



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.

Goedgekeur:

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.

BYLAAG A**VRAAG 1: NASIENRUBRIEK - ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

| SENTRUMNOMMER: | | EKSAMENNOMMER: | |
|----------------|---|----------------|--------------|
| VRAAG | BESKRYWING | MAKS. PUNTE | LEERDER-PUNT |
| 1.1 | <p>Knoppie [1.1 – Eienskappe]</p> <p>Stel die panel se kleur op 'clyellow' ✓ Stel die skrif se styl op 'italic' ✓ Verander die teks na "Ek hou van programmering!" ✓</p> | 3 | |
| 1.2 | <p>Knoppie [1.2 – Verlof-maand]</p> <p>Kry maand naam / maand indeks van cmbV1_2 ✓ Toets IF (sMaand = 'Junie') ✓ OR (sMaand = 'Julie') OR (sMaand = 'Augustus') ✓ then Vertoon boodskap 'Maatskappy gesluit, selekteer 'n ander maand.' ✓ Deselekteer maand gekies in cmbV1_2 skoon (itemindex -1) ✓ Anders ✓ Vertoon boodskap 'Jou verlof in ' + sMaand + 'is toegestaan.' ✓</p> <p>ALTERNATIEWE IF-voorwaardes</p> <ul style="list-style-type: none"> • IF iMaand IN [5,6,7] then • IF (iMaand >= 5) AND (iMaand <= 7) then <p>NOTA: As die kandidaat die indeks aanpas volgens maandnommers, moet die reeks van 6 tot 8 wees</p> | 7 | |

| | | | |
|-----|--|----|--|
| 1.3 | <p>Knoppie [1.3 – Bereken]</p> <p>Verkry A en B uit edit-blokkies ✓ 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: 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</p> | 11 | |

| | | | |
|-----|--|-----------|--|
| 1.5 | <p>Knoppie - [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 woord2 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 ShowMessage 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 om //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 | |

BYLAAG B**VRAAG 2: NASIENRUBRIEK - DATABASISPROGRAMMERING**

| SENTRUMNOMMER: | | EKSAMENNOMMER: | |
|----------------|--|----------------|--------------|
| VRAAG | BESKRYWING | MAKS. PUNTE | LEERDER-PUNT |
| 2.1 | SQL-stellings | | |
| 2.1.1 | Knoppie [2.1.1 – Bloemfontein-kliënte] SELECT * ✓ FROM tblKliente ✓ WHERE Stad = "Bloemfontein" ✓ ORDER BY KlientVan ✓ | 4 | |
| 2.1.2 | Knoppie [2.1.2 – Eerste maand] SELECT InsamelID, InsamelDatum, GetalBlikkies FROM tblBlikInsameling ✓ WHERE Month ✓ (InsamelDatum) = 1 ✓ Alternatief: Onttrek maand deur String-bewerkings | 3 | |
| 2.1.3 | Knoppie [2.1.3 – Soek] SELECT * FROM tblKliente ✓ WHERE KlientID LIKE ✓ " " + sLetter ✓ + "%" ✓ Alternatief: WHERE left(KlientID, 1) = quotedStr(sLetter) | 4 | |
| 2.1.4 | Knoppie [2.1.4 – Kliënte se verdienstes] SELECT KlientNaam, FORMAT(SUM ✓ (GetalBlikkies)/ 76 * 8) ✓, "Currency") ✓ AS ✓ [Totale Bedrag] ✓ FROM tblBlikInsameling A, tblKliente B ✓ WHERE A.KlientID = B.KlientID ✓ GROUP BY KlientNaam ✓ | 8 | |
| 2.1.5 | Knoppie [2.1.5 – Dateer op] UPDATE tblBlikInsameling ✓ SET GetalBlikkies = 250 ✓, Betaal = False ✓ WHERE InsamelID = "C003" ✓ | 4 | |
| | Subtotaal: | 23 | |

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> <p>tblKliente.Insert; tblKliente['KlientID'] := 'CHA01'; tblKliente['KlientNaam'] := 'Charles'; tblKliente['KlientVan'] := 'du Boit'; tblKliente['Adres'] := 'Van Wouwstraat 24'; tblKliente['Stad'] := 'Kaapstad'; tblKliente.Post;</p> | 4 | |
| 2.2.2 | <p>Knoppie [2.2.2 – Persentasie]</p> <p>Vertoon die naam en van, van die geselekteerde kliënt ✓ Inisialiseer 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 geformateer 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 "Sommer" ✓ | 5 | |
| 3.1.2 | procedure inkGetalPanele 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 = 'Sommer' ✓ 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 ✓ omgeskakel na regte datatipes ✓</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 ✓</p> <p>Vertoon afvoerstring in rich edit ✓</p> | 5 | |
| 4.2 | <p>Knoppie [4.2 – Bereken]</p> <p>Inisialiseer Maks-veranderlike na 0 ✓</p> <p>Buitenste lus I van 1 tot lengte van arrName (10) ✓</p> <p>Inisialiseer bedragBetaal na 0 ✓</p> <p>Binneste lus J van 1 tot lengte van arrOutomaat[J] (15) ✓</p> <p>Toets of herwin-item 'n bottel is ✓</p> <p>Inkrementeer bedragBetaal met 2.15 ✓</p> <p>Toets of herwin-item 'n blikkie is</p> <p>Inkrementeer bedragBetaal met 0.75 ✓</p> <p><i>Eindig binneste lus</i></p> <p>Vertoon naam en bedragBetaal, omgeskakel na geldeenheid ✓</p> <p>Alternatief 1:</p> <p>Toets of BedragBetaal >= Maks ✓</p> <p>As BedragBetaal > Maks ✓</p> <p>Stel Maks na BedragBetaal ✓ en maak afvoer skoon ✓</p> <p>Voeg naam en bedrag by Maks-afvoerstring ✓</p> <p><i>Eindig buitenste lus</i></p> <p>Vertoon Maks afvoer string in redV4 ✓</p> <p>Alternatief 2 (gebruik parallelle skikking)</p> <p>Toets of BetaalBedrag > Maks //1</p> <p>Stel Maks na BedragBetaal //1</p> <p>Stoor BedragBetaal in parallelle skikking //1</p> <p><i>Eindig buitenste lus</i></p> <p>Lus van 1 tot 10 //1</p> <p>As BetaalBedrag = Maks //1</p> <p>Vertoon naam en BetaalBedrag //1</p> | 14 | |

| | | | |
|---------------------------|--|------------|--|
| 4.3 | Knoppie [4.2 – Voeg item by] Toets of naam OF item NIE geselekteer is nie ✓ Vertoon boodskap as naam of item nie geselekteer is nie ✓ Kry item van komponent ✓ Kry OF bepaal ry(indeks) van naam ✓ <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 ✓ Vertoon opgedateerde skikking ✓ <u>GEEN SPASIE BOODSKAP:</u> Toets vlag OF gebruik 'n ander meganisme om te bepaal of spasio beskikbaar is ✓ Vertoon boodskap as geen spasio beskikbaar is nie ✓ | 11 | |
| TOTAAL AFDELING D: | | 30 | |
| GROOTTOTAAL: | | 150 | |

OPSOMMING VAN LEERDER SE PUNTE:

| SENTRUMNOMMER: | | LEERDER SE EKSAMENNUMMER: | | | |
|-------------------------|------------|---------------------------|------------|------------|--------------|
| | 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.');
```

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  
    iUre : Integer;  
begin  
  
    if fSeisoen = 'Somer' then  
    begin  
        iUre := 10;  
    end  
    else  
    if fSeisoen = 'Winter' then  
    begin  
        iUre := 6;  
    end  
    else  
    begin  
        iUre := 8;  
    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  
    rOppedateerdeKrag: 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;
  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.
```