2020/04/17	419	<input type="checkbox"/>	DAM07
C010	2020/04/24	500	<input checked="" type="checkbox"/>	CHR08

LET WEL:

- Konneksiekode is gegee.
- Die databasis word met 'n wagwoord beskerm, dus sal jy nie direkte toegang tot databasis kan verkry nie.



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NASIONALE
SENIOR SERTIFIKAAT**

GRAAD 12

INLIGTINGSTEGNOLOGIE V1

NOVEMBER 2022

NASIENRIGLYNE

PUNTE: 150

Hierdie nasienriglyne bestaan uit 29 bladsye.

ALGEMENE INLIGTING:

- Hierdie nasienriglyne moet as die basis vir die nasiensessie gebruik word. Dit is voorberei om deur nasieners gebruik te word. Daar word na alle nasieners verwag om 'n deeglike standaardiseringsvergadering by te woon om seker te maak dat die riglyne konsekwent geïnterpreteer en tydens die nasien van die kandidate se werk toegepas word.
- Let op dat leerders wat 'n alternatiewe korrekte oplossing as wat as voorbeeld van 'n oplossing in die nasienriglyne gegee word verskaf, volle krediet vir die relevante oplossing moet kry tensy die spesifieke instruksies in die vraestel nie gevolg is nie of die vereistes van die vraag nie nagekom is nie.
- **Bylaag A, B, C en D** (bladsy 3 tot 11) sluit die nasienrubriek vir elk van die vrae.
- **Bylaag E, F, G en H** (bladsy 12 tot 29) bevat voorbeelde van oplossings vir Vraag 1 tot Vraag 4 in programmeringskode.
- Kopieë van **Bylaag A, B, C, D** en **die opsomming van die leerder se punte** (bladsy 3 tot 11) moet vir elke leerder gemaak word en tydens die nasiensessie voltooi word.

1.3	<p>Knoppie [1.3 – Bereken]</p> <p>Verkry A en B uit edit boxes ✓ skakel om na real ✓</p> <p>$C := \text{sqrt}(\text{power}(A, 5) + \text{pi} * \text{sqr}(B));$ ✓</p> <p>Alternatiewe vir funksies</p> <ul style="list-style-type: none"> • Sqrt of Power(Waarde, 0.5) • Power (moet nie harde kodering aanvaar nie) • Pi of 22/7 of 3.14 • Sqr of Power(B, 2) of (B * B) <p>Vertoon afgekapte (Floor of Trunc) waarde van C omgeskakel na String</p> <p>NOTA: Penaliseer slegs een keer vir 'n logika fout</p>	8	
1.4	<p>Knoppie [1.3 – Punte]</p> <p>Inisialiseer teller en totaal ✓ Sleutel punt in ✓ Begin voorwaardelike lus om te toets vir $<> -1$ ✓ Vermeerder teller-veranderlike ✓ Tel punt by totale punt ✓ Vertoon die nommer van die vak en punt ✓ Sleutel volgende punt in ✓</p> <p>Bereken gemiddelde punt ✓ Vertoon gemiddelde punt in afvoerarea omgeskakel na String geformateer na TWEE desimale plekke ✓</p> <p>NOTAS:</p> <ul style="list-style-type: none"> • As <i>repeat until</i> gebruik is, is die toestand: until punt = -1 • As 'n enkel invoer in lus gebruik word, moet 'n IF binne die lus gebruik word om te voorkom dat -1 as 'n punt aanvaar word. 	11	

1.5	<p>Button - [1.5 – Anagram]</p> <p>As die lengte van die twee woord-stringe dieselfde is ✓ Lus ✓ van 1 tot lengte van woord1-string ✓ Gebruik karakter as indeks ✓ Kry posisie ✓ van karakter in woord2 ✓ Vee die karakter by posisie in woord2 uit ✓</p> <p>As die lengte van woord2 nul is ✓ Vertoon die boodskap “Die woorde vorm ‘n anagram.” ✓ Anders ✓ Vertoon die boodskap “Die woorde vorm nie ‘n anagram nie.” ✓</p> <p>Konsepte:</p> <p>Lus deur die lengte van die eerste woord //2 Onttrek karakter by indeks //1 Vind posisie van karakter in tweede woord //2</p> <p><u>Meganisme om woorde te vergelyk:</u> //3</p> <ul style="list-style-type: none"> • Skryf karakter by indeks van woord1 en karakter posisie in woord 2 met spasie • Toets en verander vlag as Nie gevind is (stel vlag voor lus) • Verwyder karakter uit tweede woord (as lengte getoets is en dieselfde is voor die lus) <p><u>Toets vir anagram geslaag en vertoon boodskap:</u> //3 As woorde gelyk / woord2 = " // vlag steeds waar ShowMessage IS anagram ShowmMssage NIE anagram</p> <p>Sortering as ‘n alternatief:</p> <p>Sorteer karakters in eerste woord: Dubbel lusse //2 Vergelyk karakters binne die woord //2 Ruil karakters //3</p> <p>Sorteer karakters in tweede woord //1</p> <p><u>Toets en vertoon boodskap:</u> //3 AS woorde gelyk is ShowMessage IS anagram ShowMessage NIE anagram</p>	11	
	TOTAAL AFDELING A:	40	

VRAAG 2: NASIENRUBRIEK – VERVOLG

2.2	DATABASISMANIPULASIE deur Delphi-kode te gebruik		
2.2.1	<p>Knoppie [2.2.1 – Voeg in]</p> <p>Insert/append mode ✓ Ken aan regte veldname toe ✓ Deur die regte waardes vir elke veld te gebruik ✓ Post ✓</p> <pre>tblKliente.Insert; tblKliente['KlientID'] := 'CHA01'; tblKliente['KlientNaam'] := 'Charles'; tblKliente['KlientVan'] := 'du Boit'; tblKliente['Adres'] := 'Van Wouwstraat 24'; tblKliente['Stad'] := 'Kaapstad'; tblKliente.Post;</pre>	4	
2.2.2	<p>Knoppie [2.2.2 – Persentasie]</p> <p>Vertoon die naam en van van die geselekteerde kliënt ✓ Instansieer veranderlikes ✓</p> <p>Gaan na die eerste rekord in die tblBlikInsameling-tabel ✓ Gebruik lus om deur die tblBlikInsameling-tabel te stap ✓ Toets of jaar = geselekteerde jaar ✓ Inkrementeer maatskappy se totale insamelings vir die jaar ✓ As KlientID in tblBlikInsameling = KlientID in tblKliente ✓ Inkrementeer kliënt se totale insamelings vir die jaar ✓ Gaan na volgende rekord in tblBlikInsameling ✓ <i>Eindig lus (tblBlikInsameling)</i></p> <p>Bereken persentasie van kliënt se insameling as deel van maatskappy se totale insamelings ✓ $(r\text{Persentasie} = i\text{KlientTotaal} / i\text{MaatskappyTotaal} * 100)$ Vertoon totale kliënt-insamelingswaarde geformateer na String ✓ Vertoon totale maatskappy-insamelingswaarde geformateer na String ✓ Vertoon persentasie van kliënt se insameling geformateerde tot twee desimale plekke met die %-simbool by die resultaat gevoeg ✓</p>	13	
	Subtotaal:	17	
	TOTAAL AFDELING B:	40	

BYLAAG C**VRAAG 3: NASIENRUBRIEK - OBJEK-GEÖRIENTEERDE PROGRAMMERING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
3.1.1	Konstruktor Create Konstruktor-opskrif met drie regte parameters ✓ met regte data tipes vir parameters ✓ Ken parameterwaarde toe aan die drie attribute ✓✓ (fAanlegKode, fGetalPanele, fKragPerPaneel) Stel seisoen-attribuut op "Somer" ✓	5	
3.1.2	Procedure veranderGetalPanele Prosedure-opskrif ✓ met heelgetalwaarde parameter ✓ fGetalPanele := fGetalPanele ✓ + parameter ✓	4	
3.1.3	Procedure setSeisoen Prosedure-opskrif ✓ met heelgetalwaarde parameter ✓ fSeisoen = parameter ✓	3	
3.1.4	funksie berekenKapasiteit funksie-opskrif met real datatipe ✓ Toets of fSeisoen = 'Somer' ✓ ure = 10 ✓ anders as fSeisoen = 'Winter' ✓ ure = 6 ✓ anders ure = 8 ✓ Result = ✓ fGetalPanele * fKragPerPaneel * ure ✓	8	
3.1.5	Funksie toString Funksie verklaar met string as terugstuurtype ✓ Al die attribute deel van string wat teruggestuur word ✓ In regte formaat met regte teks en met #13 ✓	3	
	Subtotaal: Objekklas	23	

VRAAG 3: NASIENRUBRIEK (VERVOLG)

VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER - PUNT
3.2.1	<p>Knoppie [3.2.1 – Instansieer objek]</p> <p>Verkry die data uit die komponente ✓ converted to correct data types ✓</p> <p>objAanleg := ✓ TsonkragAanleg.create ✓ Gebruik drie argumente ✓ (AanlegKode, GetalPanele, KragPerPaneel)</p> <p>Vertoon inligting in die rich edit deur toString-metode te gebruik ✓</p>	6	
3.2.2	<p>Knoppie [3.2.2 – Vermeerder panele]</p> <p>Roep inkGetalPanele-metode ✓ deur waarde van spin edit as argument te gebruik ✓</p> <p>Vertoon aanlegkode deur die getAanlegKode-metode te gebruik ✓ en getal panele deur die getGetalPanele-metode te gebruik, omgeskakel na string ✓</p>	4	
3.2.3	<p>Knoppie [3.2.3 – Dateer seisoen op]</p> <p>Roep setSeisoen-metode ✓ deur die seisoen wat in die combobox geselekteer is as argument te gebruik ✓</p> <p>Vertoon inligting in die rich edit deur toString-metode te gebruik ✓</p>	3	
3.2.4	<p>Knoppie [3.2.4 – Bereken]</p> <p>Vertoon 'n geskikte beskrywing ✓ met die seisoen, deur die getSeisoen-metode in die regte formaat te gebruik ✓</p> <p>Vertoon resultaat van die berekenKapasiteit-metode ✓, omgeskakel na string met "kW" as eenheid bygevoeg ✓</p>	4	
	Subtotaal: Vormeenheidklas	17	
	TOTAAL AFDELING C:	40	

BYLAAG D**VRAAG 4: NASIENRUBRIEK – PROBLEEMOPLOSSING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER - PUNT
4.1	<p>Knoppie [4.1 – Vertoon]</p> <p>Buitenste lus I van 1 tot lengte van skikking ✓ Of 10 Begin afvoer string met naam[I] ✓ en voeg tab (#9) by afvoer string</p> <p>Binneste lus J van 1 tot lengte van arrOutomaat[J] ✓ Of 15 Voeg karakters uit arrOutomaat by afvoerstring ✓ Vertoon afvoerstring in rich edit ✓</p>	5	
4.2	<p>Knoppie [4.2 – Bereken]</p> <p>Inisialiseer Maks-variable na 0 ✓ Buitenste lus I van 1 tot lengte van arrName (10) ✓ Inisialiseer bedragBetaal na 0 ✓ Binneste lus J van 1 tot lengte van arrOutomaat[J] (15) ✓ Toets of herwin-item 'n bottle is ✓ Inkrementeer bedragBetaal met 2.15 ✓ Toets of herwin-item 'n blikkie is Inkrementeer bedragBetaal met 0.75 ✓ <i>Eindig binneste lus</i></p> <p>Vertoon naam en bedragBetaal, omgeskakel na geldeenheid ✓</p> <p>Alternatief 1: Toets of BedragBetaal >= Maks ✓ As BedragBetaal > Maks ✓ Stel Maks na BedragBetaal ✓ en maak afvoer skoon ✓ Voeg naam en bedrag na Maks afvoer string ✓ <i>Eindig buitenste lus</i></p> <p>Vertoon Maks afvoer string in redQ4 ✓</p> <p>Alternatief 2 (gebruik parallelle skikking)</p> <p>Toets of BetaalBedrag > Maks //1 Stel Maks na BedragBetaal //1 Stoor BedragBetaal in parallelle skikking //1 <i>Eindig buitenste lus</i></p> <p>Lus van 1 tot 10 //1 As BetaalBedrag = Maks //1 Vertoon naam en BetaalBedrag //1</p>	14	

4.3	<p>Knoppie [4.2 – Voeg item by]</p> <p>Toets of naam OF item NIE gekies is nie ✓ Vertoon boodskap as naam of item nie gekies is nie ✓</p> <p>Kry item van komponent ✓ Kry OF bepaal ry(indeks) van naam ✓</p> <p><u>PLASING VAN ITEM:</u> Lus indeks van 1 tot 15 ✓ Toets of arrOutomaat leeg is by indeks ✓ Plaas item in arrOutomaat ✓ Vlag / Breek uit ✓</p> <p>Vertoon opgedateerde skikking ✓</p> <p><u>GEEN SPASIE BOODSKAP:</u> Toets vlag OF gebruik 'n ander meganisme om te bepaal of spasie beskikbaar is ✓ Vertoon boodskap as geen spasie beskikbaar is nie ✓</p>	11	
-----	---	-----------	--

TOTAAL AFDELING D:	30	
GROOTTOTAAL:	150	

OPSOMMING VAN LEERDER SE PUNTE:

SENTRUMNOMMER:		LEERDER SE EKSAMENNOMMER:			
	AFDELING A	AFDELING B	AFDELING C	AFDELING D	
	VRAAG 1	VRAAG 2	VRAAG 3	VRAAG 4	GROOT-TOTAAL
MAKS. PUNTE	40	40	40	30	150
LEERDER SE PUNTE					

BYLAAG E: OPLOSSING VIR VRAAG 1

```
//=====
// Vraag 1.1 3 punte
//=====
procedure TfrmVraag1.btnV1_1Click(Sender: TObject);
begin
  // Vraag 1.1
  pnlV1_1.Color := clYellow;
  pnlV1_1.Font.Style := [fsItalic];
  pnlV1_1.Caption := 'Ek hou van programmering!';
end;
//=====
// Vraag 1.2 7 punte
//=====
procedure TfrmVraag1.btnV1_2Click(Sender: TObject);
begin
  // Vraag 1.2
  if (cmbV1_2.ItemIndex > 4) AND (cmbV1_2.ItemIndex < 8) then
    begin
      lblV1_2.Caption := 'Maatskappy gesluit, selekteer ' +
        'ander maand.';
      cmbV1_2.ItemIndex := -1;
    end
  else
    lblV1_2.Caption := 'Jou verlof in ' + cmbV1_2.text + ' is
      toegestaan.';
end;
//=====
// Vraag 1.3 8 punte
//=====
procedure TfrmVraag1.btnV1_3Click(Sender: TObject);
var
  rA, rB, rC : Real;
begin
  // Vraag 1.3
  rA := StrToFloat(edtV1_3_1.Text);
  rB := StrToFloat(edtV1_3_2.Text);
  rC := Sqrt(Power(rA,5)+ pi * Sqr(rB));

  edtV1_3_3.Text:= IntToStr(Trunc(rC));
end;
//=====
// Vraag 1.4 11 punte
//=====
procedure TfrmVraag1.btnV1_4Click(Sender: TObject);
var
  iPunt, i, iTotaal : integer;
  rGemiddeld : Real;
begin
  // Kode wat voorsien is
  redV1_4.Clear;
```

```

// Vraag 1.4
iPunt := StrToInt(InputBox('Leerderpunte','Sleutel punt in vir vak:
                            1',''));

i := 0;
iTotaal := 0;
while iPunt > -1 do
  begin
    Inc(i);
    iTotaal := iTotaal + iPunt;
    redV1_4.Lines.Add('Vak ' + IntToStr(i) + ': ' + IntToStr(iPunt));
    iPunt := StrToInt(InputBox('Leerderpunte','Sleutel punt in vir
                                vak: ' + IntToStr(i+1), ''));

  end;
rGemiddeld:=iTotaal / i;
redV1_4.Lines.Add('Gemiddelde punt: ' +
                  FloatToStrF(rGemiddeld,ffFixed,5,2));

end;
// =====
// Vraag 1.5                                11 punte
// =====
procedure TfrmVraag1.btnV1_5Click(Sender: TObject);
var
  sWoord1,sWoord2 : String;
  i : integer;
begin
  //Kode wat voorsien is
  memV1_5.Clear;
  sWoord1 := Lowercase(edtV1_5_1.Text);
  sWoord2 := Lowercase(edtV1_5_2.Text);

  // Vraag 1.5

  if length(sWoord1) = length(sWoord2) then
  begin
    for i := 1 to length(sWoord1) do
      delete(sWoord2, (pos(sWoord1[i],sWoord2)),1);
    if sWoord2 = '' then
      memV1_5.Lines.add( 'Die woorde vorm ''n anagram.')
    else
      memV1_5.Lines.add('Die woorde vorm nie ''n anagram nie.');
  end
else
  memV1_5.Lines.add('Die woorde vorm nie ''n anagram nie.');
```

BYLAAG F: OPLOSSING VIR VRAAG 2

```

unit Vraag2_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms, Dialogs, DB, ExtCtrls, StdCtrls,
  ComCtrls, Grids, DBGrids, Buttons, ADODB, Math, DateUtils,
  ConnectDB_U;

type
  TfrmVraag2 = class(TForm)
    pgcDBAdmin: TPageControl;
    tabsV2SQL: TTabSheet;
    btnV2_1_5: TBitBtn;
    grpResultate: TGroupBox;
    btnV2_1_1: TBitBtn;
    btnV2_1_2: TBitBtn;
    tabsVDelphi: TTabSheet;
    grpV2_2_1: TGroupBox;
    redV2_2_2: TRichEdit;
    btnV2_2_2: TButton;
    dbgKliente: TDBGrid;
    dbgInsamelings: TDBGrid;
    grpV2_2_2: TGroupBox;
    btnV2_2_1: TButton;
    btnV2_1_4: TBitBtn;
    pnlBtns: TPanel;
    bmbClose: TBitBtn;
    bmbHerwinDB: TBitBtn;
    dbgrdSQL: TDBGrid;
    btnV2_1_3: TButton;
    rgpV2_2_2: TRadioGroup;
    procedure btnV2_1_1Click(Sender: TObject);
    procedure btnV2_1_2Click(Sender: TObject);
    procedure btnV2_1_4Click(Sender: TObject);
    procedure btnV2_1_3Click(Sender: TObject);
    procedure btnV2_1_5Click(Sender: TObject);
    procedure btnV2_2_2Click(Sender: TObject);
    procedure btnV2_2_1Click(Sender: TObject);
    procedure FormShow(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure bmbHerwinDBClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag2: TfrmVraag2;
  dbCONN : TConnection;

```

```
// --- Global variables provided ---
tblKliente, tblBlikInsameling : TADOTable;

implementation

{$R *.dfm}

procedure TfrmVraag2.bmbHerwinDBClick(Sender: TObject);
begin
// Herwin Databasis
dbCONN.RestoreDatabase;
redV2_2_2.Clear;
dbCONN.SetupGrids(dbgKliente, dbgInsamelings, dbgrdSQL);
end;

// =====
// Vraag 2.1.1 4 punte
// =====

procedure TfrmVraag2.btnV2_1_1Click(Sender: TObject);
var
sSQL1: String;
begin
// Vraag 2.1.1

sSQL1 := 'SELECT * FROM tblKliente ' +
'WHERE Stad = "Bloemfontein" ' +
'ORDER BY KlientVan';

// Kode wat voorsien is - moenie verander nie
dbCONN.runSQL(sSQL1);
end;

// =====
// Vraag 2.1.2 3 punte
// =====

procedure TfrmVraag2.btnV2_1_2Click(Sender: TObject);
var
sSQL2: String;
begin
// Vraag 2.1.2

sSQL2 := 'SELECT InsamelID, InsamelDatum, GetalBlikkies ' +
'FROM tblBlikInsameling ' +
'WHERE Month(InsamelDatum) = 1';

// Kode wat voorsien is - moenie verander nie
dbCONN.runSQL(sSQL2);
end;
```

```
// =====
// Vraag 2.1.3 4 punte
// =====
```

```
procedure TfrmVraag2.btnV2_1_3Click(Sender: TObject);
var
    sSQL3, sLetter : String;
begin
    // Vraag 2.1.3

    // Kode wat voorsien is
    sLetter := InputBox('', 'Sleutel eerste letter van KlientID in', '');

    sSQL3 := 'SELECT * FROM tblKliente ' + ✓
            'WHERE KlientID LIKE ' + sLetter + '%"; ✓

    { Alternative:
      WHERE left(KlientID, 1) = quotedStr(sLetter) }

    // Kode wat voorsien is - moenie verander nie
    dbCONN.runSQL(sSQL3);
end;
```

```
// =====
// Vraag 2.1.4 8 punte
// =====
```

```
procedure TfrmVraag2.btnV2_1_4Click(Sender: TObject);
var
    sSQL4: String;
begin
    // Vraag 2.1.4

    sSQL4 := 'SELECT KlientNaam, ' +
            'FORMAT(SUM(GetalBlikkies)/76*8,"Currency") ' +
            'AS [Totale Bedrag] ' +
            'FROM tblBlikInsameling A, tblKliente B ' +
            'WHERE A.KlientID = B.KlientID ' +
            'GROUP BY KlientNaam';

    // Kode wat voorsien is - moenie verander nie
    dbCONN.runSQL(sSQL4);
end;
```

```
// =====
// Vraag 2.1.5 4 punte
// =====
```

```
procedure TfrmVraag2.btnV2_1_5Click(Sender: TObject);
var
    sSQL5 : String;
    bChange : boolean;
begin
    // Vraag 2.1.5
```

```
sSQL5 := 'UPDATE tblBlikInsameling ' +
        'SET GetalBlikkies = 250, Betaal = False ' +
        'WHERE InsamelID = "C003"';

// Kode wat voorsien is - moenie verander nie
dbCONN.ExecuteSQL(sSQL5, bChange);

if bChange then
begin
    MessageDlg('Databasis is opgedateer.', mtInformation, [mbOK], 0);
end;
end;

// =====
// Vraag 2.2.1                4 punte
// =====

procedure TfrmVraag2.btnV2_2_1Click(Sender: TObject);
begin
// Vraag 2.2.1

tblKliente.Insert;
tblKliente['KlientID'] := 'CHA01';
tblKliente['KlientNaam'] := 'Charles';
tblKliente['KlientVan'] := 'du Boit';
tblKliente['Adres'] := 'Van Wouwstraat 24';
tblKliente['Stad'] := 'Kaapstad';
tblKliente.Post;
end;

// =====
// Vraag 2.2.2                13 punte
// =====

procedure TfrmVraag2.btnV2_2_2Click(Sender: TObject);
var
    iMaatskappyTotaal, iKlientTotaal : integer;
    sJaar : String;
    rPersentasie : real;
begin
// Kode wat voorsien is - moenie verander nie
    redV2_2_2.Clear;
    sJaar := rgpV2_2_2.Items[rgpV2_2_2.ItemIndex];
// =====
// Vraag 2.2.2

    redV2_2_2.Lines.Add(tblKliente['KlientNaam'] + ' ' +
                        tblKliente['KlientVan']+#13);

    iMaatskappyTotaal := 0;
    iKlientTotaal := 0;
    tblBlikInsameling.First;
    while NOT (tblBlikInsameling.Eof) do
        begin
            if sJaar = IntToStr(YearOf(tblBlikInsameling['InsamelDatum'])) then
                begin
```

```
        iMaatskappyTotaal := iMaatskappyTotaal +
                                tblBlikInsameling['GetalBlikkies'];
    if (tblBlikInsameling['KlientID'] = tblKliente['KlientID'])
    then
        iKlientTotaal := iKlientTotaal +
                                tblBlikInsameling['GetalBlikkies'];
    end;
tblBlikInsameling.Next;
end;
rPersentasie := iKlientTotaal / iMaatskappyTotaal * 100;
redV2_2_2.Lines.Add('Kliënt vir ' + sJaar + ' ingesamel: ' + #9 +
                    IntToStr(iKlientTotaal));
redV2_2_2.Lines.Add('Maatskappy vir ' + sJaar + ' se ingesamel: ' +
                    #9 + IntToStr(iMaatskappyTotaal));
redV2_2_2.Lines.Add('Kliënt se persentasie ingesamel: ' + #9 +
                    FloatToStrF(rPersentasie, ffFixed, 3, 2) + '%');
end;

procedure TfrmVraag2.FormClose(Sender: TObject; var Action:
TCloseAction);
begin // Disconnects from database and closes all open connections
    dbCONN.dbDisconnect;
end;

procedure TfrmVraag2.FormCreate(Sender: TObject);
begin
    redV2_2_2.Paragraph.TabCount := 2;
    redV2_2_2.Paragraph.Tab[0] := 180;
    redV2_2_2.Paragraph.Tab[1] := 150;
end;

procedure TfrmVraag2.FormShow(Sender: TObject);
begin
    // Sets up the connection to database and opens the tables.
    dbCONN := TConnection.Create;
    dbCONN.dbConnect;
    tblKliente := dbCONN.tblOne;
    tblBlikInsameling := dbCONN.tblMany;
    dbCONN.setupGrids(dbgKliente, dbgInsamelings, dbgrdSQL);
    pgcDBAdmin.ActivePageIndex := 0;
end;

end.
```

BYLAAG G: OPLOSSING VIR VRAAG 3**Objekklas:**

```
unit SonkragAanleg_U;

interface

type
  TSonkragAanleg = class(TObject)
  private
    var
      fAanlegKode: String;
      fGetalPanele: Integer;
      fKragPerPaneel: Real;
      fSeisoen: String;
  public
    // Kode wat voorsien is
    function getAanlegKode : String;
    function getGetalPanele : Integer;
    function getSeisoen : String;
    // Kodeer hier

    constructor create(sAanlegKode: String; iGetalPanele: Integer;
                      iKragPerPaneel:Real);
    procedure inkGetalPanele(iGetal : Integer);
    procedure setSeisoen(sNuweSeisoen : String);
    function berekenKapasiteit : Real;
    function toString: String;
  end;

implementation

uses
  SysUtils, Math;

{ TSonkragAanleg }

// =====
// Vraag 3.1.1          5 punte
// =====

constructor TSonkragAanleg.create(sAanlegKode: String;
  iGetalPanele: Integer; iKragPerPaneel: Real);
begin
  fAanlegKode := sAanlegKode;
  fGetalPanele := iGetalPanele;
  fKragPerPaneel := iKragPerPaneel;
  fSeisoen := 'Sommer';
end;
```

```
// =====  
// Vraag 3.1.2           4 punte  
// =====  
  
procedure TsonkragAanleg.inkGetalPanele(iGetal: Integer);  
begin  
    fGetalPanele := fGetalPanele + iGetal;  
end;  
  
// =====  
// Vraag 3.1.3           3 punte  
// =====  
  
procedure TsonkragAanleg.setSeisoen(sNuweSeisoen: String);  
begin  
    fSeisoen := sNuweSeisoen;  
end;  
  
// =====  
// Vraag 3.1.4           8 punte  
// =====  
  
function TsonkragAanleg.berekenKapasiteit: Real;  
var  
    rTotaal : Real;  
    iUre : Integer;  
begin  
    if fSeisoen = 'Sommer' then  
        begin  
            iUre := 10;  
        end  
    else  
        if fSeisoen = 'Winter' then  
            begin  
                iUre := 6;  
            end  
        else  
            begin  
                iUre := 8;  
            end;  
        end;  
  
    result := (fGetalPanele * fKragPerPaneel) * iUre;  
end;
```

```
// =====  
// Vraag 3.1.5          3 punte  
// =====  
  
function TsonkragAanleg.toString: String;  
var  
    sString : String;  
begin  
    sString := 'Aanlegkode: ' + fAanlegKode + #13;  
    sString := sString + 'Getal panele: ' + IntToStr(fGetalPanele) + #13;  
    sString := sString + 'Krag per paneel: ' + floatToStr(fKragPerPaneel)  
+ #13;  
    sString := sString + 'Seisoen: ' + fSeisoen + #13;  
    Result := sString;  
end;  
  
// =====  
// Kode wat voorsien is  
// =====  
  
function TsonkragAanleg.getAanlegKode: String;  
begin  
    result := fAanlegKode;  
end;  
  
function TsonkragAanleg.getGetalPanele: Integer;  
begin  
    result := fGetalPanele;  
end;  
  
function TsonkragAanleg.getSeisoen: String;  
begin  
    result := fSeisoen;  
end;  
  
end.
```

Hoofvormeenheid:

```
unit Vraag3_U;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, CheckLst, ExtCtrls, Buttons, Spin, ComCtrls,
  jpeg;

type
  TfrmVraag3 = class(TForm)
    gbxV3_2_1: TGroupBox;
    gbxV3_2_3: TGroupBox;
    btnV3_2_1: TButton;
    btnHerstel: TButton;
    gbxV3_2_2: TGroupBox;
    btnV3_2_2: TButton;
    edtV3_2_1_Krag: TEdit;
    Label2: TLabel;
    Label3: TLabel;
    sedV3_2_2: TSpinEdit;
    Panel1: TPanel;
    Panel2: TPanel;
    btnV3_2_3: TButton;
    Image1: TImage;
    Label6: TLabel;
    edtV3_2_1_Kode: TEdit;
    sedV3_2_1: TSpinEdit;
    Label4: TLabel;
    cmbV3_2_3: TComboBox;
    Label5: TLabel;
    gbxV3_2_4: TGroupBox;
    btnV3_2_4: TButton;
    redV3: TRichEdit;
    procedure btnV3_2_1Click(Sender: TObject);
    procedure btnHerstelClick(Sender: TObject);
    procedure btnV3_2_2Click(Sender: TObject);
    procedure btnV3_2_3Click(Sender: TObject);
    procedure btnV3_2_4Click(Sender: TObject);
  private
  public
  end;
var
  frmVraag3: TfrmVraag3;

implementation
{$R *.dfm}

uses
  SonkragAanleg_U;

var
  objAanleg: TSonkragAanleg;
```

```
// =====  
// Vraag 3.2.1                    6 punte  
// =====  
  
procedure TfrmVraag3.btnV3_2_1Click(Sender: TObject);  
begin  
    // Kode wat voorsien is  
    redV3.Clear;  
    // Vraag 3.2.1  
  
    objAanleg:= TsonkragAanleg.create(edtV3_2_1_Kode.Text,  
                                      sedV3_2_1.Value, strToFloat(edtV3_2_1_Krag.Text));  
    redV3.Lines.Add(objAanleg.toString);  
end;  
  
// =====  
// Vraag 3.2.2                    4 punte  
// =====  
  
procedure TfrmVraag3.btnV3_2_2Click(Sender: TObject);  
var  
    rOpgedateerdeKrag: Real;  
begin  
    // Kode wat voorsien is  
    redV3.Clear;  
    // Vraag 3.2.2  
  
    objAanleg.inkGetalPanele(sedV3_2_2.Value);  
    redV3.Lines.Add('Aanlegkode: ' + objAanleg.getAanlegKode);  
    redV3.Lines.Add('Getal panele: ' +  
                                      IntToStr(objAanleg.getGetalPanele));  
end;  
  
// =====  
// Vraag 3.2.3                    3 punte  
// =====  
  
procedure TfrmVraag3.btnV3_2_3Click(Sender: TObject);  
begin  
    // Kode wat voorsien is  
    redV3.Clear;  
    // Vraag 3.2.3  
  
    objAanleg.setSeisoen(cmbV3_2_3.Text);  
    redV3.Lines.Add(objAanleg.toString)  
end;
```

```
// =====  
// Vraag 3.2.4                    4 punte  
// =====  
  
procedure TfrmVraag3.btnV3_2_4Click(Sender: TObject);  
begin  
  // Kode wat voorsien is  
  redV3.Clear;  
  // Vraag 3.2.4  
  
  redV3.Lines.Add('Die maksimum opwekkingskapasiteit per dag in die ' +  
                  objAanleg.getSeisoen + ': ');  
  redV3.Lines.Add(floatToStr(objAanleg.berekenKapasiteit) + ' kW');  
  
end;  
  
// =====  
// Kode wat voorsien is  
// =====  
  
procedure TfrmVraag3.btnHerstelClick(Sender: TObject);  
begin  
  objAanleg.Free;  
  edtV3_2_1_Krag.Clear;  
  edtV3_2_1_Kode.Clear;  
  sedV3_2_1.Value := 15;  
  sedV3_2_2.Value := 50;  
  redV3.Clear;  
  
end;  
  
end.
```

BYLAAG H: OPLOSSING VIR VRAAG 4

```
unit Vraag4_u;

interface

uses
  Windows, Messages, SysUtils, Variants,
  Classes, Graphics, Controls, Forms, Dialogs, StdCtrls, ComCtrls,
  ExtCtrls, jpeg;

type
  TfrmVraag4 = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    btnV4_3: TButton;
    redV4: TRichEdit;
    btnV4_1: TButton;
    GroupBox1: TGroupBox;
    rgpV4: TRadioGroup;
    btnV4_2: TButton;
    Image1: TImage;
    lstV4: TListBox;
    GroupBox2: TGroupBox;
    GroupBox3: TGroupBox;
    procedure btnV4_3Click(Sender: TObject);
    procedure btnV4_1Click(Sender: TObject);
    procedure btnV4_2Click(Sender: TObject);
    procedure FormShow(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag4: TfrmVraag4;

  arrName: array [1 .. 10] of String = (
    'Ruth',
    'Nicole',
    'Loyiso',
    'Chris',
    'William',
    'Thabo',
    'Vusi',
    'Peter',
    'Jenny',
    'Tommy'
  );
```

```

arrOutomaat: array [1 .. 10, 1 .. 15] of String =
  (('K', '', '', '', '', '', '', '', '', '', '', '', '', ''),
   ('B', 'B', 'B', 'K', 'K', 'K', 'B', 'B', 'B', 'K', 'K', 'K', 'K',
'K', '' ),
   ('', '', '', '', '', '', '', '', '', '', '', '', '', ''),
   ('K', 'K', '', '', '', '', '', '', '', '', '', '', '', ''),
   ('B', 'B', 'K', 'K', 'B', 'B', 'K', 'K', 'K', 'K', 'B', 'K', 'K',
'B', '' ),
   ('K', 'K', 'B', '', '', '', '', '', '', '', '', '', '', '', ''),
   ('K', 'B', '', '', '', '', '', '', '', '', '', '', '', ''),
   ('B', 'B', '', '', '', '', '', '', '', '', '', '', '', ''),
   ('K', '', '', '', '', '', '', '', '', '', '', '', '', '', ''),
   ('B', 'K', '', '', '', '', '', '', '', '', '', '', '', '' ));
implementation

```

```
{$R *.dfm}
```

```
// =====
// Vraag 4.1 5 punte
// =====
```

```

procedure TfrmVraag4.btnV4_1Click(Sender: TObject);
var
  I, J: Integer;
  sReel: String;
begin
  // Kode wat voorsien is
  redV4.Clear;
  redV4.Lines.Add('-----');
  redV4.Lines.Add('Name'+#9+'Herwinde items');
  redV4.Lines.Add('-----');

  // Vraag 4.1

  for I := 1 to length(arrName) do
  begin
    sReel := arrName[I] + #9;
    for J := 1 to length(arrOutomaat[J]) do
    begin
      sReel := sReel + arrOutomaat[I, J];
    end;
    redV4.Lines.Add(sReel);
  end;

end;

```

```
// =====
// Vraag 4.2 14 punte
// =====
```

```

procedure TfrmVraag4.btnV4_2Click(Sender: TObject);
var
  I, J: Integer;
  rBedrag, rMaks: Real;
  arrTotaal: Array [1 .. 10] of Real;
  sMaks: String;

```

```

begin
  // Kode wat voorsien is
  redV4.Clear;
  redV4.Lines.Add('-----');
  redV4.Lines.Add('Name'+      Totale bedrag uitbetaal');
  redV4.Lines.Add('-----');

  // Vraag 4.2
  // Alternatief
  {rMaks := -1;
  for I := 1 to length(arrName) do
  begin
    rBedrag := 0;
    for J := 1 to length(arrOutomaat[I]) do
      if arrOutomaat[I, J] = 'B' then
        rBedrag := rBedrag + 2.15
      else
        if arrOutomaat[I, J] = 'K' then
          rBedrag := rBedrag + 0.75;

    if rBedrag > rMaks then
      rMaks := rBedrag;

    arrTotaal[I] := rBedrag;
    redV4.Lines.Add(arrName[I] + #9 + format('%8.2m', [rBedrag]));
  end; //I

  // Kode wat voorsien is
  redV4.Lines.Add('-----');
  redV4.Lines.Add('Grootste uitbetaling(s):');
  redV4.Lines.Add('-----');

  //Kodeer hier

  for I := 1 to length(arrTotaal) do
  begin
    redV4.SelAttributes.Style:= [fsBold];
    redV4.SelAttributes.Color:= clRed;
    if arrTotaal[I] = rMaks then
    begin
      redV4.Lines.Add(arrName[I] + #9 + format('%8.2m', [rMaks]));
    end;
  end;}

  rMaks := -1;
  for I := 1 to length(arrName) do
  begin
    rBedrag := 0;
    for J := 1 to length(arrOutomaat[I]) do
      if arrOutomaat[I, J] = 'B' then
        rBedrag := rBedrag + 2.15
      else if arrOutomaat[I, J] = 'K' then
        rBedrag := rBedrag + 0.75;
  
```

```
redV4.Lines.Add(arrName[I] + #9 + format('%8.2m', [rBedrag]));

if rBedrag >= rMaks then
begin
  if rBedrag > rMaks then
  begin
    rMaks := rBedrag;
    sMaks := '';
  end;
  sMaks := sMaks + arrName[I] + #9 + format('%8.2m',
    [rBedrag]) + #13;
end;
end; //I

// Kode wat voorsien is
redV4.Lines.Add('-----');
redV4.Lines.Add('Grootste uitbetaling(s):');
redV4.Lines.Add('-----');

redV4.Lines.Add(sMaks);
end;

// Kode wat voorsien is
procedure TfrmVraag4.FormShow(Sender: TObject);
begin
  redV4.Paragraph.TabCount := 1;
  redV4.Paragraph.Tab[0] := 70;

end;

// =====
// Vraag 4.3 11 punte
// =====

procedure TfrmVraag4.btnV4_3Click(Sender: TObject);
var
  iRy, iKolom: Integer;
  sItem: String;
  bBygevoeg: Boolean;
begin
  // Vraag 4.3
  {bBygevoeg := False;
  if lstV4.ItemIndex = -1 then
  begin
    ShowMessage('Selekteer asb 'n naam');
    exit;
  end
  else
  begin
    case rgpV4.ItemIndex of
      -1:
        showMessage('Selekteer asb 'n item');
    else
      sItem := rgpV4.Items[rgpV4.ItemIndex][1];
    end;
  end;
```

```
iRy := lstV4.ItemIndex+1;
iKolom:=0;
while (bBygevoeg=false) AND ( iKolom<15) do
begin
  Inc(iKolom);
  if arrOutomaat[iRy,iKolom] ='' then
  begin
    arrOutomaat[iRy,iKolom] := sItem;
    bBygevoeg := True;
  end;
end;

if bBygevoeg = false then
begin
  ShowMessage('Outomaat is vol.');
```

```
end; }

bBygevoeg := False;
if (lstV4.ItemIndex > -1) AND (rgpV4.ItemIndex > -1) then
begin
  sItem := rgpV4.Items[rgpV4.ItemIndex][1];
  iRy := lstV4.ItemIndex+1;
  iKolom:=0;
  while (bBygevoeg=false) AND ( iKolom<15) do
  begin
    Inc(iKolom);
    if arrOutomaat[iRy,iKolom] = '' then
    begin
      arrOutomaat[iRy,iKolom] := sItem;
      bBygevoeg := True;
    end;
  end;
  btnV4_1.Click;
end;

if bBygevoeg = false then
begin
  ShowMessage('Outomaat is vol.');
```

```
end;
else
  ShowMessage('Selekteer asseblief beide ' 'n naam en ' 'n item.');
```

```
// Kode wat voorsien is
rgpV4.ItemIndex := -1;
lstV4.ItemIndex := -1;

end;

end.
```