



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

NASIONALE SENIOR SERTIFIKAAT

GRAAD 12

INLIGTINGSTEGNOLOGIE V1

FEBRUARIE/MAART 2015

MEMORANDUM

PUNTE: 150

Hierdie memorandum bestaan 32 bladsye.

ALGEMENE INLIGTING:

- Hierdie nasienriglyne moet as basis tydens die nasiensessie gebruik word. Dit is geskep om deur die nasieners gebruik te word. Daar word van alle nasieners verwag om 'n standaardiseringsvergadering by te woon om te verseker dat die riglyne konsekwent geïnterpreteer en toegepas word tydens die nasien van kandidate se werk.
- Let daarop dat kandidate wat alternatiewe korrekte oplossings voorsien as dié wat as voorbeelde van oplossings in hierdie nasienriglyne gegee is, volle krediet vir die betrokke antwoord sal ontvang tensy die instruksies in die vraestel nie gevolg is nie of die vereistes van die vraag nie nagekom word nie.
- **Bylae A, B en C** (bladsy 3–7) bevat die nasienrubriek vir elke vraag ongeag watter een van die twee programmeringstale gebruik is.
- **Bylae D, E en F** (bladsy 8–17) bevat voorbeelde van oplossings in programmeringskode vir Java vir Vraag 1 tot 3.
- **Bylae G, H en I** (bladsy 18–32) bevat voorbeelde van oplossings in programmeringskode vir Delphi vir Vraag 1 tot 3.
- Kopieë van **Bylae A, B en C** (bladsy 3–7) moet vir elke kandidaat gemaak word en tydens die nasiensessie voltooi word.

BYLAE A:**AFDELING A:****VRAAG 1: NASIENRUBRIEK – ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER SE PUNT
1.1	[Knoppie – Genereer advertensie] Onttrek die markprys ✓ Onttrek die verkoopprijs ✓ Onttrek die getal slaapkamers ✓ Onttrek die getal badkamers ✓ Gebruik die inligting wat onttrek is om die advertensie-reël saam te stel ✓ Kontroleer of swembad geselekteer is ✓ Voeg Swembad# by advertensie ✓ Kontroleer of verkoopprijs < markprys ✓ Voeg Winkoop by advertensie ✓ Vertoon opskrif in die afvoer-area ✓ Vertoon advertensie in die afvoer-area ✓	11	
1.2.1	Radioknoppie Verwerk: Stoor geselekteerde opknappingstipe as V ✓	1	
1.2.2	Radioknoppie Teelwerk: Stoor geselekteerde opknappingstipe as T ✓	1	
1.2.3	[Knoppie – Bereken en vertoon opkappingskoste] (a) Inisialiseer alle tel-veranderlikes na 0 ✓ Skep en inisialiseer koste-veranderlike ✓ Onttrek die oppervlakte uit die teksblokkie en skakel om na desimale waarde ✓ Kontroleer of die tipe opknapping verwerk is ✓ Bereken die volume verf benodig in liter. Formule: $\text{getalLiter} = (\text{oppervlakte} / 8)$ ✓ Rond getalLiter op ✓ Bereken die hoeveelheid van elke dromgrootte : As die getal liter meer is as 5 ✓ Bereken hoeveelheid 5 liter-dromme ✓ Bereken koste van die getal 5 liter-dromme ✓ Bereken die oorblywende verf benodig ✓ Herhaal vir: 2 liter-dromme ✓ Ken die oorblywende verf toe aan een liter-dromme ✓ Bereken koste ✓ Vertoon oppervlakte en volume verf in liter ✓ Vertoon elke drom-grootte ✓ en getal dromme benodig ✓ Vertoon met twee desimale plekke ✓	17	

(b)	Kontroleer of die tipe opknapping teelwerk is ✓ Gebruik dialoogblokkie om die koste per vierkante meter in te sleutel ✓ Bereken die koste van teelwerk deur van die onderstaande formule gebruik te maak: $\text{KosteVanTeelwerk} = (\text{oppervlakte} + 5) \times \text{kostePerVierkanteMeter} \checkmark^*$ Vertoon opskrif ✓ Vertoon koste van teelwerk ✓	6	
1.3.1	[Knoppie – Bereken bedrag verskuldig] Onttrek vorige lesing uit teksblokkie as heelgetal-waarde ✓ Onttrek huidige bedrag uit teksblokkie as heelgetal-waarde ✓ As huidige lesing < vorige lesing ✓ Stel huidige lesing na 0 ✓ en vertoon boodskap ✓ anders $\text{elekVerbruik} = (\text{huidigeLesing} - \text{vorigeLesing}) \checkmark$ as $\text{elekVerbruik} \leq 600 \checkmark$ $\text{elekRekening} = \text{elekVerbruik} \checkmark$ anders $\text{elekRekening} = 600 \checkmark + (\text{elekVerbruik} - 600) \times 1.5 \checkmark$ Vertoon elektrisiteitsrekening in die byskrif-komponent wat voorsien is Formaat: geldeenheid en twee desimale plekke ✓	11	
1.3.2	[Knoppie – Lys Geisers] Onttrek die grootte van die geiser uit die teksblokkie ✓ Gebruik lus om deur skikking te stap ✓ As die inskrywing in die skikking die ingesleutelde grootte van geiser bevat ✓ Begin die if-blok met kode ✓ Onttrek die naam van die geiser uit die inskrywing in die skikking ✓ Vertoon die naam van die geiser in die teks-area ✓	6	
	TOTAAL:	23	

BYLAE B:**AFDELING B:****VRAAG 2: NASIENRUBRIEK – OBJEK-GEORIËNTEERDE PROGRAMMERING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER SE PUNT
2.1.1	Vyf attribute is verklaar✓ as privaat ✓ String-datatype: aansoeker naam } Heelgetal-datatype: jare } ✓ Reële datatype: besteebare inkomste, lening-bedrag, rentekoers ✓	4	
2.1.2	Konstruktor: Opskrif ✓ met drie parameters✓ Parameters korrekte data tipes ✓ Ken parameterwaardes aan drie korrekte attribute toe ✓ Verstekwaardes toegeken aan jare en rentekoers ✓	5	
2.1.3	Toegangsmetode ('Accessor') vir naam van aansoeker: Metode se opskrif ✓ return-stelling ✓	2	
2.1.4	Wysigingsmetode ('Mutator') vir jare en rentekoers: setJare: parameter is ontvang ✓ en toegeken aan regte attribuut✓ setRentekoers: is ontvang ✓ en toegeken aan regte attribuut✓	4	
2.1.5	berekenPaalement-metode: Haal kommentaar van kode af ✓	1	
2.1.6	isGoedgekeur-metode: Skep boolese veranderlike vir goedgekeur wat op true ✓ gestel is As leningBedrag > 800000 en ✓ jare < 25✓ verander goedgekeur na false ✓ As leningBedrag <= 600000 en ✓ besteebareInkomste < ✓ 130% van paalement ✓ verander goedgekeur na false ✓ Stuur goedgekeur terug ✓	9	
2.1.7	toString-metode: Uitleg van string-veranderlike: Elke deel van die string op 'n nuwe reël ✓, korrekte byskrifte ✓ Korrekte inligting: Aansoeker se naam, besteebare inkomste ✓, lening-bedrag, jare ✓, rentekoers met %-teken ✓	5	

2.2.1	<p>[Knoppie – Vind aansoeker]</p> <p>Onttrek die aansoeker se naam uit die kombinasielys ✓ As 'n naam nie geselekteer is nie ✓ vertoon geskikte boodskap ✓ As 'n naam geselekteer is ✓ vind die tekslêer wat dieselfde naam as die aansoeker het ✓ As die lêer bestaan ✓ maak die tekslêer oop ✓ en lees die volgende uit tekslêer: inkomste, uitgawe ✓ lening-bedrag ✓ Bereken die besteebare inkomste ✓ Instansieer (skep) 'n nuwe lening-objek ✓ deur gebruik te maak van die korrekte parameters ✓✓ Vertoon geskikte boodskap deur gebruik te maak van 'n dialoogblokkie ✓ Maak die "Evalueer aansoek"-knoppie aktief ✓ As die lêer nie bestaan nie, ✓ Vertoon 'n geskikte boodskap ✓ Maak gebruik van 'n korrekte validasietegniek om uit 'n tekslêer te lees. ✓</p>	18	
2.2.2	<p>[Knoppie – Evalueer aansoek]</p> <p>Onttrek die rentekoers uit die teksblokkie ✓ en skakel om na reële getal ✓ Lees jare uit die kombinasielys ✓ en verander na heelgetal ✓ Stel die rentekoers van die objek deur van die 'mutator'-metode gebruik te maak ✓ Stel die jare van die objek deur van die 'mutator'-metode gebruik te maak ✓ Kontroleer of die lening goedgekeur is deur die relevante metode te roep ✓ As goedgekeur ✓ vertoon die inligting van die huislening-objek deur van die toString-metode gebruik te maak ✓ vertoon byskrif met paalement ✓ anders ✓ vertoon die naam van die aansoeker deur gebruik te maak van die 'accessor'-metode en vertoon die boodskap 'NIE GOEDGEKEUR NIE' ✓</p>	12	
	TOTAAL:	60	

BYLAE C:**AFDELING C:****VRAAG 3: NASIENRUBRIEK – PROBLEEMOPLOSSINGPROGRAMMERING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER SE PUNT
3.1	[Knoppie – Search for agent's name] Skep veranderlikes wat nodig word ✓ Gebruik lus om deur om deur skikking met die naam arrAgente ✓ Toets of kode in die skikkingsinskrywing ✓ die kode wat ingetik is, bevat ✓ Onttrek agent se naam uit die skikkingsinskrywing ✓ Vertoon die naam op die byskrif wat voorsien is ✓ Staak die soek-proses ✓ Inkrementeer teller ✓ As die agent se kode nie gevind is nie ✓ Vertoon die boodskap 'Ongeldige agentkode' in die byskrif ✓ Maak die teksblokkie vir agent se kode skoon ✓	11	
3.2.1	Voeg 'n geskikte komponent soos 'n tabel/matriks/afvoer-area op die GJK (GUI) ✓ Herbenoem die komponent ✓	2	
3.2.2	[Knoppie – Agent se verkope] Skep 'n 2d-skikking of enige ander struktuur/strukture vir verwerking ✓✓ Inisialiseer die 2d-skikking/ander struktuur/strukture ✓✓ Skep 'n veranderlike vir totale verkope ✓ Gebruik lus om deur die skikking met die naam arrVerkope te stap ✓ As agent se kode = kode in skikkingsinskrywing ✓ Onttrek maand uit skikkingsinskrywing ✓ Stel kolom se waarde deur maand te gebruik ✓ Onttrek tipe eiendom uit skikkingsinskrywing ✓ Stel ry se waarde deur tipe eiendom te gebruik ✓✓✓ Vermeerder die waarde van 2dArr[ry][kol] met 1 ✓ Vermeerder die waarde van 2dArr[ry][4] met 1 vir ry met totale ✓ Vermeerder die totale verkope veranderlike met die prys in skikkingsinskrywing ✓✓ Vertoon die inhoud van die 2d-skikking (of ander struktuur/e) in geskikte komponent, geformateer en gespaseer. Vertoon die volgende inligting: Maande as opskrifte ✓ Tipe eiendomme as byskrifte ✓ Ry- en kolomwaardes ✓✓✓ Totale van rye ✓ Vertoon die totale waarde van verkope in 'n geskikte komponent ✓	24	
TOTAAL:		37	

OPSOMMING VAN LEERDER SE PUNTE:

	AFDELING A	AFDELING B	AFDELING C	
	VRAAG 1	VRAAG 2	VRAAG 3	GROOTTOTAAL
MAKS. PUNTE	53	60	37	150
LEERDER SE PUNTE				

BYLAE D: OPLOSSING VIR VRAAG 1: JAVA

```
//      'n Moontlike oplossing vir Vraag 1

//=====
//Hierdie kode is in die program gegee
//=====

    public Vraag1_Memo() {

        initComponents();
        this.setLocationRelativeTo(this);
    }

//=====
// Nuwe kode
//=====

DecimalFormat df = new DecimalFormat("R 0.00");

    char opknappingsTipe = ' ';
    double eletrRekening;

//=====
//      Vraag 1.1
//=====
    private void btnVerkoopsAdvActionPerformed(java.awt.event.ActionEvent
evt) {
        double mark = Double.parseDouble(txfMarkwaarde.getText());
        double verkoop = Double.parseDouble(txfVerkoopsprys.getText());
        String bed = txfBeddens.getText();
        String badkamer = txfBadkamers.getText();
        String advertensie = "R" + (int)(verkoop) + "#" + bed + "Bed#"
                            + badkamer + "Bad#";

        if (chkSwembad.isSelected()) {
            advertensie += "Swembad#";
        }
        if (verkoop < mark) {
            advertensie += "Winskoop";
        }
        txafvoer.setText("Huis te koop:\n");
        txafvoer.append(advertensie);
    }

//=====
//      Vraag 1.2.1
//=====
    private void rbtVerfActionPerformed(java.awt.event.ActionEvent evt) {
        opknappingsTipe = 'V';
    }

//=====
//      VRAAG 1.2.2
//=====
    private void rbtTeelActionPerformed(java.awt.event.ActionEvent evt) {
        opknappingsTipe = 'T';
    }

```

```
//=====
//  Vraag 1.2.3 (a)
//=====
private void btnBerekOpknappingActionPerformed(java.awt.event.ActionEvent
evt) {
    int aantEenLiterDrom = 0;
    int aantTweeLiterDrom = 0;
    int aantVyfLiterDrom = 0;
    double koste = 0.0;
    double oppervlakte = Double.parseDouble(txfOppervlakte.getText());
    if (opknappingsTipe == 'V') {
        double aantLiterVerf = oppervlakte / 8;
        int volumeLiter = (int) (Math.round(aantLiterVerf + 0.4));
        if (volumeLiter / 5 > 0) {
            aantVyfLiterDrom += (int) (volumeLiter / 5);
            volumeLiter -= aantVyfLiterDrom * 5;
            koste += aantVyfLiterDrom * 199;
        }
        if (volumeLiter / 2 > 0) {
            aantTweeLiterDrom += (int) (volumeLiter / 2);
            volumeLiter -= aantTweeLiterDrom * 2;
            koste += aantTweeLiterDrom * 92.30; }
        aantEenLiterDrom = volumeLiter;
        koste += aantEenLiterDrom * 55.50;
        txaAfvoerOpknapping.setText("Oppervlakte: " +
            (int)(oppervlakte) + " vierkante meter");
        txaAfvoerOpknapping.append("\nVolume verf wat nodig word: " +
            aantLiterVerf + " liter");
        txaAfvoerOpknapping.append("\n\n1 liter-dromme: " +
            aantEenLiterDrom);
        txaAfvoerOpknapping.append("\n2 liter-dromme: " +
            aantTweeLiterDrom);
        txaAfvoerOpknapping.append("\n5 liter-dromme: " +
            aantVyfLiterDrom);
        txaAfvoerOpknapping.append("\n\nTotale koste: " +
            df.format(koste)); }
    if (opknappingsTipe == 'T') {
        double teelKoste =
            Double.parseDouble(JOptionPane.showInputDialog("Sleutel die
            koste per vierkante meter in"));
        koste = (oppervlakte + 5) * teelKoste;
        txaAfvoerOpknapping.setText("Oppervlakte: " +
            Math.round(oppervlakte) + " vierkante meter" + "\nTotale
            koste: " + df.format(koste));}
//=====
//  Vraag 1.2.3 (b)
//=====
if (opknappingsTipe == 'T') {
    double teelKoste =
        Double.parseDouble(JOptionPane.showInputDialog("Sleutel die koste per
        vierkante meter in"));
    koste = (oppervlakte + 5) * teelKoste;
    txaAfvoerOpknapping.setText("Oppervlakte: " +
        Math.round(oppervlakte) + " vierkante meter" + "\nTotale koste: " +
        df.format(koste));
} }

```

```
//=====
//  Vraag 1.3.1
//=====
private void btnBerekenBedragActionPerformed(java.awt.event.ActionEvent
evt) {
    int vorigeLesing = 0;
    int huidigeLesing = 0;
    double eletrRekening = 0;
    vorigeLesing = Integer.parseInt(txfVorige.getText());
    huidigeLesing = Integer.parseInt(txfHuidige.getText());
    if (vorigeLesing > huidigeLesing) {
        JOptionPane.showMessageDialog(null, "Vorige lesing is groter as
huidige lesing, sleutel weer in");
        txfHuidige.setText("0");
    } else {
        int elektrGebruik = huidigeLesing - vorigeLesing;
        if (elektrGebruik <= 600) {
            eletrRekening = elektrGebruik;
        } else {
            eletrRekening = 600 + ((elektrGebruik - 600) * 1.5);
        }
    }
    lblElektrisiteit.setText(df.format(eletrRekening));
}

//=====
//  Vraag 1.3.2
//=====

private void btnVindGeisersActionPerformed(java.awt.event.ActionEvent evt)
{
    String geiserGrootte = txfGeiserGrootte.getText().trim();
    txaAfvoerGeisers.setText("");
    for (int i = 0; i < arrGeisers.length; i++) {
        int psnDash = arrGeisers[i].indexOf("-");
        String grootte = arrGeisers[i].substring(0, psnDash);
        if (grootte.equals(geiserGrootte)) {
            txaAfvoerGeisers.append(arrGeisers[i].substring(psnDash +
1) + "\n");
        }
    }
}
```

BYLAE E: OPLOSSING VIR VRAAG 2: JAVA**Objekklas: Huislening.java**

```
//      'n Oplossing vir Vraag 2
//=====
//  Vraag 2.1.1
//=====
    private String aansoekerNaam;
    private double besteebareInkomste;
    private double leningBedrag;
    private int jare;
    private double renteKoers;
//=====
//  Vraag 2.1.2
//=====
    public HuisLening(String aansoekerNaam, double besteebareInkomste, double
leningBedrag) {
        this.aansoekerNaam = aansoekerNaam;
        this.besteebareInkomste = besteebareInkomste;
        this.leningBedrag = leningBedrag;
        this.jare = 0;
        this.renteKoers = 0;

}
//=====
//  VRAAG 2.1.3
//=====
    public String getAansoekerNaam() {
        return aansoekerNaam;
    }
//=====
//  Vraag 2.1.4
//=====
    public void setJare(int jare) {
        this.jare = jare;
    }
    public void setRenteKoers(double renteKoers) {
        this.renteKoers = renteKoers;
    }
//=====
//  Vraag 2.1.5 - Kode wat gegee is (verwyder kommentaartekens)
//=====
    public double berekenPaaieiment() {
        double paaieiment, koers;
        int result;
        koers = renteKoers / 100.0;
        paaieiment = (koers * leningBedrag) / (1 - ((Math.pow((1 + koers),
            (jare * -1)))));
        result = (int) (Math.round(paaieiment / 12));
        return result;
    }
}
```

```
//=====
//  Vraag 2.1.6
//=====
public boolean isGoedgekeur() {
    boolean goedgekeur = true;
    if (leningBedrag > 800000 && jare < 25) {
        goedgekeur = false;
    }
    if (leningBedrag <= 600000) {
        if (besteembareInkomste < (berekenPaalement() * 1.3)) {
            goedgekeur = false;
        }
    }
    return goedgekeur;
}

//=====
//  Vraag 2.1.7
//=====
public String toString() {
    String afvoer = "Naam van aansoeker: "+aansoekerNaam+
        "\nBesteembare inkomste: " + df.format(besteembareInkomste) +
        "\nLeningbedrag: " + df.format(leningBedrag) + "\nGetal jare: " +
        jare + "\nRentekoers: " + renteKoers + "%";
    return afvoer;
}

//=====
```

GGK-klas ('GUI class'): Vraag2.java

```
//=====
//  Kode wat gegee is
//=====
HuisLening leningObj;
//=====
//  Vraag 2.2.1
//=====
private void btnVindAansoekerActionPerformed(java.awt.event.ActionEvent
evt) {
    String aansoeker = (String) (cmbAansoekers.getSelectedItem());
    if (aansoeker.isEmpty()) {
        JOptionPane.showMessageDialog(rootPane, "Kies 'n aansoeker",
            "Warning", 0);
        txaAfvoer.setText("");
    }
    else {
        try {
            Scanner sc = new Scanner(new FileReader(aansoeker + ".txt"));
            txaAfvoer.setText("\n Huisleningaansoek is ingedien deur " +
                aansoeker);

            String naam = sc.nextLine();
            double inkomste = Double.parseDouble(sc.nextLine());
            double uitgawes = Double.parseDouble(sc.nextLine());
            double leningBedrag = Double.parseDouble(sc.nextLine());
            double besteembareInkomste = inkomste - uitgawes;
            leningObj = new HuisLening(aansoeker, besteembareInkomste,
                leningBedrag);
            JOptionPane.showMessageDialog(null, "Huisleningobjek suksesvol
```

```
        geskep");
        btnEvalueer.setEnabled(true);
    } catch (FileNotFoundException e) {
        txafvoer.setText("Geen huislening aansoek is deur " + aansoeker
            + " ingedien nie.");
    }
}

//=====
//  Vraag 2.2.2
//=====

private void btnEvalueerActionPerformed(java.awt.event.ActionEvent evt) {
    int jare = Integer.parseInt((String) (cmbJare.getSelectedItem()));
    double rentekoers = Double.parseDouble(txfRentekoers.getText());
    leningObj.setRentekoers(rentekoers);
    leningObj.setJare(jare);
    if (leningObj.isGoedgekeur()) {
        txafvoer.setText(leningObj.toString() + "\n\nLening GOEDGEKEUR
            met 'n maandlikse paaient van " +
            df.format(leningObj.berekenPaaient()));
    }
    else {
        txafvoer.setText("Naam van aansoeker: " +
            leningObj.getAansoekerNaam() + "\n");
        txafvoer.append("Huislening NIE GOEDGEKEUR NIE");
    }
}
```

BYLAE F: OPLOSSING VIR VRAAG 3: JAVA

```
//      'n Moontlike oplossing vir Vraag 3

public class Vraag3_Memo extends javax.swing.JFrame
{
//=====
//Hierdie kode is gegee in die program
//=====

// Klas wat gegee is

public String [] populateArrVerkope(){
    String[] arrVerkope = {"01#K#A122;500000", "01#K#A120;1200000",
        "01#K#A123;850000", "01#R#A121;75000", "02#K#A123;6200000",
        "02#L#A121;190000", "02#K#A121;500000", "02#K#A120;1200000",
        "02#K#A120;850000", "02#R#A124;75000", "02#K#A123;620000",
        "02#L#A124;690000", "02#K#A121;500000", "02#K#A120;1200000",
        "02#K#A120;850000", "02#R#A124;75000", "02#K#A123;620000",
        "02#L#A124;790000", "02#R#A124;1500000", "02#K#A125;200000",
        "02#L#A121;400000", "03#K#A120;175000", "03#K#A123;6200000",
        "03#L#A123;190000", "03#K#A124;500000", "03#K#A125;1200000",
        "03#K#A123;850000", "03#R#A122;75000", "03#K#A123;620000",
        "03#L#A123;890000", "03#R#A123;1500000", "03#K#A125;200000",
        "03#L#A123;400000", "04#R#A124;175000", "04#K#A120;6200000",
        "04#L#A123;190000", "03#R#A124;1500000", "03#K#A125;200000",
        "03#L#A122;400000", "04#R#A121;175000", "04#K#A123;6200000",
        "04#L#A123;190000", "04#K#A121;500000", "04#K#A120;1200000",
        "04#K#A120;850000", "04#R#A124;75000", "04#K#A123;620000",
        "04#L#A123;590000", "04#K#A123;500000", "04#K#A125;120000",
        "04#K#A124;850000", "04#R#A122;75000", "04#K#A123;6200000",
        "04#L#A120;490000", "04#K#A125;500000", "04#K#A125;120000",
        "04#K#A124;850000", "04#R#A122;75000", "04#K#A123;6200000",
        "04#L#A120;390000", "04#R#A120;1500000", "04#K#A121;200000",
        "04#L#A122;400000", "04#R#A124;175000", "04#K#A121;6200000",
        "04#L#A123;190000", "11#K#A122;500000", "07#K#A120;1200000",
        "10#K#A123;850000", "12#R#A121;75000", "12#K#A123;6200000",
        "11#L#A121;190000", "12#K#A121;500000", "12#K#A120;1200000",
        "11#K#A120;850000", "12#R#A124;75000", "12#K#A123;620000",
        "11#L#A124;690000", "09#K#A121;500000", "08#K#A120;1200000",
        "11#K#A120;850000", "12#R#A124;75000", "12#K#A123;620000",
        "10#L#A124;790000", "08#R#A124;1500000", "08#K#A125;200000",
        "11#L#A121;400000", "11#R#A120;175000", "07#K#A123;6200000",
        "09#L#A123;190000", "07#K#A124;500000", "07#K#A125;1200000",
        "12#K#A123;850000", "10#R#A122;75000", "08#K#A123;620000",
        "08#L#A123;890000", "06#R#A123;1500000", "09#K#A125;200000",
        "09#L#A123;400000", "09#R#A124;175000", "08#K#A120;6200000",
        "07#L#A123;190000", "08#R#A124;1500000", "06#K#A125;200000",
        "07#L#A122;400000", "08#R#A121;175000", "08#K#A123;6200000",
        "07#L#A123;190000", "11#K#A121;500000", "06#K#A120;1200000",
        "08#K#A120;850000", "08#R#A124;75000",
        "12#K#A123;620000", "07#L#A123;590000",
        "12#K#A123;500000", "07#K#A125;120000",
        "08#K#A124;850000", "07#R#A122;75000",
        "11#K#A123;6200000", "09#L#A120;490000",
        "08#K#A125;500000", "07#K#A125;120000",
        "09#K#A124;850000", "06#R#A122;75000",
        "10#K#A123;6200000", "09#L#A120;390000",
        "10#R#A120;1500000", "07#K#A121;200000",
        "09#L#A122;400000", "06#R#A124;175000",
        "10#K#A121;6200000", "09#L#A123;190000"};
    return arrVerkope;
}
}
```

```
public String [] populateArrAgente(){  
String[] arrAgente = {"A120:Wes Seelig",  
    "A121:Tandra Peters",  
    "A122:Lisha Champ",  
    "A123:Melda Bissonette",  
    "A124:Leta Hansell",  
    "A125:Sid Rebello"};  
return arrAgente;  
}  
}
```

Vraag 3 – GJK-klas ('GUI class')

```

public class VRAAG3_Memo extends javax.swing.JFrame {

//=====
//Given code
//=====

PopulateArrays objAssign = new PopulateArrays();

    String [] arrSales = objAssign.populateArrSales();
    String [] arrAgents = objAssign.populateArrAgents();

public Q3() {
    initComponents();
    this.setLocationRelativeTo(this);
}

//=====
// Nuwe kode
//=====
    String agentKode = "";
    String agentNaam = "";
    String[] arrTipes = {"Kommersieel", "Resedensieel", "Landbou"};
    String[] arrMaande = {"Jan", "Feb", "Mrt", "Apr", "Mei", "Jun", "Jul", "Aug",
"Sep", "Okt", "Nov", "Des"};

    int [][] arrStats = new int [3][5];

    public void clearSales() {
        for (int r = 0; r < 3; r++) {
            for (int c = 0; c < 5; c++) {
                arrStats[r][c] = 0;
            }
        }
    }

//=====
// VRAAG 3.1
//=====
private void btnAgentNaamActionPerformed(java.awt.event.ActionEvent evt) {
    txv3.setText("");
    agentKode = txfAgentKode.getText();
    boolean gevind = false;
    int p = 0;
    while (!gevind && p < arrAgente.length) {
        if (arrAgente[p].substring(0, 4).equals(agentKode)) {
            agentNaam =
                arrAgente[p].substring(arrAgente[p].indexOf(":") + 1);
            lblAgentNaam.setText(agentNaam);
            gevind = true;
        }
        p++;
    }
    if (!gevind) {
        lblAgentNaam.setText("Ongeldige agentkode");
        txfAgentKode.setText("");
    }
}
}

```

```

=====
// VRAAG 3.2.1
=====
Plaas 'n geskikte afvoerkomponent op die GGK (GUI)
Hierdie oplossing maak gebruik van 'n teksarea-komponent
'n Grid kan ook gebruik word met 'n label-komponent om totale verkope te
vertoon.
=====
// VRAAG 3.2.2
=====
private void btnAgentVerkopeActionPerformed(java.awt.event.ActionEvent evt)
{
    btnAgentNaam.doClick();
    initVerkope();
    int totaal = 0;
    for (int i = 0; i < arrVerkope.length; i++) {
        if (arrVerkope[i].substring(5, 9).equals(agentKode)) {
            int maand = Integer.parseInt(arrVerkope[i].substring(0, 2));
            int kol = maand - 1;
            char temp = arrVerkope[i].charAt(3);
            int ry = 0;
            switch (temp) {
                case 'K':
                    ry = 0;
                    break;
                case 'R':
                    ry = 1;
                    break;
                case 'L':
                    ry = 2;
                    break;
            }
            arrAgentVerkope[ry][kol]++;
            arrAgentVerkope[ry][12] += 1;

            totaal += Integer.parseInt(arrVerkope[i].substring
                (arrVerkope[i].indexOf(";") + 1));
        }
    }

    //Vertoon in teksarea
    txaV3.setText(String.format("%-13s", ""));
    for (int i = 0; i < 12; i++) {
        txaV3.append(String.format("%-4s", arrMaande[i]));
    }
    txaV3.append("\n\n");
    for (int r = 0; r < 3; r++) {
        txaV3.append(String.format("%-13s", arrTipes[r]));

        for (int c = 0; c < 13; c++) {
            txaV3.append(String.format("%-4s",
                arrAgentVerkope[r][c]));
        }
        txaV3.append("\n\n");
    }
    txaV3.append("Totale waarde van verkope: R " + totaal);
}

```

BYLAE G: OPLOSSING VIR VRAAG 1: DELPHI

```
unit Vraag1_UMemo;

interface

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

type
  TfrmVraag1 = class(TForm)
    pgcVraag1: TPageControl;
    tsQ1_1: TTabSheet;
    pnlButtons: TPanel;
    bmbClose: TBitBtn;
    tsQ1_3: TTabSheet;
    tsQ1_2: TTabSheet;
    lblMarkWaarde: TLabel;
    lblVerkoopprys: TLabel;
    lblSlaapkamers: TLabel;
    lblBadkamers: TLabel;
    chkSwembad: TCheckBox;
    btnVerkoopsAdvertensie: TButton;
    edtMarkwaarde: TEdit;
    edtVerkoopprys: TEdit;
    edtSlaapkamers: TEdit;
    edtBadkamers: TEdit;
    lblOpskrifKiloW: TLabel;
    lblVorigeLesing: TLabel;
    lblHuidigeLesing: TLabel;
    lblElektrisiteit: TLabel;
    lblGeiserGroottes: TLabel;
    edtVorige: TEdit;
    edtHuidig: TEdit;
    edtGrootte: TEdit;
    lblOpknapArea: TLabel;
    lblVMeter: TLabel;
    redOpknapping: TRichEdit;
    btnBerekenOpknapping: TButton;
    btnVindGeisers: TButton;
    grpOpknappingsArea: TGroupBox;
    grpElektrisiteitGebruik: TGroupBox;
    btnBerekenBedrag: TButton;
    lblBedragVerskuldig: TLabel;
    grpGeiserOpsies: TGroupBox;
    grpAdvertensie: TGroupBox;
    redV1: TRichEdit;
    rgbVerf: TRadioButton;
    rgbTeel: TRadioButton;
    memGeisers: TMemo;
    grpOpknappingsMetode: TGroupBox;
    edtArea: TEdit;
    procedure btnVerkoopsAdvertensieClick(Sender: TObject);
    procedure btnBerekenOpknappingClick(Sender: TObject);
```

```

    procedure btnBerekenBedragClick(Sender: TObject);
    procedure rgbVerfClick(Sender: TObject);
    procedure rgbTeelClick(Sender: TObject);
    procedure btnVindGeisersClick(Sender: TObject);

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

var
    frmVraag1: TfrmVraag1;
    cTipeOpknap : char;
    arrGeisers : array [1..20] of string = ('50-QuickSun50','100-QuickSun100','150-QuickSun150','50-Solar Magic','50-InHotWater','100-SunnyBath 100','150-SunnyBath 150','50-WaterJoy 50','100-WaterJoy 100','150-BigTub 150','50-Small Wonder','100-Medium Wonder','150-Large Wonder','100-SolarWarmth 100','150-SolarWarmth 150','50-Sun Magic','50-Eco Wonder 50','100-Eco Wonder 100','150-Eco Wonder 150','150-Big Earth Saver');

```

Implementation

```

{$R *.dfm}
{$R+}
=====
// Vraag 1.1
=====
procedure TfrmVraag1.btnVerkoopsAdvertensieClick(Sender: TObject);
var
    rMarkWaarde, rVerkoop : real;
    sAdvertensie          : string;
    iBed, iBad            : integer;
begin
    rMarkWaarde := StrToFloat(edtMarkwaarde.Text);
    rVerkoop    := StrToFloat(edtVerkoopprys.Text);
    iBed        := StrToInt(edtSlaapkamers.Text);
    iBad        := StrToInt(edtBadkamers.Text);
    sAdvertensie := 'R' + FloatToStr(rVerkoop) + '#' +
                    IntToStr(iBed) + 'Bed#' + IntToStr(iBad) + 'Bad#';
    if chkSwembad.Checked then
        sAdvertensie := sAdvertensie + 'Swembad#';

    if rVerkoop < rMarkWaarde then
        sAdvertensie := sAdvertensie + 'Winskoop';

    redV1.Lines.Clear;
    redV1.Lines.Add('Huis te koop:' + #13 + sAdvertensie);
end;

```

```
=====
// Vraag 1.2.1
=====
procedure TfrmVraag1.rgbVerfClick(Sender: TObject);
begin
    cTipeOpknap := 'V';
end;
=====
// Vraag 1.2.2
=====
procedure TfrmVraag1.rgbTeelClick(Sender: TObject);
begin
    cTipeOpknap := 'T';
end;
=====
// Vraag 1.2.3
=====
procedure TfrmVraag1.btnBerekenOpknappingClick(Sender: TObject);
var
    rKoste, rLiterVerf, rDromkoste, rTeelKoste : real;
    iOppervlakte, iVolLiter, iEenLiter, iTweeLiter, iVyfLiter : integer;
begin
    // Vraag 1.2.3(a)
    iOppervlakte := StrToInt(edtArea.Text);
    iEenLiter := 0;
    iTweeLiter := 0;
    iVyfLiter := 0;
    rKoste := 0;
    iVolLiter := 0;
    rLiterVerf := 0;
    rDromkoste := 0;
    rTeelKoste := 0;

    if cTipeOpknap = 'V' then
    begin
        rLiterVerf := iOppervlakte / 8;
        iVolLiter := Round(rLiterVerf + 0.4);
        // Alternatief >> Afronding na naaste heelgetal
        // iVolLiter := Round(rLiterVerf);
        // if (iArea mod 8 > 0)
        // then iVolLiter := iVolLiter + 1;

        if (iVolLiter div 5) > 0 then
        begin
            iVyfLiter := iVolLiter div 5;
            dec(iVolLiter, (iVyfLiter * 5));
            rKoste := rKoste + (iVyfLiter * 199.00);
        end;
        if (iVolLiter div 2) > 0 then
        begin
            iTweeLiter := iVolLiter div 2;
            dec(iVolLiter, (iTweeLiter * 2));
            rKoste := rKoste + (iTweeLiter * 92.30);
        end;

        iEenLiter := iVolLiter;
        rKoste := rKoste + (iEenLiter * 55.50);
        redOpknapping.Lines.Clear;
    end;
end;
```

```

        redOpknapping.Lines.Add('Oppervlakte: ' + IntToStr(iOppervlakte) + '
vierkante meter' +
        #13 +
        'Volume verf wat benodig word: ' + FloatToStrF(rLiterVerf,
ffFixed, 8,2) +
        ' liter' + #13 + #13 +
        '1 liter-dromme: ' + #9 + IntToStr(iEenLiter) + #13 +
        '2 liter-dromme: ' + #9 + IntToStr(iTweeLiter) + #13 +
        '5 liter-dromme: ' + #9 + IntToStr(iVyfLiter) + #13 +
        #13 +
        'Totale koste: R ' + FloatToStrF(rKoste, ffFixed, 8, 2));
end; //verf

```

// Vraag 1.2.3(b)

```

if cTipeOpknep = 'T' then
begin
    rTeelKoste := StrToFloat(Inputbox('Teelkoste',
        'Sleutel die koste per vierkante meter in', ''));
    rKoste := (iOppervlakte + 5) * rTeelKoste;
    redOpknapping.Lines.Clear;
    redOpknapping.Lines.Add('Oppervlakte: ' + IntToStr(iOppervlakte) + '
vierkante meter' +
        #13 +
        'Totale koste: R ' + FloatToStrF(rKoste, ffFixed, 8, 2));
end; //teel
end;

```

=====

// Vraag 1.3.1

=====

```

procedure TfrmVraag1.btnBerekenBedragClick(Sender: TObject);
var
    rGebruik, rBedrag      : real;
    iVorigeLesing, iHuidigeLesing : integer;
begin
    // Vraag 1.3.1
    iVorigeLesing := StrToInt(edtVorige.Text);
    iHuidigeLesing := StrToInt(edtHuidig.Text);

    if (iVorigeLesing > iHuidigeLesing) then
    begin
        MessageDlg('Vorige lesing groter as huidige lesing. Sleutel weer in',
            mtError, [mbOk], 0);
        edtHuidig.Clear;
        rBedrag := 0;
    end
    else
    begin
        rGebruik := iHuidigeLesing - iVorigeLesing;
        if rGebruik <= 600 then
            rBedrag := rGebruik
        else
            rBedrag := 600 + (rGebruik - 600) * 1.5;
        end;
        lblElektrisiteit.Caption := FloatToStrF(rBedrag, ffCurrency, 8,2);
    end;
end;

```

```
=====
//   Vraag 1.3.2
=====
```

```
procedure TfrmVraag1.btnVindGeisersClick(Sender: TObject);
```

```
var
```

```
    K : integer;
```

```
    sGrootte, sGeiser : String;
```

```
begin
```

```
    // Vraag 1.3.2
```

```
    memGeisers.Clear;
```

```
    for K := 1 to 20 do
```

```
        begin
```

```
            sGeiser := arrGeisers[K];
```

```
            sGrootte := copy(sGeiser, 1, pos('-', sGeiser)-1);
```

```
            Delete(sGeiser, 1, pos('-', sGeiser));
```

```
            if sGrootte = edtGrootte.Text then
```

```
                memGeisers.Lines.Add(sGeiser);
```

```
        end;
```

```
end;
```

```
//=====
```

```
procedure TfrmVraag1.FormCreate(Sender: TObject);
```

```
begin
```

```
    CurrencyString := 'R ';
```

```
end;
```

```
end.
```

BYLAE H: OPLOSSING VIR VRAAG 2: DELPHI**Eenheidklas: Huislening_U**

```

unit Huislening_UMemo;
interface

uses
  SysUtils, Math, Messages, Dialogs;
=====
// Vraag 2.1.1
=====
Type
  THuislening = class(TObject)
    fAansoekerNaam : string;
    fBesteembareInkomste,
    fLeningBedrag  : real;
    fJare          : integer;
    fRentekoers    : real;
  public
    function berekenPaalement : real;
    constructor Create(pAansoekerNaam : String; pBesteembareInkomste,
                                                              pLeningBedrag : real);

    function GetAansoekerNaam : String;
    procedure SetJare(pJare : integer);
    procedure SetRentekoers(pRentekoers : real);
    function isGoedgekeur : boolean;
    function toString : String;
  end;

implementation

{ THuislening }
=====
// Vraag 2.1.2
=====
constructor THuisLening.Create(pAansoekerNaam: String; pBesteembareInkomste,
  pLeningBedrag: real);
begin
  fAansoekerNaam      := pAansoekerNaam;
  fBesteembareInkomste := pBesteembareInkomste;
  fLeningBedrag       := pLeningBedrag;
  fJare               := 0;
  fRentekoers         := 0;
end;
=====
// Vraag 2.1.3
=====
function THuisLening.GetAansoekerNaam: String;
begin
  result := fAansoekerNaam;
end;
=====
// Vraag 2.1.4
=====
procedure THuisLening.SetJare(pJare: integer);
Kopiereg voorbehou

```

```

begin
    fJare := pJare;
end;

procedure THuisLening.SetRentekoers(pRentekoers: real);
begin
    fRentekoers := pRentekoers;
end;

=====
// Vraag 2.1.5 Gegewe kode (Verwyder kommentaartekens)
=====
function THuisLening.berekenPaalement: real;
var
    rBedrag, rKoers : real;
begin
    rKoers := fRenteKoers / 100;
    rBedrag := (rKoers * fLeningBedrag)/(1-Power((1+rKoers),-fJare));
    Result := Round(rBedrag / 12);
end;

=====
// Vraag 2.1.6
=====
function THuisLening.isGoedgekeur: boolean;
var
    bGeldig : boolean;
begin
    bGeldig := True;

    if (fLeningBedrag > 800000)
    then bGeldig := (fJare > 20);

    if (fLeningBedrag <= 600000)
    then bGeldig := fBesteembareInkomste > (berekenPaalement * 1.3);

    Result := bGeldig;
end;
end.

=====
// Vraag 2.1.7
=====
function THuisLening.toString: String;
var
    sAfvoer : String;
begin
    sAfvoer := 'Naam van aansoeker: ' + fAansoekerNaam + #13;
    sAfvoer := sAfvoer + 'Besteembare inkomste: ' +
        FloatToStrF(fBesteembareInkomste, ffCurrency, 8, 2) + #13;
    sAfvoer := sAfvoer + 'Leningbedrag: ' + FloatToStrF(fLeningBedrag,
        ffCurrency, 8, 2) + #13;
    sAfvoer := sAfvoer + 'Getal jaar: ' + IntToStr(fJare) + #13;
    sAfvoer := sAfvoer + 'Rentekoers: ' + FloatToStrF(fRentekoers, ffFixed,
        8, 1) + '%' + #13;
    sAfvoer := sAfvoer + 'Lening GOEDGEKEUR met ' + 'n maandelikse paalement van '
        + FloatToStrF(berekenPaalement, ffCurrency, 8, 2);
    Result := sAfvoer;
end;

end.
Kopiereg voorbehou

```

Vormklas: Vraag2_P

```
interface
```

```
uses
```

```
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
  Forms,  
  Dialogs, StdCtrls, ComCtrls, Buttons, ExtCtrls, Math;
```

```
type
```

```
  TfrmVraag2 = class(TForm)  
    grpLenings: TGroupBox;  
    redV2: TRichEdit;  
    cboAansoekers: TComboBox;  
    pnlVerlaat: TPanel;  
    bmbVerlaat: TBitBtn;  
    lblRentekoers: TLabel;  
    edtRenteKoers: TEdit;  
    btnVindAansoeker: TButton;  
    btnEvalueer: TButton;  
    grpAansoekStatus: TGroupBox;  
    grpAansoekers: TGroupBox;  
    cboJare: TComboBox;  
    lblJare: TLabel;  
    procedure FormCreate(Sender: TObject);  
    procedure btnEvalueerClick(Sender: TObject);  
    procedure btnVindAansoekerClick(Sender: TObject);  
  private  
    { Private declarations }  
  public  
    { Public declarations }  
  end;
```

```
var
```

```
  frmVraag2: TfrmVraag2;
```

```
implementation
```

```
uses
```

```
  HuisLening_UMemo;
```

```
const
```

```
  iAantalJare = 25;
```

```
var
```

```
  LeningObj : THuisLening;
```

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

```
{ $R+ }
```

```

=====
// Vraag 2.2.1
=====

procedure TfrmVraag2.btnVindAansoekerClick(Sender: TObject);
var
  sAansoekerNaam, sLeerNaam: String;
  rInkomste, rUitgawes, rLening, rBesteeInkomste: real;
  TLeer : Textfile;
  sNaam : String;
begin
  if cboAansoekers.ItemIndex = -1
  then
    begin
      MessageDlg('Kies ' 'n aansoeker', mtError, [mbOk], 0);
      Exit;
    end
  else
    begin
      sAansoekerNaam := cboAansoekers.Items[cboAansoekers.ItemIndex];
      sLeernaam := sAansoekerNaam + '.txt';
    end;

    redV2.Lines.Clear;
    if NOT FileExists(sLeerNaam)
    then
      begin
        redV2.Lines.Add('Geen huisleningaansoek is deur ' + sAansoekerNaam
+ ' ingedien nie.');
```

btnEvalueer.Enabled := False;

Exit;

end;

AssignFile(TLeer, sLeernaam);

Reset(TLeer);

Readln(TLeer, sNaam);

Readln(TLeer, rInkomste);

Readln(TLeer, rUitgawes);

Readln(TLeer, rLening);

CloseFile(TLeer);

redV2.Lines.Add(' 'n Huisleningaansoek is ingedien deur ' +

sAansoekerNaam);

rBesteeInkomste := rInkomste - rUitgawes;

LeningObj := THuisLening.Create(sAansoekerNaam, rBesteeInkomste,

rLening);

MessageDlg('Huisleningobjek is suksesvol geskep', mtInformation,

[mbOk], 0);

btnEvalueer.Enabled := true;

end;

```

=====
// Vraag 2.2.2
=====

procedure TfrmVraag2.btnEvalueerClick(Sender: TObject);
var
  rRentekoers : real;
  iJare       : integer;
begin

```

```
// Vraag 2.2.2
iJare := StrToInt(cboJare.Items[cboJare.ItemIndex]);
rRentekoers := StrToFloat(edtRenteKoers.Text);

LeningObj.SetJare(iJare);
LeningObj.SetRentekoers(rRentekoers);

redV2.Lines.Clear;
if LeningObj.isGoedgekeur
then
begin
    redV2.Lines.Add(LeningObj.toString + #13 +
        'Lening GOEDGEKEUR met ' 'n maandelikse paaient van R ' +
        FloatToStrF(LeningObj.berekenPaaient, ffFixed, 8, 2));
end
else
begin
    redV2.Lines.Add('Naam van aansoeker: ' + LeningObj.GetAansoekerNaam +
#13      + 'Lening NIE GOEDGEKEUR NIE');
end;
end;
//=====
procedure TfrmVraag2.FormCreate(Sender: TObject);
begin
    CurrencyString := 'R ';
    //  cboAansoekers.ItemIndex := 0;
end;

end.
```

BYLAE I: OPLOSSING VIR VRAAG 3: DELPHI

```

unit Vraag3_UMemo;
//Moontlike oplossing vir Vraag 3
interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, ComCtrls, Buttons, Grids;

type
  TfrmVraag3 = class(TForm)
    grpAgentBesonderhede: TGroupBox;
    lblAgentNaam: TLabel;
    edtKode: TEdit;
    btnAgentNaam: TButton;
    btnAgentVerkoopRekord: TButton;
    grpVertoon: TGroupBox;
    lblAgentKode: TLabel;
    bmbVerlaat: TBitBtn;
    PageControl1: TPageControl;
    tabREdit: TTabSheet;
    tabGrid: TTabSheet;
    redV3: TRichEdit;
    sgrdV3: TStringGrid;
    lblTotaal: TLabel;
    procedure btnAgentNaamClick(Sender: TObject);
    procedure btnAgentVerkoopRekordClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag3: TfrmVraag3;

arrVerkope : array[1..132] of string = ('01#K#A122;500000',
  '01#K#A120;1200000', '01#K#A123;850000', '01#R#A121;75000',
  '02#K#A123;6200000', '02#L#A121;190000', '02#K#A121;500000',
  '02#K#A120;1200000', '02#K#A120;850000', '02#R#A124;75000',
  '02#K#A123;620000', '02#L#A124;690000', '02#K#A121;500000',
  '02#K#A120;1200000', '02#K#A120;850000', '02#R#A124;75000',
  '02#K#A123;620000', '02#L#A124;790000', '02#R#A124;1500000',
  '02#K#A125;200000', '02#L#A121;400000', '03#R#A120;175000',
  '03#K#A123;6200000', '03#L#A123;190000', '03#K#A124;500000',
  '03#K#A125;1200000', '03#K#A123;850000', '03#R#A122;75000',
  '03#K#A123;620000', '03#L#A123;890000', '03#R#A123;1500000',
  '03#K#A125;200000', '03#L#A123;400000', '04#R#A124;175000',
  '04#K#A120;6200000', '04#L#A123;190000', '03#R#A124;1500000',
  '03#K#A125;200000', '03#L#A122;400000', '04#R#A121;175000',
  '04#K#A123;6200000', '04#L#A123;190000', '04#K#A121;500000',
  '04#K#A120;1200000', '04#K#A120;850000', '04#R#A124;75000',
  '04#K#A123;620000', '04#L#A123;590000', '04#K#A123;500000',
  '04#K#A125;120000', '04#K#A124;850000', '04#R#A122;75000',
  '04#K#A123;6200000', '04#L#A120;490000', '04#K#A125;500000',

```

```
'04#K#A125;120000', '04#K#A124;850000', '04#R#A122;75000',
'04#K#A123;6200000', '04#L#A120;390000', '04#R#A120;1500000',
'04#K#A121;200000', '04#L#A122;400000', '04#R#A124;175000',
'04#K#A121;6200000', '04#L#A123;190000', '11#K#A122;500000',
'07#K#A120;1200000', '10#K#A123;850000', '12#R#A121;75000',
'12#K#A123;6200000', '11#L#A121;190000', '12#K#A121;500000',
'12#K#A120;1200000', '11#K#A120;850000', '12#R#A124;75000',
'12#K#A123;620000', '11#L#A124;690000', '09#K#A121;500000',
'08#K#A120;1200000', '11#K#A120;850000', '12#R#A124;75000',
'12#K#A123;620000', '10#L#A124;790000', '08#R#A124;1500000',
'08#K#A125;200000', '11#L#A121;400000', '11#R#A120;175000',
'07#K#A123;6200000', '09#L#A123;190000', '07#K#A124;500000',
'07#K#A125;1200000', '12#K#A123;850000', '10#R#A122;75000',
'08#K#A123;620000', '08#L#A123;890000', '06#R#A123;1500000',
'09#K#A125;200000', '09#L#A123;400000', '09#R#A124;175000',
'08#K#A120;6200000', '07#L#A123;190000', '08#R#A124;1500000',
'06#K#A125;200000', '07#L#A122;400000', '08#R#A121;175000',
'08#K#A123;6200000', '07#L#A123;190000', '11#K#A121;500000',
'06#K#A120;1200000', '08#K#A120;850000', '08#R#A124;75000',
'12#K#A123;620000', '07#L#A123;590000', '12#K#A123;500000',
'07#K#A125;120000', '08#K#A124;850000', '07#R#A122;75000',
'11#K#A123;6200000', '09#L#A120;490000', '08#K#A125;500000',
'07#K#A125;120000', '09#K#A124;850000', '06#R#A122;75000',
'10#K#A123;6200000', '09#L#A120;390000', '10#R#A120;1500000',
'07#K#A121;200000', '09#L#A122;400000', '06#R#A124;175000',
'10#K#A121;6200000', '09#L#A123;190000');
```

```
arrAgente:array[1..6] of string =('A120:Wes Seelig',
'A121:Tandra Peters',
'A122:Lisha Champ',
'A123:Melda Bissonette',
'A124:Leta Hansell',
'A125:Sid Rebello');
```

```
implementation
```

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

```
{ $R+ }
```

```
var
```

```
    sAgentKode : String = '';
```

```
    sAgentNaam : String = '';
```

```
=====
// Vraag 3.1
```

```
=====
procedure TfrmVraag3.btnAgentNaamClick(Sender: TObject);
```

```
var
```

```
    K          : integer;
```

```
    bGevind    : boolean;
```

```
begin
```

```
    // Vraag 3.1
```

```
    sAgentKode := edtKode.Text;
```

```
    bGevind := false;
```

```
    K := 1;
```

```
    while (K <= Length(arrAgente)) AND NOT bGevind do
```

```
    begin
```

```
        if pos(sAgentKode, arrAgente[K]) = 1 then
```

```
        begin
```

```

        sAgentNaam := copy(arrAgente[K], pos(':', arrAgente[K])+1,
length(arrAgente[K]));
        lblAgentNaam.Caption := sAgentNaam;
        bGevind := True;
    end
else
    inc(K, 1);
end; //while
if NOT bGevind then
    begin
        edtKode.Clear;
        lblAgentNaam.Caption := 'Ongeldige agentkode';
    end;
end;

```

```

=====
// Vraag 3.2.1
=====

```

Plaas 'n geskikte afvoerkomponent op die GGK (GUI)
 Hierdie oplossing maak gebruik van 'n teksarea-komponent
 'n Grid kan ook gebruik word met 'n label-komponent om totale verkope te
 vertoon.

```

=====
// Vraag 3.2.2
=====

```

```

procedure TfrmVraag3.btnAgentVerkoopRekordClick(Sender: TObject);

```

```

var

```

```

    arrAgentVerkope : array[1..3, 1..13] of integer;
    iIndex, iRy, iKol, iPrys, iTotaal, iRyTotaal : integer;
    sTemp : String;

```

```

begin

```

```

    // Vraag 3.2

```

```

    //inisialiseer 2D-skikking

```

```

    for iRy := 1 to 3 do

```

```

        for iKol := 1 to 13 do

```

```

            arrAgentVerkope[iRy, iKol] := 0;

```

```

        //analiseer verkope-skikking

```

```

        iTotaal := 0;

```

```

        for iIndex := 1 to length(arrVerkope) do

```

```

            begin

```

```

                sTemp := arrVerkope[iIndex];

```

```

                if pos(sAgentKode, sTemp) > 0 then

```

```

                    begin

```

```

                        iKol := StrToInt(Copy(sTemp, 1, pos('#', sTemp)-1));

```

```

                        iPrys := StrToInt(Copy(sTemp, pos(';', sTemp)+1, length(sTemp)));

```

```

                        inc(iTotaal, iPrys);

```

```

                        Delete(sTemp, 1, pos('#', sTemp));

```

```

                        case sTemp[1] of //tipe verkope

```

```

                            'K' : iRy := 1;

```

```

                            'R' : iRy := 2;

```

```

                            'L' : iRy := 3;

```

```

                        end; //case

```

```

                        inc(arrAgentVerkope[iRy, iKol], 1);

```

```

                    end; //if

```

```

                end; //for

```

```

            //bepaal ry-totale

```

```

            for iRy := 1 to 3 do

```

```

Kopiereg voorbehou

```

Blaai om asseblief

```

begin
    iRyTotaal := 0;
    for iKol := 1 to 12 do
        begin
            inc(iRyTotaal, arrAgentVerkope[iRy, iKol]);
        end;
    arrAgentVerkope[iRy, 13] := iRyTotaal;
end;

//Afvoer: RichEdit
redv3.Lines.Clear;
redV3.Paragraph.TabCount := 1;
redV3.Paragraph.Tab[0] := 70;
sTemp := ' ' + #9 + 'Jan' + #9 + 'Feb' + #9 + 'Mrt' + #9 + 'Apr' + #9 +
'Mei' + #9 +
        'Jun' + #9 + 'Jul' + #9 + 'Aug' + #9 + 'Sep' + #9 + 'Okt' + #9 +
        'Nov' + #9 + 'Des';
redV3.Lines.Add(sTemp);
redV3.Lines.Add(' ');
sTemp := 'Kommersieel' + #9;
for iKol := 1 to 13 do
    sTemp := sTemp + IntToStr(arrAgentVerkope[1, iKol]) + #9 ;
redV3.Lines.Add(sTemp);

redV3.Lines.Add(' ');
sTemp := 'Residensieel' + #9;
for iKol := 1 to 13 do
    sTemp := sTemp + IntToStr(arrAgentVerkope[2, iKol]) + #9 ;
redV3.Lines.Add(sTemp);

redV3.Lines.Add(' ');
sTemp := 'Landbou' + #9;
for iKol := 1 to 13 do
    sTemp := sTemp + IntToStr(arrAgentVerkope[3, iKol]) + #9 ;
redV3.Lines.Add(sTemp);

redV3.Lines.Add(' ');
redV3.Lines.Add('Totale waarde van verkope: R ' + IntToStr(iTotaal));

//Afvoer: StringGrid
//Kolomopskrifte
{ sgrdV3.Cells[1,0] := 'Jan';
  sgrdV3.Cells[2,0] := 'Feb';
  sgrdV3.Cells[3,0] := 'Mrt';
  sgrdV3.Cells[4,0] := 'Apr';
  sgrdV3.Cells[5,0] := 'Mei';
  sgrdV3.Cells[6,0] := 'Jun';
  sgrdV3.Cells[7,0] := 'Jul';
  sgrdV3.Cells[8,0] := 'Aug';
  sgrdV3.Cells[9,0] := 'Sep';
  sgrdV3.Cells[10,0] := 'Okt';
  sgrdV3.Cells[11,0] := 'Nov';
  sgrdV3.Cells[12,0] := 'Des';
  //Ryopskrifte.
  sgrdV3.Cells[0,1] := 'Kommersieel';
  sgrdV3.Cells[0,2] := 'Residensieel';
  sgrdV3.Cells[0,3] := 'Landbou';

```

```
sgrdV3.ColWidths[0] := 100;

for iRy := 1 to 3 do
  for iKol := 1 to 13 do
    sgrdV3.Cells[iKol, iRy] := IntToStr(arrAgentVerkope[iRy, iKol]);

  lblTotaal.Caption := 'Totale waarde van verkope: R' + IntToStr(iTotaal); }
end;

end.
```