



# basic education

---

Department:  
Basic Education  
**REPUBLIC OF SOUTH AFRICA**

## **NASIONALE SENIOR SERTIFIKAAT**

**GRAAD 12**

**INLIGTINGSTEGNOLOGIE V1**

**NOVEMBER 2019 (2)**

**NASIENRIGLYNE**

**PUNTE: 150**

**Hierdie nasienriglyne bestaan uit 26 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.
- **BYLAE A, B, C en D** (bladsy 3–9) sluit die nasienrubriek vir elke om te gebruik vir enigeen van die twee programmeringstale in.
- **BYLAE E, F, G en H** (bladsy 10–26) bevat voorbeelde in programmeringskode van oplossings vir Vraag 1 tot Vraag 4.
- Kopieë van **BYLAE A, B, C, D** (bladsy 3–9) moet vir elke leerder gemaak word en tydens die nasiensessie voltooi word.

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

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
1.1	<b>Button [1.1 – Welkom-boodskap]</b> Verklaar 'n string veranderlike ✓ Onttrek naam ✓ Met opskrif (label) ✓ Verander lettertipe ('style') ✓ Verander lettergrootte ✓ Display message ✓	6	
1.2	<b>Knoppie [1.2 – Ewekansige getalle]</b> Genereer twee getalle ✓ in die korrekte reeks ✓ Bepaal of ✓ die eerste getal > tweede getal ✓ Stel tydelik = eerste getal ✓ Stel eerste getal = tweede getal ✓ Stel tweede getal = tydelik ✓ Vertoon opskrif 'Stygende volgorde:' ✓ Vertoon getalle gesorteer ✓ Bepaal gemiddeld: tel twee getalle bymekaar ✓ deel antwoord deur 2 ✓ Vertoon die gemiddeld ✓ geformateer tot een desimale plek ✓	13	
1.3	<b>Knoppie [1.3 – Vertoon die waarde van y]</b> Verklaar x en y as heelgetalle ✓ iX := spnV1_3.Value; ✓ iY := Ceil ✓ (20 / ✓ sqrt(iX)); ✓ Vertoon op paneel pnIV1_3 ✓ 'y = ' + IntToStr(iY) ✓	7	
1.4	<b>Knoppie [1.4 – Vertoon afvoer]</b> Vertoon kolom-opskrifte ✓ Stel die jaarteller gelyk aan 1 ✓ Lus terwyl ✓ balans >= 0 ✓ Bereken rentebedrag: balans * ✓ rente / 100 ✓ Bereken nuwe balans: balans + rentebedrag ✓ Vertoon in rachedit: jaarteller, balans ✓ rentebedrag, nuwe balans ✓ Vertoon in kolomformaat ✓ bedrae geformateer as geldeenheid met 2 desimale plekke ✓ balans = nuwe balans - 1000 ✓ Inkrementeer jaarteller ✓ Vertoon op rachedit: boodskap met jaarteller + 1 ✓	14	
<b>TOTAAL:</b>		<b>40</b>	

**BYLAE B****AFDELING B****VRAAG 2: NASIENRUBRIEK - DATABASISPROGRAMMERING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
2.1.1	<b>Knoppie [2.1.1 – Manlike gidse]</b>	4	
	<b>SQL:</b> SELECT Voornaam, Van FROM tblGidse WHERE Geslag = 'M' ORDER BY Van		
	<b>Konsepte:</b> SELECT regte velde ✓ FROM regte tabel ✓ WHERE Condition: Geslag = 'M' ✓ ORDER BY Van ✓ Asc		
2.1.2	<b>Knoppie [2.1.2 – Aktiwiteit]</b>	3	
	<b>SQL:</b> SELECT Voornaam, Van, Geslag, AantUitstappies FROM tblGidse WHERE Aktiwiteit = '' + sAkt + ''		
	<b>Konsepte:</b> SELECT regte velde ✓ FROM regte tabel ✓ WHERE Aktiwiteit = veranderlike (korrekte formaat) ✓		
2.1.3	<b>Knoppie [2.1.3 – Hoogste aantal uitstappies]</b>	5	
	SELECT Aktiwiteit, Geslag, max(AantUitstappies) as HoogsteAantal FROM tblGidse GROUP BY Aktiwiteit, Geslag		
	<b>Konsepte:</b> SELECT Aktiwiteit en Geslag ✓ Maksimum van AantUitstappies ✓ as HoogsteAantal ✓ FROM tblGidse GROUP BY Aktiwiteit ✓ en Geslag ✓		
2.1.4	<b>Knoppie [2.1.4 – Verwyder gidsinligting]</b>	3	
	DELETE FROM tblGidse WHERE GidsID = ' + sGidsID		
	<b>Konsepte:</b> DELETE FROM ✓ regte tabel ✓ WHERE GidsID = veranderlike (korrekte formaat) ✓		

2.1.5	<b>Knoppie [2.1.5 – Knysna en Montagu]</b>	7	
	<pre>SELECT (Van + '. ' + left(Voornaam,1)) As GidsNaam, Plek, tblUitstappies.Aktiwiteit FROM tblUitstappies,tblGidse WHERE tblUitstappies.Aktiwiteit = tblGidse.Aktiwiteit AND Plek IN ('Knysna','Montagu');</pre>		
	<b>Konsepte:</b> <pre>SELECT Plek, (Van + '. ' + left(Voornaam,1) ✓) ✓ AS GidsNaam FROM tblUitstappies, tblGidse ✓ WHERE tblUitstappies.Aktiwiteit = tblGidse. Aktiwiteit ✓ AND ✓ Plek IN ✓ ('Knysna','Montagu') ✓</pre>		
<b>Subtotaal: SQL</b>		<b>[22]</b>	

2.2.1	<b>Kombinasielys – cmbV2_2_1</b> Lees sAktiwiteit uit kombinasielys ('combo box') ✓ Gaan na eerste (first) rekord van tblUitstappies ✓ Lus terwyl nie einde van tabel ✓ if (sAktiwiteit = tblUitstappies ['Aktiwiteit']) ✓ sPlek := tblUitstappies ['Plek'] ✓ Laai prent in image-komponent deur naam van plek te gebruik ✓ Vertoon plek se naam op label lblV2_2_1 ✓ Gaan na volgende (next) rekord ✓	8	
2.2.2	<b>Knoppie [2.2.2 – Bespreking]</b> Onttrek groote van groep in ✓ Stel boolean veranderlike na false ✓ Gaan na eerste (first) rekord van tblUitstappies Lus terwyl nie einde van tabel tblUitstappies ✓ Toets of groote van die groep <= max groepgrootte ✓ en aktiwiteit = sAktiwiteit ✓ Vertoon bespreking wat by plek bevestig is ✓ Stel boolean op true ✓ Gaan na volgende (next) rekord in tblUitstappies ✓ As groepgrootte te groot is ✓ – vertoon geen bespreking ✓	10	
<b>Subtotaal: Delphi-kode</b>		<b>[18]</b>	

<b>TOTAAL AFDELING B:</b>		<b>40</b>	
---------------------------	--	-----------	--



3.2.1	<b>Knoppie [3.2.1 – Instansieer programmeerder-objek]</b>  <i>Instansieër die objProgrammeerder-objek:</i> objProgrammeerder := ✓ TProgrammeerder. Create ✓ Maak gebruik van Vollename, Vaardighede en Graderings veranderlikes ✓	3	
3.2.2	<b>Knoppie [3.2.2–Vertoon programmeerder se besonderhede]</b>  Vertoon objek in rich edit deur toString-metode ✓ Roep berekenGradering ✓	2	
3.2.3	<b>Knoppie[3.2.3 – Kontroleer beskikbaarheid en vaardighede]</b>  Toets of programmeerder vaardigheid het mbv hetVaardigheid-metode ✓ met regte argumente ✓ EN ✓ programmeerder beskikbaar is dmv getBeskikbaar-metode ✓ Vertoon Vollename en boodskap vir beskikbaarheid ✓ Stel btnV3_2_4 beskikbaar ✓ Indien programmeerder nie hetVaardigheid nie ✓ Vertoon boodskap dat hy nie vaardigheid het nie ✓ Indien programmeerder nie isBeskikbaar nie ✓ Vertoon boodskap dat hy nie beskikbaar is om in diens geneem te word nie ✓	10	
3.2.4	<b>Knoppie [3.2.4 – Neem programmeerder in diens]</b>  Roep setBeskikbaar met argument False ✓ Versteek btnV3_2_4 ✓ Vertoon naam van programmeerder ✓ en boodskap	3	
<b>TOTAAL AFDELING C</b>		<b>[40]</b>	

## BYLAE D

## VRAAG 4: NASIENRUBRIEK – PROBLEEMOPLOSSING

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
4.1	<p><b>Knoppie [4.1 – Bereken gemiddeld per rondte]</b></p> <p>Inisialiseer veranderlike vir saamstel van afvoerstring (sAfvoer) ✓  Lus deur kolomme ✓      Inisialiseer som ✓      Lus deur rye ✓          som = som + arrRondtePunte[iRy, iKol] ✓      Heg gemiddeld (som/6) aan afvoerstring ✓  Vertoon sAfvoer veranderlike in redV4 ✓</p>	7	
4.2	<p><b>Knoppie [4.2 – Bepaal beste rondte(s)]</b></p> <p>Onttrek ItemIndex uit cmbBoogskutter plus 1 ✓  Inisialiseer veranderlike om grootste waarde te bepaal ✓  Lus deur arrRondteTellings kolomme ✓      As arrRondteTellings [iSelekteer, iKol] &gt; grootste ✓          grootste = arrRondteTellings [iSelekteer, iKol] ✓  Lus deur arrRondteTellings kolomme ✓      As arrRondteTellings [iSelekteer, iKol] = grootste ✓          Voeg rondte en telling by afvoerstring ✓  Vertoon afvoerstring ✓</p>	9	



4.3	<p><b>Knoppie [4.3 – Bepaal mediaantelling]</b></p> <p><b>Konsepte:</b>                  Sorteër die data reg – 10 punte                  Regte indeksverwysings vir mediaan – 2 punte                  Tel middelste twee waardes bymekaar en deel deur 2 – 1 punt                  Vertoon mediaan – 1 punt</p> <p><b>1D skikkingoplossing</b>                  Verklaar gepaste skikking om te gebruik vir sortering                  Inisialiseer plekhouer as 'n teller van skikking ✓                  //Skuif van data van 2D skikking na 1D skikking                  Lus deur arrRondteTellings rye ✓                      Lus deur arrRondteTellings kolomme ✓                          arrRondteTellings [iTyd] = arrRondteTellings [i, j] ✓                          Inkrementeer plekhouer ✓                  Buitenste lus vir sortering ✓                      Binneste lus vir sortering ✓                      As arrTydTellings[i] &gt; arrTydTellings [j] ✓                          iTyd = arrTydTellings [i] ✓                          arrTydTellings [i] = arrTydTellings [j] } ✓                          arrTydTellings [j] = iTyd }                  Bepaal die eerste middelste element in die skikking ✓                  Bepaal die tweede middelste element in die skikking ✓                  Tel die twee middelste elemente bymekaar en deel deur 2 ✓                  Vertoon die mediaan ✓</p> <p><b>2D-skikkingoplossing</b>                  Lus i van 1 tot 5 (1 punt)                      Lus j van 1 tot 4 (1 punt)                          Lus k van 1 tot 5 (2 punt)                              Lus L van 1 tot 4 (2 punt)                                  As arr [i, j] (1 punt) &gt; arr [k, L] (1 punt)                                      iTyd = arr[i, j] (1 punt)                                      arr[i, j] = arr[k, L] } (1 punt)                                      arr[k, L] = iTyd }                  Bepaal die eerste middelste element in die skikking (1 punt)                  Bepaal die tweede middelste element in die skikking (1 punt)                  Tel die twee middelste elemente bymekaar en deel deur 2                  (1 punt)                  Vertoon die mediaan (1 punt)</p>	14	
<b>TOTAAL AFDELING D:</b> <b>GROOT TOTAAL:</b>		<b>30</b> <b>150</b>	

**OPSOMMING VAN LEERDER SE PUNTE:**

	AFDELING A	AFDELING B	AFDELING C	AFDELING D	
	VRAAG 1	VRAAG 2	VRAAG 3	VRAAG 4	GROOTTOTAAL
<b>MAKS. PUNTE</b>	40	40	40	30	150
<b>LEERDER SE PUNTE</b>					

**BYLAE E: OPLOSSING VIR VRAAG 1**

```
unit Vraag1_U;

interface

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

type
  TfrmVraag1 = class(TForm)
    gpbV1_2: TGroupBox;
    btnV1_2: TButton;
    gpbV1_3: TGroupBox;
    spnV1_3: TSpinEdit;
    Label6: TLabel;
    btnV1_3: TButton;
    gpbV1_4: TGroupBox;
    gpbV1_1: TGroupBox;
    Label1: TLabel;
    edtV1_1: TEdit;
    lblV1_1: TLabel;
    btnV1_1: TButton;
    redV1_2: TRichEdit;
    pnlV1_3: TPanel;
    edtV1_4: TEdit;
    Label2: TLabel;
    btnV1_4: TButton;
    redV1_4: TRichEdit;
    procedure btnV1_2Click(Sender: TObject);
    procedure btnV1_3Click(Sender: TObject);
    procedure btnV1_1Click(Sender: TObject);
    procedure btnV1_4Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag1: TfrmVraag1;

implementation

{$R *.dfm}
```

---

---

**Vraag 1.1 – 6 punte**

---

---

```
procedure TfrmVraag1.btnV1_1Click(Sender: TObject);
var
  sNaam: String;

begin
  // Vraag 1.1
  sNaam := edtV1_1.Text;

  with lblV1_1 do
  begin
    Font.Style := [fsBold];
    Font.Size := 14;
    Caption := 'Welkom ' + sNaam;
  end;
end;
```

---

---

---

---

**Vraag 1.2 – 13 punte**

---

---

```
procedure TfrmVraag1.btnV1_2Click(Sender: TObject);
var
  // Veranderlikes wat voorsien is
  iEersteGetal, iTweedeGetal: integer;
  rGemiddeld: real;
  // Voltooi
  iStoorGetal: integer;

begin
  // Kode wat voorsien is
  redV1_2.Clear;

  //Vraag 1.2
  iEersteGetal := randomrange(50, 76);
  iTweedeGetal := randomrange(50, 76);

  if iEersteGetal > iTweedeGetal then
  begin
    iStoorGetal := iEersteGetal;
    iEersteGetal := iTweedeGetal;
    iTweedeGetal := iStoorGetal;
  end;
  redV1_2.Lines.Add('Stygende volgorde:');
  redV1_2.Lines.Add(IntToStr(iEersteGetal));
  redV1_2.Lines.Add(IntToStr(iTweedeGetal));
  rGemiddeld := (iEersteGetal + iTweedeGetal) / 2;
  redV1_2.Lines.Add('Gemiddeld: ' + FloatToStrF(rGemiddeld, ffFixed, 7,
1));
end;
```

## =====

**Vraag 1.3 - 7 punte**

```
=====
procedure TfrmVraag1.btnV1_3Click(Sender: TObject);
var
    iX, iY: integer;

begin
    //Vraag 1.3

    iX := spnV1_3.Value;
    iY := Ceil(20 / sqrt(iX));
    pnlV1_3.Caption := 'y = ' + IntToStr(iY);
end;
=====
```

## =====

**Vraag 1.4 - 14 punte**

```
=====
procedure TfrmVraag1.btnV1_4Click(Sender: TObject);
// Konstante wat voorsien is
const
    RENTEKOERS = 8.1;

// Veranderlikes wat voorsien is
var
    rBalans, rRente, rNuweBalans: real;
    iTelJaar: integer;

begin
    // Kode wat voorsien is
    redV1_4.Clear;
    rBalans := StrToFloat(edtV1_4.Text);

    //Vraag 1.4

    redV1_4.Lines.Add('Jaar' + #9 + 'Balans' + #9 + 'Rente' + #9 + 'Balans
met rente');
    iTelJaar := 1;
    while rBalans >= 0 do
        begin
            rRente := rBalans * RENTEKOERS / 100;
            rNuweBalans := rBalans + rRente;
            redV1_4.Lines.Add(IntToStr(iTelJaar) + #9 + FloatToStrF(rBalans,
                ffCurrency, 8, 2) + #9 + FloatToStrF(rRente, ffCurrency, 8,
                2) + #9 + FloatToStrF(rNuweBalans, ffCurrency, 8, 2));
            rBalans := rNuweBalans - 1000;
            Inc(iTelJaar);
        end;
    redV1_4.Lines.Add
        (#13 + 'Aantal jaar om al die geld uit die rekening te onttrek: ' +
        IntToStr(iTelJaar));
end;
=====
```

**BYLAE F: OPLOSSING VIR VRAAG 2**

```
unit Vraag2_U;

interface

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

type
  TfrmVraag2 = class(TForm)
    pgcDatabasis: TPageControl;
    dbgGidse: TDBGrid;
    dbgUitstappies: TDBGrid;
    tshV2_1: TTabSheet;
    tshV2_2: TTabSheet;
    dbgSQL: TDBGrid;
    btnV2_1_1: TButton;
    btnV2_1_3: TButton;
    btnV2_1_2: TButton;
    btnHerstel: TBitBtn;
    btnV2_1_4: TButton;
    grpBox1: TGroupBox;
    cmbV2_2_1: TComboBox;
    btnV2_2_2: TButton;
    grpBox2: TGroupBox;
    GroupBox4: TGroupBox;
    GroupBox5: TGroupBox;
    imgV2_2_1: TImage;
    lblV2_2_1: TLabel;
    spnV2_2_2: TSpinEdit;
    redV2_2_2: TRichEdit;
    Label1: TLabel;
    btnV2_1_5: TButton;
    procedure FormCreate(Sender: TObject);
    procedure UpdateDBGrid();
    procedure btnV2_1_1Click(Sender: TObject);
    procedure btnV2_1_3Click(Sender: TObject);
    procedure btnV2_1_2Click(Sender: TObject);
    procedure btnV2_1_4Click(Sender: TObject);
    procedure cmbV2_2_1Change(Sender: TObject);
    procedure btnV2_2_2Click(Sender: TObject);
    procedure btnHerstelClick(Sender: TObject);
    procedure btnV2_1_5Click(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag2: TfrmVraag2;
```

```
database: TConnection;
qry: TADOQuery;
tblGidse, tblUitstappies: TADOTable;
dsrGidse, dsrUitstappies, dsrSQL: TDataSource;

// Veranderlike wat voorsien is - Vraag 2.2.1 en Vraag 2.2.2
sAktiwiteit: String;
```

```
implementation
```

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

### Vraag 2.1 – SQL-afdeling

---

#### Vraag 2.1.1 – 4 punte

---

```
procedure TfrmVraag2.btnV2_1_1Click(Sender: TObject);
var
  sSQL1: String;
begin
  // Vraag 2.1.1
  sSQL1 :=
    'SELECT Voornaam, Van FROM tblGidse WHERE Geslag = 'M' ORDER BY
    Van';

  // Kode wat voorsien is
  database.RunSQL(sSQL1);
  database.setupGrids(dbgUitstappies, dbgGidse, dbgSQL);
end;
```

---

#### Vraag 2.1.2 – 3 punte

---

```
procedure TfrmVraag2.btnV2_1_2Click(Sender: TObject);
var
  sSQL2: String;
  sAkt: String;
begin
  // Kode wat voorsien is
  sAkt := InputBox('Invoer', 'Sleutel die aktiwiteit van die gids in',
    'Fietsry');

  // Voltooi
  sSQL2 :=
    'SELECT Voornaam, Van, Geslag, AantUitstappies FROM tblGidse WHERE
    Aktiwiteit = ' + sAkt + ''';

  // Kode wat voorsien is
  database.RunSQL(sSQL2);
  database.setupGrids(dbgUitstappies, dbgGidse, dbgSQL);
end;
```

---

---

**Vraag 2.1.3 – 5 punte**

---

---

```
procedure TfrmVraag2.btnV2_1_3Click(Sender: TObject);
var
  sSQL3: String;
begin
  sSQL3 :=
    'SELECT Aktiwiteit, Geslag, max(AantUitstappies) as HoogsteAantal
FROM tblGidse GROUP BY Aktiwiteit, Geslag';

  // Gegewe kode
  database.RunSQL(sSQL3);
  database.setupGrids(dbgUitstappies, dbgGidse, dbgSQL);
end;
```

---

---

---

---

**Vraag 2.1.4 – 3 punte**

---

---

```
procedure TfrmVraag2.btnV2_1_4Click(Sender: TObject);
var
  sSQL4: String;
  sGidsID: String;
  bVerander: boolean;
begin
  sGidsID := InputBox('Verwyder gids',
    'IDnommer van gids wat uit die tabel verwyder moet word', '');

  sSQL4 := 'DELETE FROM tblGidse WHERE GidsID = ' + sGidsID;

  // Gegewe kode
  database.ExecuteSQL(sSQL4, bVerander);
  database.setupGrids(dbgUitstappies, dbgGidse, dbgSQL);
end;
```

---

---

---

---

**Vraag 2.1.5 – 7 punte**

---

---

```
procedure TfrmVraag2.btnV2_1_5Click(Sender: TObject);
var
  sSQL5: String;
begin
  sSQL5 :=
    'SELECT (Van + '.' + left(Voornaam,1)) As GidsNaam, Plek,
tblUitstappies.Aktiwiteit FROM tblUitstappies,tblGidse WHERE
tblUitstappies.Aktiwiteit = tblGidse.Aktiwiteit AND Plek IN
('Knysna','Montagu)';

  // Gegewe kode
  database.RunSQL(sSQL5);
  database.setupGrids(dbgUitstappies, dbgGidse, dbgSQL);
end;
```

**Vraag 2.2 - Delphi-afdeling**

## =====

**Vraag 2.2.1 - 8 punte**

```
=====
procedure TfrmVraag2.cmbV2_2_1Change(Sender: TObject);
var
  sPlek: String;
begin
  // Lees waarde van aktiwiteit
  sAktiwiteit := cmbV2_2_1.Text;
  tblUitstappies.First;
  while NOT tblUitstappies.eof do
  begin
    if sAktiwiteit = tblUitstappies['Aktiwiteit'] then
    begin
      sPlek := tblUitstappies['Plek'];
      imgV2_2_1.Picture.LoadFromFile(sPlek + '.jpg');
      lblV2_2_1.Caption := sPlek;
    end;
    tblUitstappies.Next;
  end;
end;
```

## =====

**Vraag 2.2.2 - 10 punte**

```
=====
procedure TfrmVraag2.btnV2_2_2Click(Sender: TObject);
var
  iGroepGrootte: integer;
  bGevind: boolean;
begin
  redV2_2_2.Clear;
  bGevind := false;
  iGroepGrootte := spnV2_2_2.Value;

  tblUitstappies.First;
  while NOT tblUitstappies.eof do
  begin
    if (tblUitstappies['Aktiwiteit'] = sAktiwiteit) then
    begin
      redV2_2_2.Lines.Add('Maksimum groepgrootte: ' + intToStr
        (tblUitstappies['MaksGroepGrootte']));
      if (iGroepGrootte <= tblUitstappies['MaksGroepGrootte']) then
      begin
        bGevind := true;
      end;
    end;
    tblUitstappies.Next;
  end;

  if NOT bGevind then
    redV2_2_2.Lines.Add('Groep te groot')
  else
    redV2_2_2.Lines.Add('Bespreking bevestig');
end;
```



```
=====
Stel DB-konneksie op - MOENIE VERANDER NIE!
=====
```

```
procedure TfrmVraag2.FormCreate(Sender: TObject);
begin
  database := TConnection.Create();
  database.DBConnect();
  tblUitstappies := database.tblOne;
  tblGidse := database.tblMany;
  database.setupGrids(dbgUitstappies, dbgGidse, dbgSQL);
  pgcDatabasis.ActivePageIndex := 0;
end;

procedure TfrmVraag2.btnHerstelClick(Sender: TObject);
var
  bFail: boolean;
begin
  database.RestoreDatabase;

  redV2_2_2.Clear;
  database.setupGrids(dbgUitstappies, dbgGidse, dbgSQL);
end;

procedure TfrmVraag2.UpdateDBGrid;
var
  i: integer;
begin
  for i := 0 to dbgSQL.Columns.Count - 1 do
  begin
    dbgSQL.Columns[i].Width := 79;
  end;
end;
end.
```

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

```
unit Programmeerder_U;

interface

type
  TProgrammeerder = class(TObject)
  private
    fVolleName, fVaardighede, fGraderings: String;
    fBeskikbaar: boolean;
  public
    constructor Create(sVolleName, sVaardighede, sGraderings: String);
    procedure setBeskikbaar(bBeskikbaar: boolean);
    function getVolleName: String;
    function getBeskikbaarheid: boolean;
    function berekenGradering: double;
    function hetVaardighede(sVaardighede: String): boolean;
    function toString: String;
  end;

implementation

uses
  SysUtils;
```

---

---

**Vraag 3.1.1 – 4 punte**

---

---

```
constructor TProgrammeerder.Create(sVolleName, sVaardighede,
sGraderings: String);
begin
  fVolleName := sVolleName;
  fVaardighede := sVaardighede;
  fGraderings := sGraderings;
  fBeskikbaar := True;
end;
```

---

---

**Vraag 3.1.2 – 2 punte**

---

---

```
procedure TProgrammeerder.setBeskikbaar(bBeskikbaar: boolean);
begin
  fBeskikbaar := bBeskikbaar;
end;
```

---

---

**Vraag 3.1.3 – 7 punte**

---

---

```
function TProgrammeerder.berekenGradering: double;
var
    i, iTotaal: integer;
begin
    iTotaal := 0;
    for i := 1 to Length(fGraderings) do
        begin
            iTotaal := iTotaal + StrToInt(fGraderings[i]);
        end;
    Result := iTotaal / Length(fGraderings);
end;
```

---

---

**Vraag 3.1.4 – 6 punte**

---

---

```
function TProgrammeerder.hetVaardighede(sVaardighede: String): boolean;
begin
    Result := Pos(sVaardighede, fVaardighede) > 0;
end;
```

---

---

**Vraag 3.1.5 – 3 punte**

---

---

```
function TProgrammeerder.toString: String;
var
    sBeskikbaar: String;
begin
```

---

---

**Kode wat voorsien is**

---

---

```
    if fBeskikbaar then
        begin
            sBeskikbaar := 'Ja';
        end
    else
        begin
            sBeskikbaar := 'Nee';
        end;
```

---

---

**Voltooi kode**

---

---

```
Result := 'Volle name: ' + fVolleName + #10 +
    'Vaardighede: ' + fVaardighede + #10 +
    'Graderings: ' + fGraderings + #10 +
    'Beskikbaar: ' + sBeskikbaar;
end;
```

## =====

**Kode wat voorsien is**

```
=====
function TProgrammeerder.getBeskikbaarheid: boolean;
begin
    Result := fBeskikbaar;
end;

function TProgrammeerder.getVolleName: String;
begin
    Result := fVolleName;
end;
end.
```

**HOOFVORMEENHEID:**

```
unit Vraag3_U;

interface

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

type
  TfrmVraag3 = class(TForm)
    btnV3_2_1: TButton;
    redV3_2_2: TRichEdit;
    btnV3_2_2: TButton;
    gbxV3_2_3: TGroupBox;
    cmbV3_2_3: TComboBox;
    btnV3_2_3: TButton;
    Label1: TLabel;
    btnV3_2_4: TButton;
    lblV3_2_4: TLabel;
    pnlV3_2_2: TPanel;
    pnlV3_2_3: TPanel;
    procedure btnV3_2_1Click(Sender: TObject);
    procedure btnV3_2_2Click(Sender: TObject);
    procedure btnV3_2_3Click(Sender: TObject);
    procedure btnV3_2_4Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag3: TfrmVraag3;

implementation

uses Programmeerder_U;

{$R *.dfm}

const
  sVolleName = 'Rupert Grint';
  sVaardighede = 'Webontwerp;JavaScript';
  sGraderings = '434545445';

var
  objProgrammeerder: TProgrammeerder;
```

---

---

**Vraag 3.2.1 – 3 punte**

---

---

```
procedure TfrmVraag3.btnV3_2_1Click(Sender: TObject);
begin
    objProgrammeerder := TProgrammeerder.Create(sVolleName, sVaardighede,
sGraderings);
end;
```

---

---

**Vraag 3.2.2 – 2 punte**

---

---

```
procedure TfrmVraag3.btnV3_2_2Click(Sender: TObject);
begin
    redV3_2_2.Text := objProgrammeerder.toString;
    pnlV3_2_2.Caption := 'Gemiddelde gradering: ' +
FloatToStrF(objProgrammeerder.berekenGradering, ffixed,8,2);
end;
```

---

---

**Vraag 3.2.3 – 10 punte**

---

---

```
procedure TfrmVraag3.btnV3_2_3Click(Sender: TObject);
var
    sVereisteVaardigheid: String;
begin
    sVereisteVaardigheid := cmbV3_2_3.Text;
    if (objProgrammeerder.hetVaardighede(sVereisteVaardigheid) = True) and
(objProgrammeerder.getBeskikbaarheid = True) then
        begin
            pnlV3_2_3.Caption := objProgrammeerder.getVolleName + ' is
besikbaar.';
            btnV3_2_4.Enabled := True;
        end
    else if objProgrammeerder.hetVaardighede(sVereisteVaardigheid) = False
then
        begin
            pnlV3_2_3.Caption := objProgrammeerder.getVolleName + ' het nie
daardie vaardighede nie.';
        end
    else if objProgrammeerder.getBeskikbaarheid = False then
        begin
            pnlV3_2_3.Caption := objProgrammeerder.getVolleName + ' is nie
besikbaar om in diens geneem te word nie.';
        end;
end;
```

---

---

**Vraag 3.2.4 – 3 punte**

---

---

```
procedure TfrmVraag3.btnV3_2_4Click(Sender: TObject);
begin
    objProgrammeerder.setBeskikbaar(false);
    btnV3_2_4.Hide;
    lblV3_2_4.Caption := objProgrammeerder.getVolleName + ' is suksesvol
aangestel!';
end;
end.
```

**BYLAE H: OPLOSSING VIR VRAAG 4**

```

unit Vraag4_U;

interface

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

type
  TfrmVraag4 = class(TForm)
    gpbV4_3: TGroupBox;
    gpbV4_2: TGroupBox;
    lblV4_2: TLabel;
    Label1: TLabel;
    Label2: TLabel;
    cmbBoogskutter: TComboBox;
    gpbV4_1: TGroupBox;
    redV4_1: TRichEdit;
    btnV4_1: TButton;
    btnV4_2: TButton;
    btnV4_3: TButton;
    pnlV4_3: TPanel;
    procedure btnV4_1Click(Sender: TObject);
    procedure btnV4_3Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure btnV4_2Click(Sender: TObject);
  private
    procedure VertoonToernooiData();
  public
    { Public declarations }
  end;

var
  frmVraag4: TfrmVraag4;

implementation

{$R *.dfm}

var
  arrBoogskutters: array [1 .. 6] of string = (
    'Nicolas Green',
    'Thomas Ndlovu',
    'Alice Garcia',
    'Jayshree Patel',
    'Tristan Evermore',
    'Eugene Geary'
  );
  arrRondtePunte: array [1 .. 6, 1 .. 4] of Integer = ((7, 7, 6, 7),
    (9, 10, 10, 8), (6, 6, 10, 8), (5, 10, 9, 6), (10, 4, 8, 4), (8, 5,
    7, 8));

```

## =====

**Vraag 4.1 - 7 punte**

```
=====
procedure TfrmVraag4.btnV4_1Click(Sender: TObject);
var
  iRy, iKol, iSom: Integer;
  sAfvoer: String;
begin
  // Vraag 4.1
  sAfvoer := 'Gemiddeld' + #9;
  for iKol := 1 to Length(arrRondtePunte[1]) do
  begin
    iSom := 0;
    for iRy := 1 to Length(arrRondtePunte) do
    begin
      iSom := iSom + arrRondtePunte[iRy, iKol];
    end;
    sAfvoer := sAfvoer + FloatToStrF(iSom / 6, ffFixed, 6, 2) + #9;
  end;
  redV4_1.Lines.Add(sAfvoer);
end;
```

## =====

**Vraag 4.2 - 9 punte**

```
=====
procedure TfrmVraag4.btnV4_2Click(Sender: TObject);
var
  iKol, iSelekteer, iMaks: Integer;
  sAfvoer: String;
begin
  // Vraag 4.2
  iSelekteer := cmbBoogskutter.ItemIndex + 1;

  iMaks := 0;
  for iKol := 2 to Length(arrRondtePunte[1]) do
  begin
    if iMaks < arrRondtePunte[iSelekteer, iKol] then
    begin
      iMaks := arrRondtePunte[iSelekteer, iKol];
    end;
  end;

  sAfvoer := '';
  for iKol := 1 to Length(arrRondtePunte[1]) do
  begin
    if iMaks = arrRondtePunte[iSelekteer, iKol] then
    begin
      sAfvoer := sAfvoer + 'Rondte ' + IntToStr(iKol) + ' (' +
      IntToStr(iMaks) + '), ';
    end;
  end;
  Delete(sAfvoer, Length(sAfvoer) - 1, 2);

  lblV4_2.Caption := sAfvoer;
end;
```



---

---

**Vraag 4.3 – 14 punte**

---

---

```
procedure TfrmVraag4.btnV4_3Click(Sender: TObject);
var
  i, j, iTydelik: Integer;
  arrTydPunte: array [1 .. 24] of Integer;
  dMediaan: double;
begin
  // Vraag 4.3
  iTydelik := 1;
  for i := 1 to Length(arrRondtePunte) do
  begin
    for j := 1 to Length(arrRondtePunte[1]) do
    begin
      arrTydPunte[iTydelik] := arrRondtePunte[i, j];
      Inc(iTydelik);
    end;
  end;

  for i := 1 to Length(arrTydPunte) - 1 do
  begin
    for j := i + 1 to Length(arrTydPunte) do
    begin
      if arrTydPunte[i] > arrTydPunte[j] then
      begin
        iTydelik := arrTydPunte[i];
        arrTydPunte[i] := arrTydPunte[j];
        arrTydPunte[j] := iTydelik;
      end;
    end;
  end;

  dMediaan := (arrTydPunte[Length(arrTydPunte) div 2] + arrTydPunte
    [Length(arrTydPunte) div 2 + 1]) / 2;
  pnlV4_3.Caption := 'Die mediaantelling is: ' + FloatToStr(dMediaan);
end;
{$REGION}
```

---

---

**Kode wat voorsien is – moet nie verander nie**

---

---

```
procedure TfrmVraag4.VertoonToernooiData;
var
  sLyn: string;
  iRy, iKol: Integer;
begin
  sLyn := '';
  for iKol := 1 to Length(arrRondtePunte[1]) do
  begin
    sLyn := sLyn + #9 + 'Rondte ' + IntToStr(iKol);
  end;
  redV4_1.Lines.Add(sLyn);

  for iRy := 1 to Length(arrRondtePunte) do
  begin
    sLyn := arrBoogskutters[iRy];
```

```
    for iKol := 1 to Length(arrRondtePunte[1]) do
    begin
        sLyn := sLyn + #9 + IntToStr(arrRondtePunte[iRy, iKol]);
    end;
    redV4_1.Lines.Add(sLyn);
end;
end;

procedure TfrmVraag4.FormCreate(Sender: TObject);
var
    i: Integer;
begin
    // Kode wat voorsien is - moenie verander nie.
    redV4_1.Paragraph.TabCount := 7;
    redV4_1.Paragraph.Tab[1] := 140;
    redV4_1.Paragraph.Tab[2] := 220;
    redV4_1.Paragraph.Tab[3] := 300;
    redV4_1.Paragraph.Tab[4] := 380;
    redV4_1.Paragraph.Tab[5] := 460;

    for i := 1 to Length(arrBoogskutters) do
    begin
        cmbBoogskutter.Items.Add(arrBoogskutters[i]);
    end;
    cmbBoogskutter.ItemIndex := 0;

    VertoonToernooiData();

end;
{$ENDREGION}

end.
```