



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**SENIOR SERTIFIKAAT/
NASIONALE SENIOR SERTIFIKAAT**

GRAAD 12

INLIGTINGSTEGNOLOGIE V1

NOVEMBER 2020

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 3-12) sluit die nasienrubriek vir elke om te gebruik vir enigeen van die twee programmeringstale in.
- **Bylaag E, F, G en H** (bladsy 13-29) bevat voorbeelde in programmeringskode van oplossings vir **VRAAG 1** tot **VRAAG 4**.
- Kopieë van **Bylaag A, B, C, D en die opsomming van die leerder se punte** (bladsy 3-12) 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	Knoppie [1.1 – Kleure] Verander die skrifgrootte van cmbV1_1 na 12pt✓ cmbV1_1.items.add✓ Voeg blou by✓ Stel item se indeks op 0 /stel text to 'Groen' ✓	4	
1.2	Knoppie [1.2 – Vlieër] Skep 'n veranderlike vir oppervlakte (rOppvlak)✓ Toets of $iDiagA > iDiagB$ ✓ $rOppvlak = iDiagA * iDiagB / 2$ ✓ Vertoon rOppvlak op lbIV1_2✓ geformatteer tot een desimaal✓ Anders✓ Vertoon foutboodskap✓	8	
1.3	Knoppie [1.3 – Binêre getal] Verkry desimale getal uit 'edit box'✓, skakel om na heelgetal✓ Inisialiseer afvoerstring vir binêre waarde✓ Lus (Loop) ✓ terwyl desimale getal > 0 ✓ Res = desimale getal MOD 2 ✓ (resultaat 0 of 1) Desimale getal ✓ = desimale getal DIV 2 ✓ Voeg res wat omgeskakel is na 'n string✓ by aan die begin van die afvoerstring ✓ Vertoon binêre getal op 'label' lbIV1_3 ✓ Alternatiewe: Lus terwyl desimale getal $div\ 2 > 0$ Herhaal totdat desimale getal = 0	11	

1.4	Knoppie [1.4 – Woordspeletjie] Inisialiseer totaal na 0 ✓ Verkry woord uit 'edit box' ✓ en verander na hoofletter ✓ Toets of dit nie enkele woord is nie (Spasie is gevind) ✓ Maak 'edit box' skoon ✓ Stel fokus op die 'edit box' ✓ Vertoon foutboodskap ✓ Anders ✓ Lus (Loop) deur woord deur lengte van woord te gebruik ✓ Verkry karakter by lus se veranderlike ✓ Toets of dit 'n klinker is ✓ Tel 3 by totaal ✓ Anders Toets vir ander alfabetiese karakter ✓ Tel 2 by totaal ✓ Anders Tel 1 by totaal ✓ Vertoon totale waarde van woord in 'richedit' ✓	17	
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 – Lys van werknemers] <pre>SELECT * FROM tblWerknemers ORDER BY PosTitel, UurlikseLoon DESC</pre> Concepts: SELECT * ✓ FROM tblWerknemers ✓ ORDER BY PosTitel ✓ ,UurlikseLoon DESC ✓	4	
2.1.2	Knoppie [2.1.2 – Ingenieurs] <pre>SELECT WerknemerID, Van, Naam FROM tblWerknemers WHERE PosTitel LIKE "%Engineer%"</pre> Concepts: SELECT WerknemerID, Van, Naam ✓ FROM tblWerknemers ✓ WHERE PosTitel LIKE ✓ "%Ingenieur%" ✓	4	
2.1.3	Knoppie [2.1.3 – Postitel] <pre>SELECT DISTINCT PosTitel FROM tblWerknemers OF SELECT PosTitel FROM tblWerknemers GROUP BY PosTitel</pre>	3	

	Concepts: SELECT DISTINCT ✓ PosTitel ✓ FROM tblWerknemers ✓ OF SELECT PosTitel (1) FROM tblWerknemers (1) GROUP BY PosTitel (1)		
2.1.4	Knoppie [2.1.4 – Verwyder rekords]	2	
	DELETE FROM tblLogUre WHERE UreGewerk = 99		
	Konsepte: DELETE FROM tblLogUre ✓ WHERE UreGewerk = 99 ✓		

VRAAG 2: NASIENRUBRIEK – VERVOLG

<p>2.1.5</p>	<p>Knoppie [2.1.5 – Oortyd]</p> <hr/> <p>SELECT Van, FORMAT(SUM((UreGewerk - 8) * UurlikseLoon * 2), "CURRENCY") AS OortydBedrag FROM tblWerknemers E, tblLogUre H WHERE E.WerknemerID = H.WerknemerID AND ureGewerk > 8 GROUP BY Van</p> <hr/> <p>SELECT Van, ✓ FORMAT(SUM ✓ ((UreGewerk - 8) ✓ * UurlikseLoon * 2 ✓), "CURRENCY") ✓ AS OortydBedrag ✓ FROM tblWerknemers E, tblLogUre H ✓ WHERE E.WerknemerID = H.WerknemerID ✓ AND ureGewerk > 8 GROUP BY Van ✓</p> <p>Konsepte: Verkry Van (1) Formateer oortyd betaal as geldeenheid (1) Gebruik SUM-funksie reg (1) Bereken aantal ure oortyd gewerk (1) Vermenigvuldig resultaat van bostaande stap met UurlikseLoon * 2 (1) Noem berekende veld OortydBedrag (1) Verkry data uit tblWerknemers en tblLogUre (1) Toets of daar 'n verwantskap is tussen PKen FK (1) Groepeer resultate volgens Van (1)</p>	<p>9</p>	
	<p style="text-align: right;">Subtotaal:</p>	<p>22</p>	

2.2	DATABASISMANIPULASIE deur Delphi-kode te gebruik		
2.2.1	Knoppie [2.2.1 – Werknemers met noodhulp] Vertoon kolomopskrifte // Is voorsien Inisialiseer teller as 0 Set tblWerknemers om van eerste rekord af te begin lees ✓ Lus (Loop) terwyl nie tblWerknemers.Eof ✓ Toets of tblWerknemers['NoodHulpOpleiding'] = True ✓ Vertoon WerknemerID, Van en PosBeskrywing ✓ Inkrementeer teller met 1 ✓ Gaan na volgende rekord in tblWerknemers ✓ Vertoon teller ✓ omgeskakel na string	7	
2.2.2	Knoppie [2.2.2 – Voeg nuwe werknemer by] tblWerknemers.Insert ✓ tblWerknemers['WerknemerID'] := 'EMP986'; ✓ tblWerknemers['Naam'] := 'Robert' tblWerknemers['Van'] := 'Laubscher' ✓ tblWerknemers['UurlikseLoon'] := 195.00 ✓ tblWerknemers['PosTitel'] := 'Mariene Ingenieur' tblWerknemers['NoodHulpOpleiding'] := True ✓ tblWerknemers.Post ✓ Alternatiewe tblWerknemers[UurlikseLoon] := 'R195.00'; tblWerknemers [UurlikseLoon] := FloatToStrF(195.00, ffCurrency, 6, 2); Konsepte: Insert/Append (1) Stringveranderlikes – ID (1) Naam en Van (1) PosTitel en UurlikseLoon (1) NoodHulpOpleiding (1) Post (1)	6	
2.2.3	Knoppie [2.2.3 – Dateer ure gewerk op] Kry aantal ure (iUre of sUre) uit edtV2_2_3✓, Skakel om na heelgetal tblLogUre.Edit ✓ tblLogUre['UreGewerk'] ✓ = iUre ✓ tblLogUre.Post ✓	5	
	Subtotaal:	18	
	TOTAAL AFDELING B:	40	

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

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
3.1.1	Konstruktor-metode: Hoof met regte parameterwaardes (korrekte volgorde en datatipes) ✓ Ken klant se ID parameterwaarde toe aan fKlantID ✓ Ken houergrootte se parameterwaarde toe aan fHouerGrootte ✓ Ken stoortyperk se parameterwaarde toe aan fStoorTydperk Stel fBedragBetaal op 0 ✓	4	
3.1.2	getBedragBetaal-metode: Funksie se opskrif met real/double as terugstuur-datatype ✓ fBedragBetaal toegeken aan result ✓	2	
3.1.3	OpdateerBedragBetaal-metode: Prosedure-opskrif met real/double parameterwaarde ✓ Inkrementeer fBedragBetaal ✓ Gebruik die parameterwaarde ✓	3	

3.1.4	<p>berekenKoste-metode:</p> <p>Funksie verklaar met double/real terugstuur-datatipe As fHouerGrootte = 'S' Stel aanvanklike koste op 1000.00 Anders as fHouerGrootte = 'M' Stel aanvanklike koste op 1750.00 Anders Stel aanvanklike koste op 2500.00 Koste = StoorTydperk * aanvanklike koste Persentasie afslag = 10% vir elke vermeerdering van 6 maande deur die floor/trunc/div-funksie te gebruik of ander toepaslike kode Toets of afslag > 50 Stel afslag op 50 Bereken afslagwaarde Stuur koste as 'result' terug</p> <p>Aangepaste merkbenadering</p> <p><i>Ken punte toe vir die volgende konsepte:</i></p> <p>Funksie verklaring✓ met terugstuur datatipe✓ (heelgetal/string) Toets vir die houergrootte ✓✓ Toekenning van korrekte koste✓ per grootte✓ Bereken koste✓ met formule Bepaal persentasie afslag deur maande te gebruik✓ Stel afslag persentasie na 50✓ Bereken afslag waarde✓ Stuur koste terug in result✓</p>	11	
3.1.5	<p>toString-metode:</p> <p>'Labels' (Klant se ID, Houergrootte, Stoortydperk, Bedrag betaal) ✓ Regte attribute✓ Regte omskakelings (bedrag betaal – float; stoortydperk – integer) ✓</p>	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>Het KlantID uit 'edit box' as argument gebruik ✓ Het eerste karakter ✓ van geselekteerde item in 'list box' as argument gebruik ✓ Het aantal maande uit 'spin edit' as argument gebruik ✓</p> <p><i>Instansieer die objTransaksie-objek:</i> objTransaksie:= ✓ TTransaksie.Create✓ (correct order, data types)✓</p> <p>Stel btnV3_2_1 onaktief ✓</p>	7	
3.2.2 (a)	<p>Knoppie [3.2.2 (a) – Vertoon bedrag verskuldig]</p> <p>Roep die berekenKoste-metode as 'n funksie✓ Vertoon die bedrag verskuldig✓ op pnIV3_2_2 ✓</p>	3	
3.2.2 (b)	<p>Knoppie [3.2.2 (b) – Verwerk betaling]</p> <p>verkry waarde uit edtV3_2_2 ✓ Roep opdateerBedragBetaal ✓ met onttrekte waarde as argument</p> <p>BedragVerskuldig := berekenKoste ✓ - getBedragBetaal ✓ Konsepte: Korrekte roepstelling van metode Korrekte berekening (-)</p> <p>Vertoon die opgedateerde bedrag verskuldig op pnIV3_2_2 ✓</p>	5	
3.2.3	<p>Knoppie [3.2.3 – Sien besonderhede]</p> <p>Roep die toString-metode ✓ Vertoon in die richedit ✓</p>	2	
	Subtotaal: Vormklas	17	
	TOTAAL AFDELING C:	40	

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

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER - PUNT
4.1.1	<p>Knoppie [4.1 – Skep hawehouers] Lus (Loop) 50 keer ✓ Genereer reële getalle ✓ tussen 1 en 99 (inklusief) ✓ Rond af tot 1 desimaal Stoor die waardes in arrHouers-skikking ✓</p> <p>Alternatiewe</p>	4	
4.1.2	<p>Knoppie [4.1.2 – Vertoon hawehouers] Stel indeks op 0 ✓ Lus deur 5 keer (vir rye) ✓ sReel := ' ' ✓ Lus 10 keer (vir kolomme) ✓ Inkrementeer die indeks met 1 ✓ sReel := sReel + FloatToStr(arrHouers[iIndeks]) ✓ + #9; ✓ Vertoon sReel in die rich edit redV4_1_2 ✓</p> <p>Alternatief: sReel := sReel + FloatToStr(arrHouers[c+(10*(r-1))] (3) + #9(1)</p> <p>Konsepte: Gebruik lus (loop) veranderlikes (1) Berekening (1) Floattostr (1) Space(1)</p>	8	

4.2	<p>Knoppie [4.2 – Houers gelaai om verskeep te word] Stel TotaleGewig op nul (0) ✓ Inisialiseer die vertoonveranderlike (sReel) ✓</p> <p>Stel indeks op 1 ✓ Lus index < 50 ✓ As (TotaleGewig + gewig van die houer by die indeks) ✓ <= 200 ✓ dan Voeg gewig van houer by indeks ✓ van arrHouers skikking by TotaleGewig ✓ Voeg die gewig van die houer by indeks ✓ van arrHouers- en voeg #9 ✓ by sReel ✓ Inkrementeer die indeks ✓</p> <p>Buite lus: Vertoon die gewig van die houers wat gelaai is om verskeep te word in die richedit ✓ Vertoon die totale gewig van die houers wat gelaai is op die paneel ✓</p> <p>Ken lêerveranderlike aan die naam van die tekslêer toe ✓ Rewrite ✓ Skryf die gewig van die houers wat gelaai is om verskeep te word na die tekslêer Tonne.txt ✓ Maak tekslêer toe ✓</p> <p>Alternatief: For loop 1 tot 50 (3 punte)</p>	18	
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

```

unit Vraag1_U;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, ExtCtrls, Spin, pngimage, ComCtrls, Math,
  Buttons;

type
  TfrmVraag1 = class(TForm)
    GroupBoxV1_1: TGroupBox;
    GroupBoxV1_4: TGroupBox;
    edtV1_4: TEdit;
    Label9: TLabel;
    redV1_4: TRichEdit;
    GroupBoxQ1_3: TGroupBox;
    edtV1_3: TEdit;
    btnV1_3: TButton;
    Label10: TLabel;
    lblV1_3: TLabel;
    btnV1_1: TButton;
    cmbV1_1: TComboBox;
    btnClose: TBitBtn;
    GroupBoxV1_2: TGroupBox;
    Image2: TImage;
    lblV1_2: TLabel;
    btnV1_2: TButton;
    btnV1_4: TButton;
    procedure btnV1_2Click(Sender: TObject);
    procedure btnV1_4Click(Sender: TObject);
    procedure btnV1_3Click(Sender: TObject);
    procedure btnV1_1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  frmVraag1: TfrmVraag1;

implementation

{$R *.dfm}

//=====
//
//                      Vraag 1.1 - 4 punte
//=====
procedure TfrmVraag1.btnV1_1Click(Sender: TObject);
begin
  // Vraag 1.1
  cmbV1_1.Font.Size := 12;
  cmbV1_1.Items.Add('Blou');
  cmbV1_1.ItemIndex := 0;
end;

```

```
//=====
//
//                               Vraag 1.2 - 8 punte
//=====
procedure TfrmVraag1.btnV1_2Click(Sender: TObject);
var
  iDiagA, iDiagB: integer;
  rOppvlak: real;
begin
  // Kode wat voorsien is

  iDiagA := StrToInt(TextBox('Diagonaal een van vlieër (cm)', 'Sleutel
die waarde van diagonaal een (A) in: ', '20'));
  iDiagB := StrToInt(TextBox('Diagonaal twee van vlieër (cm)',
  'Sleutel die waarde van diagonaal twee (B) in:', ''));

  // Vraag 1.2
  if iDiagA > iDiagB then
  begin
    rOppvlak := (iDiagA * iDiagB) / 2;
    lblV1_2.Caption := 'Die oppervlakte van die vlieër is ' +
      FloatToStrF(rOppvlak, ffFixed, 10, 1) + ' vierkante cm.';
  end
  else
  begin
    Showmessage('Die waarde van diagonaal twee (B) moet kleiner wees as
die waarde van diagonaal een (A)');
  end;
end;

//=====
//
//                               Vraag 1.3 - 11 punte
//=====

procedure TfrmVraag1.btnV1_3Click(Sender: TObject);
// Kode wat voorsien is
var
  iGetal, iRes: integer;
  sBiner: String;
begin
  // Vraag 1.3
  iGetal := StrToInt(edtV1_3.Text);
  sBiner := '';
  while iGetal <> 0 do
  begin
    iRes := iGetal MOD 2;
    iGetal := iGetal DIV 2;
    sBiner := IntToStr(iRes) + sBiner;
  end;
  lblV1_3.Caption := 'Binêre getal: ' + sBiner;
end;
```

```
//=====
//                               Vraag 1.4 - 17 punte
//=====
procedure TfrmVraag1.btnV1_4Click(Sender: TObject);
var
  sWoord: String;
  iTotaal, iWaarde, iRand, iLen, iTel: integer;
  cKarakter: char;
begin
  // Kode wat voorsien is
  redV1_4.Clear;
  // Vraag 1.4
  iTotaal := 0;
  sWoord := Uppercase(edtV1_4.Text);

  if pos(' ', sWoord) = 0 then
  begin
    iLen := Length(sWoord);

    for iTel := 1 to iLen do
    begin
      cKarakter := sWoord[iTel];

      if cKarakter in ['A', 'E', 'I', 'O', 'U'] then
        iTotaal := iTotaal + 3
      else
        if cKarakter in ['A'..'Z'] then
          iTotaal := iTotaal + 2
        else
          iTotaal := iTotaal + 1;
    end;

    redV1_4.lines.Add('Totale getal punte: ' + IntToStr(iTotaal));
  end
  else
  begin
    edtV1_4.Clear;
    edtV1_4.SetFocus;
    ShowMessage('Gediskwalifiseer - meer as een woord is ingesleutel.');
```

BYLAAG F: OPLOSSING VIR VRAAG 2

```

unit Vraag2_U;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, Buttons, ExtCtrls, ConnectDB_U, DB, ADODB,
  Grids, DBGrids, ComCtrls, DateUtils, DBCtrls, Spin;

type
  TfrmVraag2 = class(TForm)
    pnlBtns: TPanel;
    bmbHerstelDB: TBitBtn;
    grpTblUurAangeteken: TGroupBox;
    grpTblWerknemers: TGroupBox;
    dbgWerknemers: TDBGrid;
    dbgLogUre: TDBGrid;
    tabsV2_2ADO: TTabSheet;
    tabsV2_1SQL: TTabSheet;
    btnV2_2_1: TButton;
    redV2: TRichEdit;
    grpResultate: TGroupBox;
    dbgSQL: TDBGrid;
    grpOutput: TGroupBox;
    pgcTabs: TPageControl;
    btnV2_1_1: TButton;
    btnV2_1_2: TButton;
    btnV2_1_3: TButton;
    btnV2_1_4: TButton;
    btnV2_1_5: TButton;
    btnV2_2_2: TButton;
    btnClose: TBitBtn;
    GroupBox1: TGroupBox;
    edtV2_2_3: TLabelledEdit;
    btnV2_2_3: TButton;
    procedure bmbHerstelDBClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure btnV2_1_1Click(Sender: TObject);
    procedure btnV2_1_2Click(Sender: TObject);
    procedure btnV2_1_3Click(Sender: TObject);
    procedure btnV2_1_4Click(Sender: TObject);
    procedure btnV2_1_5Click(Sender: TObject);
    procedure btnV2_2_1Click(Sender: TObject);
    procedure btnV2_2_3Click(Sender: TObject);
    procedure btnV2_2_2Click(Sender: TObject);

  private
  public
  end;

var
  frmVraag2: TfrmVraag2;
  dbCONN: TConnection;

```



```
// --- Globale veranderlikes wat voorsien is ---
tblWerknemers, tblLogUre: TADOTable;
qryDB: TADOQuery;
implementation
{$R *.dfm}
{$R+}
```

Vraag 2.1 - SQL-afdeling

```
// =====
//                               Vraag 2.1.1 - 4 punte
// =====
procedure TfrmVraag2.btnV2_1_1Click(Sender: TObject);
var
    sSQL1: String;
begin
    sSQL1 := 'SELECT * FROM tblWerknemers ' +
            'ORDER BY PosTitel, UurlikseLoon DESC';

    // Kode wat voorsien is - moenie verander nie
    dbCONN.RunSQL(sSQL1, dbgSQL, 1);
end;

// =====
//                               Vraag 2.1.2 - 4 punte
// =====
procedure TfrmVraag2.btnV2_1_2Click(Sender: TObject);
var
    sSQL2: String;
begin
    sSQL2 := 'SELECT WerknemerID, Van, Naam FROM tblWerknemers ' +
            'WHERE PosTitel LIKE "%Ingenieur%";

    // Kode wat voorsien is - moenie verander nie
    dbCONN.RunSQL(sSQL2, dbgSQL, 2);
end;

// =====
//                               Vraag 2.1.3 - 3 punte
// =====
procedure TfrmVraag2.btnV2_1_3Click(Sender: TObject);
var
    sSQL3: String;
begin
    sSQL3 := 'SELECT DISTINCT PosTitel FROM tblWerknemers';

    // Kode wat voorsien is - moenie verander nie
    dbCONN.RunSQL(sSQL3, dbgSQL, 3);
end;
```

```
// =====  
// Vraag 2.1.4 - 2 punte  
// =====  
procedure TfrmVraag2.btnV2_1_4Click(Sender: TObject);  
  var  
    sSQL4: String;  
    bChange: boolean;  
begin  
  sSQL4 := 'DELETE FROM tblLogUre ' +  
    'WHERE UreGewerk = 99';  
  
  // Kode wat voorsien is - moenie verander nie  
  dbCONN.ExecuteSQL(sSQL4, bChange);  
  if bChange then  
    begin  
      MessageDlg('Databasis is opgedateer.', mtInformation, [mbOK], 0);  
    end;  
end;  
  
// =====  
// Vraag 2.1.5 - 9 punte  
// =====  
procedure TfrmVraag2.btnV2_1_5Click(Sender: TObject);  
  var  
    sSQL5: String;  
begin  
  sSQL5 := 'SELECT Van, FORMAT(SUM((UreGewerk - 8) * (UurlikseLoon *  
2)), "CURRENCY") ' +  
    'AS OortydBedrag ' +  
    'FROM tblWerknemers E, tblLogUre H ' +  
    'WHERE E.WerknemerID = H.WerknemerID AND UreGewerk > 8' +  
    'GROUP BY Van';  
  
  // Kode wat voorsien is - moenie verander nie  
  dbCONN.RunSQL(sSQL5, dbgSQL, 5);  
end;
```

Vraag 2.2 - Delphi-afdeling

```
// =====  
// Vraag 2.2.1 - 7 punte  
// =====  
procedure TfrmVraag2.btnV2_2_1Click(Sender: TObject);  
    var  
        itel: integer;  
begin  
    // Kode wat voorsien is  
    redV2.Clear;  
    redV2.Paragraph.TabCount := 3;  
    redV2.Paragraph.Tab[0] := 100;  
    redV2.Paragraph.Tab[1] := 190;  
    redV2.SelAttributes.Style := [fsBold, fsUnderline];  
    redV2.Lines.Add('WerknemerID' + #9 + 'Van' + #9 + ' PostTitel  
' );  
  
    // Sleutel jou kode vir Vraag 2.2.1 hier in  
    iTel := 0;  
    tblWerknemers.First;  
    while tblWerknemers.Eof = False do  
        begin  
            if tblWerknemers['NoodHulpOpleiding'] = True then  
                begin  
                    redV2.Lines.Add(tblWerknemers['WerknemerID'] + #9 +  
                        tblWerknemers['Van'] + #9 +  
                        tblWerknemers[' PostTitel ']);  
  
                    Inc(itel);  
                end;  
            tblWerknemers.Next;  
        end;  
    redV2.Lines.Add(#10 + 'Totale getal werknemers met noodhulpopleiding:  
' + IntToStr(itel));  
end;  
  
// =====  
// Vraag 2.2.2 - 6 punte  
// =====  
procedure TfrmVraag2.btnV2_2_2Click(Sender: TObject);  
begin  
    // Vraag 2.2.2  
    tblWerknemers.Insert();  
    tblWerknemers['WerknemerID'] := 'EMP986';  
    tblWerknemers['Naam'] := 'Robert';  
    tblWerknemers['Van'] := 'Laubscher';  
    tblWerknemers['UurlikseLoon'] := 195.00;  
    tblWerknemers[' PostTitel ' ] := 'Mariene Ingenieur';  
    tblWerknemers['NoodHulpOpleiding'] := True;  
    tblWerknemers.Post();  
end;
```

```
// =====  
//                               Vraag 2.2.3 - 5 punte  
// =====  
procedure TfrmVraag2.btnV2_2_3Click(Sender: TObject);  
    var  
        iUre: Integer;  
begin  
    // Vraag 2.2.3  
    iUre := StrToInt(edtV2_2_3.Text);  
    tblLogUre.Edit();  
    tblLogUre['UreGewerk'] := iUre;  
    tblLogUre.Post();  
end;  
  
//===== Stel DB-konneksies op - MOENIE VERANDER NIE! =====  
//===== Stel DB-konneksies op - MOENIE VERANDER NIE! =====  
{ $REGION DB CONNECTION}  
// =====  
procedure TfrmVraag2.bmbHerstelDBClick(Sender: TObject);  
begin  
    // Restores the Database  
    dbCONN.RestoreDatabase;  
    redV2.Clear;  
    dbCONN.setupGrids(dbgWerknemers, dbgLogUre, dbgSQL);  
// redV2.Clear;  
    // Formatting field datatypes  
// tblWerknemers := dbCONN.tblOne;  
// tblLogUre := dbCONN.tblMany;  
// qryDB := dbCONN.qry;  
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  
    // Sets up the connection to database and opens the tables.  
    dbCONN := TConnection.Create;  
    dbCONN.dbConnect;  
    tblWerknemers := dbCONN.tblOne;  
    tblLogUre := dbCONN.tblMany;  
// qryDB := dbCONN.qry;  
    dbCONN.SetupGrids(dbgWerknemers, dbgLogUre, dbgSQL);  
    pgcTabs.ActivePageIndex := 0;  
end;  
// =====  
{ $ENDREGION}  
  
end.
```

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

```
unit Transaksie_U;

interface

type
  TTransaksie = class(TObject)
  private
    var
      fKlantID: String;
      fHouerGrootte: char;
      fStoorTydperk: integer;
      fBedragBetaal: real;
  public
    constructor create( sKlantID: String; cHouerGrootte: char;
      iStoorTydperk: integer);
    function getBedragBetaal: real;
    procedure opdateerBedragBetaal(rBedragBetaal: real);
    function berekenKoste: real;
    function toString: String;
  end;

implementation

{ TStorage }

uses
  SysUtils, Math;

//=====
//                      Vraag 3.1.1 - 4 punte
//=====

constructor TTransaksie.create(sKlantID: String;cHouerGrootte: char;
iStoorTydperk: integer );
begin
  fKlantID := sKlantID;
  fHouerGrootte := cHouerGrootte;
  fStoorTydperk := iStoorTydperk;
  fBedragBetaal := 0.00;
end;

//=====
//                      Vraag 3.1.2 - 2 punte
//=====

function TTransaksie.getBedragBetaal: real;
begin
  result := fBedragBetaal;
end;
```

```
//=====
//
//                               Vraag 3.1.3 - 3 punte
//=====
```

```
procedure TTransaksie.opdateerBedragBetaal(rBedragBetaal: real);
begin
    fBedragBetaal := fBedragBetaal + rBedragBetaal;
end;
```

```
//=====
//
//                               Vraag 3.1.4 - 11 punte
//=====
```

```
function TTransaksie.berekenKoste: real;
var
    rKoste, rPersentasieAfslag: real;
begin
    case fHouerGrootte of
        'K': rKoste := 1000.00;
        'M': rKoste := 1750.00;
        'G': rKoste := 2500.00;
    end;
    rKoste := fStoorTydperk * rKoste;
    rPersentasieAfslag := Floor(fStoorTydperk / 6) * 0.1;
    if rPersentasieAfslag > 0.5 then
        begin
            rPersentasieAfslag := 0.5;
        end;
    result := rKoste - (rKoste * rPersentasieAfslag);
end;
```

```
//=====
//
//                               Vraag 3.1.5 - 3 punte
//=====
```

```
function TTransaksie.toString: String;
begin
    result := 'Klant se ID: ' + fKlantID + #10 +
        'Houergrootte: ' + fHouerGrootte + #10 +
        'Stoortydperk (maande): ' + IntToStr(fStoorTydperk) + #10 +
        'Bedrag betaal: ' + Format('%.2m', [fBedragBetaal]);
end;

end.
```

HOOFVORMEENHEID:

```
unit Vraag3_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, CheckLst, ExtCtrls, Buttons, Spin, ComCtrls;

type
  TfrmVraag3 = class(TForm)
    gbxV3_2_1: TGroupBox;
    gbxVraghouerGrootte: TGroupBox;
    lstV3_2_1: TListBox;
    gbxKlientID: TGroupBox;
    edtV3_2_1: TEdit;
    gbxAantalMaande: TGroupBox;
    spnV3_2_1: TSpinEdit;
    gbxV3_2_3: TGroupBox;
    redV3_2: TRichEdit;
    btnV3_2_1: TButton;
    btnV3_2_3: TButton;
    btnHerstel: TButton;
    gbxV3_2_2: TGroupBox;
    btnV3_2_2_b: TButton;
    gbxBedrag: TGroupBox;
    edtV3_2_2: TEdit;
    pnlV3_2_2: TPanel;
    btnV3_2_2_a: TButton;
    procedure btnV3_2_1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure btnV3_2_3Click(Sender: TObject);
    procedure btnHerstelClick(Sender: TObject);
    procedure btnV3_2_2_bClick(Sender: TObject);
    procedure btnV3_2_2_aClick(Sender: TObject);
  private

  public

  end;

var
  frmVraag3: TfrmVraag3;

implementation

{$R *.dfm}

uses
  Transaksie_U;

var
  objTransaksie: TTransaksie;
```

```
//=====
//
//                               Vraag 3.2.1 - 7 punte
//=====
```

```
procedure TfrmVraag3.btnV3_2_1Click(Sender: TObject);
var
    sKlantID: String;
    cHouerGrootte: char;
    iStoorTydperk: integer;
begin
    // Vraag 3.2.1
    sKlantID := edtV3_2_1.Text;
    cHouerGrootte := lstV3_2_1.Items[lstV3_2_1.ItemIndex][1];
    iStoorTydperk := spnV3_2_1.Value;

    objTransaksie := TTransaksie.create(sKlantID,cHouerGrootte,
                                        iStoorTydperk);

    btnV3_2_1.Enabled := False;
end;
```

```
//=====
//
//                               Vraag 3.2.2 (a) - 3 punte
//=====
```

```
procedure TfrmVraag3.btnV3_2_2_aClick(Sender: TObject);
begin
    // Vraag 3.2.2 (a)
    pnlV3_2_2.Caption := 'Bedrag verskuldig: ' + Format('%.2m',
[objTransaksie.berekenKoste]);
end;
```

```
//=====
//
//                               Vraag 3.2.2 (b) - 5 punte
//=====
```

```
procedure TfrmVraag3.btnV3_2_2_bClick(Sender: TObject);
var
    rBedragBetaal: real;
begin
    // Vraag 3.2.2 (b)
    rBedragBetaal := StrToFloat(edtV3_2_2.Text);

    objTransaksie.opdateerBedragBetaal(rBedragBetaal);
    pnlV3_2_2.Caption := 'Bedrag verskuldig: ' + Format('%.2m',
[objTransaksie.berekenKoste - objTransaksie.getBedragBetaal]);
end;
```



```
//=====
//                               Vraag 3.2.3 - 2 punte
//=====

procedure TfrmVraag3.btnV3_2_3Click(Sender: TObject);
begin
    // Kode wat voorsien is
    redV3_2.Clear;
    //  Vraag 3.2.3
    redV3_2.Lines.Add(objTransaksie.toString);
end;

//=====
// Kode wat voorsien is
//=====

procedure TfrmVraag3.FormCreate(Sender: TObject);
begin
    CurrencyString := 'R';
    lstV3_2_1.ItemIndex := 0;
end;

procedure TfrmVraag3.btnHerstelClick(Sender: TObject);
begin
    FreeAndNil(objTransaksie);
    btnV3_2_1.Enabled := True;
end;

end.
```

BYLAAG H: OPLOSSING VIR VRAAG 4

```

unit Vraag4_U;

interface

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

type
  TfrmVraag4 = class(TForm)
    GroupBox1: TGroupBox;
    redV4_1: TRichEdit;
    btnV4_1_2: TButton;
    GroupBox2: TGroupBox;
    GroupBox3: TGroupBox;
    Image1: TImage;
    redV4_2: TRichEdit;
    pnlV4: TPanel;
    btnV4_2: TButton;
    btnV4_1_1: TButton;
    procedure btnV4_2Click(Sender: TObject);
    procedure btnV4_1_2Click(Sender: TObject);
    procedure btnV4_1_1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);

  private
    { Private declarations }
    tLeer : TextFile;
    arrHouers : array [1..50] of real;
    // arrSkip  : array [1..5, 1..10] of real;
  public
    { Public declarations }
  end;

//=====
// Kode wat voorsien is - MOENIE VERANDER NIE
//=====

const
  arrTempHouers : array[1..50] of real =
    (31.2,43.4,5.1,41.2,52.6,41.9,97,49.3,15.8,39.5,
     78.8,96.3,67.8,47.2,31.4,18.4,27,1.4,33.5,16.5,
     58.3,9.9,62.9,11.8,62.5,59,13.4,49.8,60.4,74.5,
     13.5,67.7,94,39.4,47.4,13.1,26.2,63,89,22.3,
     54,16.9,38.6,46.2,22.5,11.4,65.1,48.6,41.4,47.9);

var
  frmVraag4: TfrmVraag4;

implementation

{$R *.dfm}

//=====
Kopiereg voorbehou

```

```
//          Vraag 4.1.1 - 4 punte
//=====

procedure TfrmVraag4.btnV4_1_1Click(Sender: TObject);
var
  iRy : integer;
begin
  // Kode wat voorsien is
  btnV4_2.Enabled := false;
  redV4_2.Clear;
  redV4_1.Clear;
  pnlV4.Caption := ' ' ;

  // Vraag 4.1.1

  for iRy := 1 to 50 do
    arrHouers[iRy] := RoundTo(Random() + Random(99),-1);

  btnV4_1_2.Enabled := true;
  //for iRy := 1 to 50 do
  //arrHouers[iRy] := arrTempHouers[iRy];

end;
//=====
//          Vraag 4.1.2 - 8 punte
//=====

procedure TfrmVraag4.btnV4_1_2Click(Sender: TObject);
var
  iRy, iKolom, iIndeks : integer;
  sReel : String;

begin
  // Kode wat voorsien is
  redV4_1.Clear;
  btnV4_2.Enabled := true;

  // Vraag 4.1.2
  iIndeks := 0;
  for iRy := 1 to 5 do
    begin
      sReel := '';
      for iKolom := 1 to 10 do
        begin
          Inc(iIndeks);
          sReel := sReel + FloatToStr(arrHouers[iIndeks])+ #9;
        end;
      redV4_1.Lines.Add(sReel);
    end;

end;
```

```
//=====
//
//                               Vraag 4.2 - 18 punte
//=====

procedure TfrmVraag4.btnV4_2Click(Sender: TObject);
var
  rTotaleGewig : real;
  iRy,iKolom,iIndeks : integer;
  sReel : String;

begin
  // Kode wat voorsien is
  redV4_2.Clear;

  //Vraag 4.2

  rTotaleGewig := 0;
  iIndeks := 1;
  sReel := '';
  while (iIndeks <= 50) and (rTotaleGewig <= 200) do
    begin
      if (rTotaleGewig + arrHouers[iIndeks]<= 200) then
        rTotaleGewig := rTotaleGewig + arrHouers[iIndeks];

      Inc(iIndeks);
    end;

  // Alternatief
  { iRy := 1;
    iIndeks := 1;    //one dim array
    sReel := '';

  while (rTotaleGewig < 200) and (iRy < 5) do
    begin
      iKolom := 1;
      while (iKolom < 11) and (iIndeks <= 50) do
        begin
          arrSkip[iRy,iKolom] := arrHouers[iIndeks];
          if rTotaleGewig + arrSkip[iRy,iKolom] <= 200 then
            //rTotaleGewig + arrHouers[iIndeks] <= 200
            begin
              sReel := sReel + FloatToStr(arrSkip[iRy,iKolom])+#9 ;
              rTotaleGewig := rTotaleGewig + arrSkip[iRy,iKolom];
              Inc(iKolom);
            end;
          Inc(iIndeks);
        end;
      Inc(iRy);
    end; }
  AssignFile(tLeer, 'Tonne.txt');
  Rewrite(tLeer);
  writeln(tLeer, sReel);
  CloseFile(tLeer);
```

```
    redV4_2.Lines.Add(sReel);
    pnlV4.Caption := 'Totale gewig van die vraag: ' + FloatToStr
(rTotaleGewig );

end;

// Kode wat voorsien is
procedure TfrmVraag4.FormCreate(Sender: TObject);
begin
    btnV4_1_2.Enabled := false;
    btnV4_2.Enabled := false;
end;

end.
```