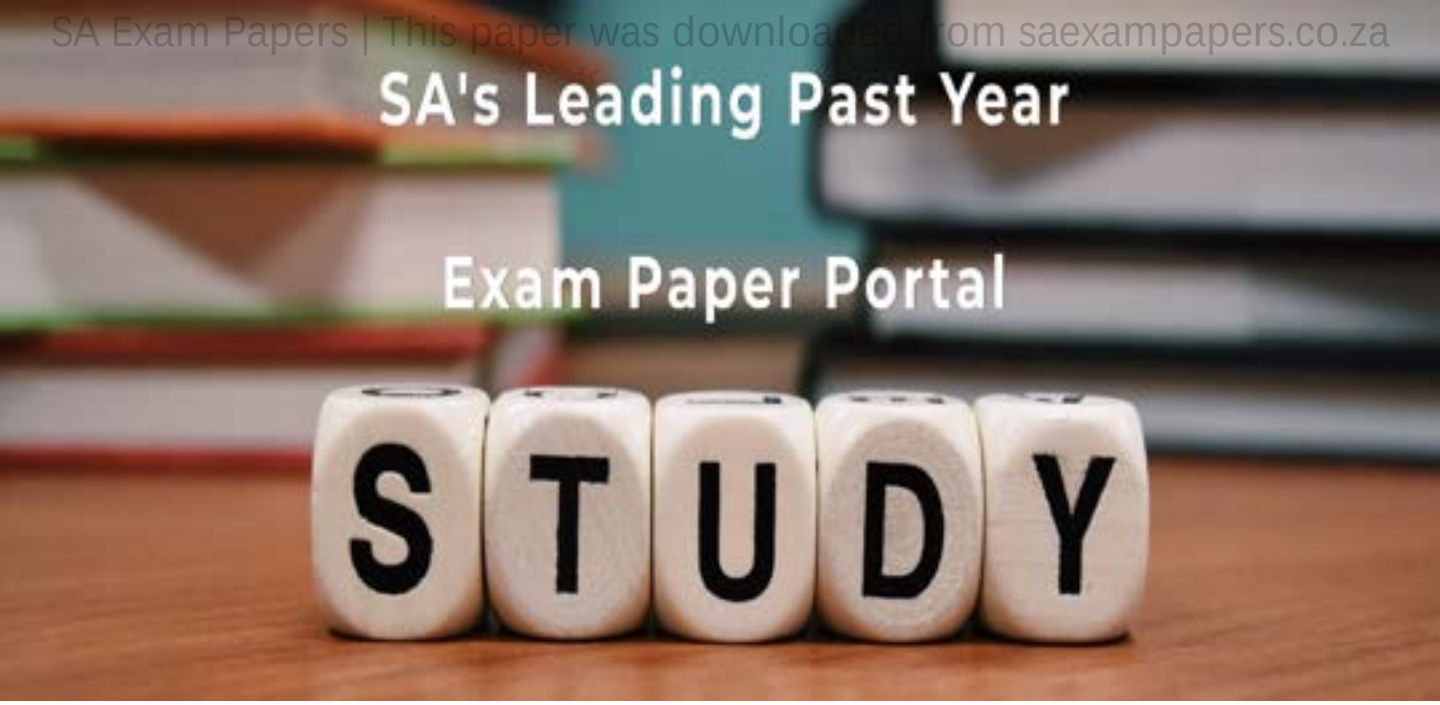

You have Downloaded, yet Another Great Resource to assist you with your Studies ☺

Thank You for Supporting SA Exam Papers

Your Leading Past Year Exam Paper Resource Portal

Visit us @ www.saexampapers.co.za

Province of the
EASTERN CAPE
EDUCATION

NASIONALE SENIORSERTIFIKAAT

GRAAD 12

SEPTEMBER 2023

INLIGTINGSTEGNOLOGIE V1
NASIENRIGLYN

PUNTE: **150**

Hierdie nasienriglyn bestaan uit 20 bladsye.

NAAM VAN LEERDER:				
TOTAAL VRAAG 1	**TOTAAL VRAAG 2**	**TOTAAL VRAAG 3**	**TOTAAL VRAAG 4**	**TOTAAL**
/40	**/35**	**/35**	**/40**	**/150**

VRAAG 1		MAKS PUNTE	PUNTE BEHAAL
1.1	**KNOPPIE: [V1.1 – Bepaal koste]** Gebruik 'n if / case stelling ✓ Bepaal die lisensietipe uit die radio group ✓ Vertoon die lisensietipe ✓ asook die koste van die lisensie in die Koste-label komponent ✓	**4**	
1.2.1	**KNOPPIE: [V1.2.1 – Voeg voorste wiel by]** Skep dinamiese vorm: shpVoorsteWiel ✓ Stel 'parent' van shpFrontWheel na die paneel pnlV1_2_Ligte ✓ Verander die eienskappe van shpVoorsteWiel: Type → Circle ✓ Top → 25 ✓ Left → 35 Width → 50 ✓ Height → 50 Colour → White ✓	**6**	
1.2.2	**KNOPPIE: [V1.2.2 – Lig landingsrat op]** Verander die kleur van al drie vorme na Rooi ✓	**1**	
1.2.3	**KNOPPIE: [V1.2.3 – Sit landingsrat neer]** Verander die kleur van al drie vorme na Groen ✓	**1**	
1.2.4	**KNOPPIE: [V1.2.4 – Toets landingsrat]** Gebruik 'n case of if statement, bepaal die toestand van die ligte en skep die korrekte boodskap met 'n Message DLG As al 3 ligte = Rooi ✓ → MessageDLG ✓ ('Katastrofiese fout') ✓ Else ✓ As al 3 ligte = Groen ✓ → MessageDLG('Veilig om te land') ✓ Else ✓ MessageDLG('Waarskuwing, nie veilig om te land nie') ✓	**8**	

1.3	**KNOPPIE: [V1.3 Hoogtepunt van Daling]** **Vertikale Spoed** Kry BeginSpoed Kry EindSpoed ✓ Kry Wind Kry Helling GemiddeldeSpoed = (BeginSpoed + EindSpoed) / 2 + Wind ✓✓ VertikaleSpoed_NM_P_Min = (tan(Helling * DEG_TO_RAD) * GemiddeldeSpoed) / 60 ✓✓ VertikaleSpoed := VertikaleSpoed_NM_P_Min * NM_TO_FT ✓ **Afstand** Kry BeginHoogte ✓ Kry EindHoogte DeltaHoogte = BeginHoogte – EindHoogte ✓ Afstand = (DeltaHoogte * FT_TO_NM) / tan(Helling * DEG_TO_RAD); ✓✓ DeltaSpoed = BeginSpoed - EindSpoed; ✓ Afstand = Afstand + ceil(DeltaSpoed / 10) ✓ Afstand = Afstand + ceil(Windspoed / 10) ✓ **Tyd** Geskatte tyd = (Afstand / GemiddeldeSpoed) * 60 ✓ Maak rich edit skoon Voeg by rich edit: Vertikale Spoed: VertikaleSpoed (fpm) – *geformateer na 2 desimale* ✓✓ Afstand: Afstand (nm) – *geformateer na 2 desimale* ✓✓ Geskatte Tyd: Geskatte tyd (min) – *geformateer na 2 desimale* ✓✓	**20**	
	Vraag 1 Totaal	**40**	

VRAAG 2		MAKS PUNTE	PUNTE BEHAAL
2.1.1	**Knoppie [V2.1.1]** 'SELECT Naam, Van, Ouderdom, Epos FROM tblVlieeniers WHERE Ouderdom > 35 AND Ouderdom < 43 ORDER BY Ouderdom DESC' SELECT vier korrekte velde ✓ FROM korrekte tabel ✓ WHERE Ouderdom in die reeks van 36 (insl) en 42 (insl) ✓ ORDER BY korrekte veld DESC ✓	**4**	
2.1.2	**Knoppie [V2.1.2]** 'SELECT * FROM tblVlugte WHERE Bestemming LIKE' + QuotedStr('%' + sLine + '%') SELECT * (alle velde) ✓ FROM korrekte tabel ✓ WHERE Bestemming LIKE ✓ QuotedStr('%' + sLine + '%') ✓	**4**	
2.1.3	**Knoppie [V2.1.3]** 'SELECT count(*) AS [Vlugte in September] FROM tblVlugte WHERE Month(VertrekDatum) = 9' SELECT count(*) ✓AS [Vlugte in September] FROM tblVlugte WHERE Month ✓ (VertrekDatum) = 9 ✓	**3**	
2.1.4	**Knoppie [V2.1.4]** 'SELECT Bestemming, Format(sum(VlieenierKostePerVlug), "Currency") AS [Vlieenierkoste], Format(sum(Vlugkoste), "Currency") AS [Vlugkoste], Format(sum(VlieenierKostePerVlug) + sum(VlugKoste), "Currency") AS [Totale Koste] FROM tblVlieeniers, tblVlugte WHERE tblVlieeniers.VlieenierID = tblVlugte.VlieenierID GROUP BY Bestemming' SELECT Bestemming ✓ , Format ✓ (sum ✓ (VlieenierKostePerVlug) ✓, "Currency"✓) AS [Vlieenierkoste] ✓, Format(sum(Vlugkoste), "Currency") AS [Vlugkoste] ✓, Format((sum(VlieenierKostePerVlug) ✓ + sum(Vlugkoste))✓, "Currency") AS [Totale Koste] FROM albei tabelle ✓ (tblVlieeniers, tblVlugte) WHERE verbinding tussen tabelle ✓ (tblVlieeniers.VlieenierID = tblVlugte.VlieenierID) GROUP BY Bestemming ✓	**12**	
2.1.5	**Knoppie [V2.1.5]** 'UPDATE tblVlieeniers SET VlieenierKostePerVlug = VlieenierKostePerVlug * 1.07 WHERE LisensieTipe = "KVL"' UPDATE korrekte tabel ✓ SET VlieenierKostePerVlug = VlieenierKostePerVlug * 1.07 ✓ WHERE LisensieTipe = "KVL" ✓	**3**	

2.2.1	**Knoppie [V2.2.1]** if FieldByName('Ouderdom').AsInteger < 21 then ✓ delete ✓ else next; ✓	**3**	

2.2.2	**Knoppie [V2.2.2]**
	begin
	if Bestemming = sBestemming then ✓
	begin
	if LisensieVereiste = 'KVL' then
	inc(iKVL)
	else
	if LisensieVereiste = 'PVL' then ✓
	inc(iPVL)
	else
	if LisensieVereiste = 'MVL' then
	inc(iMVL);
	if MedeVlieenier = True then ✓
	inc(imedevlieenier)
	end ✓ *NB: Medevlieenier moet binne die begin en end wees, anders tel dit op I die medevlieeniers vir al die bestemmings.*
	Next ✓
	end
	Afvoer ('KVL: ' + IntToStr(iKVL) + #13 +
	'PVL: ' + IntToStr(iPVL) + #13 +
	'MVL: ' + IntToStr(iMVL) + #13 + ✓
	'Medevlieenier nodig: ' + IntToStr(imedevlieenier));
		6	

	Vraag 2 Totaal	**35**	

VRAAG 3		MAKS PUNTE	PUNTE BEHAAL
3.1.1	**Konstruktor Create** Korrekte opskrif en parameters ✓✓ fNaam := sNaam fVervaardiger := sVervaadiger fSpoed := rSpoed fHoogte := rHoogte fAfstand := rAfstand fGewig := rGewig fVlerkspan := rVlerkspan fVuurkrag := iVuurkrag fPrentjieNaam := sPrentjieNaam fLand := sLand fBeskrywing := sBeskrywing Korrekte toekenning van al die attribute ✓✓✓	**5**	
3.1.2	**Mutator Metode – setWaardes** Korrekte opskrif (prosedure setWaardes) ✓ Korrekte omskakeling van elke attribuut ✓✓✓✓	**5**	

3.2.1	**OnChange gebeurtenis van cmbV3_KiesVliegtuig**
	1. Kry die gebruikers se keuse uit die combo box ✓
	2. Toets om te sien of tekslêer bestaan en ken lêer toe. As die lêer nie bestaan nie, vertoon 'n boodskap en exit
	AssignFile(MyFile, 'Vliegtuig_Lys.csv'); ✓
	Try ✓
	Reset(MyFile); ✓
	except
	ShowMessage('lêer nie gevind nie'); ✓
	Exit; ✓
	end; *Of alternatief: if not (fileexists(textfile) = true) then*
	3. Lus deur die tekslêer totdat die vliegtuig gevind is.
	bFound := False; ✓
	while (not eof(MyFile)) ✓ AND (bFound = False) ✓ do
	if pos(UpperCase(sSearch),UpperCase(sOneline)) ✓ <> 0 ✓ then
	bFound := True; ✓
	4. As die vliegtuig gevind is, dan:
	Lus en onttrek die inligting van die tekslêer
	//Vliegtuignaam
	iPos := pos(',',sOneLine); ✓
	sNaam := copy(sOneLine,1,iPos-1); ✓
	delete(sOneLine,1,iPos); ✓
	//Ander velde ✓✓
	Instansieer (skep) die objek **objVliegtuig**.
	objVliegtuig := TVliegtuig.Create ✓(sNaam, sVervaardiger,
	rSpoed, rHoogte, rAfstand, rGewig, rVlerkspan,
	iVuurkrag, sPrentjieNaam, sLand, sBeskrywing); ✓✓
	5. Roep die setWaarde-metode ✓
	6. Laai die objekdata in die komponente ✓✓✓✓
		25	

	Vraag 3 Totaal	**35**	

VRAAG 4		MAKS PUNTE	PUNTE BEHAAL
4.1	Kry die bestemming uit die combo box ✓ Ken die bestemmingskikking aan die ar2Bespreking toe ✓✓✓✓✓	**6**	
4.2	1. Kry die sitpleknommer ✓✓ Lus deur die skikking en bepaal of die sitplek bespreek is ✓✓ Vertoon 'n boodskap as die sitplek reeds bespreek is ✓ 2. Bespreek die sitplek in ar2Bespreking ✓ Kry al die data ✓ Bepaal klas ✓ Bepaal prys ✓ If Besigheidsklas * 1.95 ✓ Vertoon opskrif Besprekinginligting ✓ in bold ✓ Vertoon besprekingskaartjie-inligting geformateer: 'Naam en Van: ' + #13#9 + sNaamVan + #13 + ✓ 'Bestemming: ' + #13#9 + sBestemming + #13 + 'Datum en Tyd: ' + #13#9 + sDatum + #13 + #9 + sTyd + ' vlug' + #13 + 'Klas: ' + #13#9 + sKlas + #13 + 'Sitpleknommer: ' + #13#9 + cCol ✓+ IntToStr(iRow + 1)✓+ #13 + 'Koste: ' + #13#9 + FloatToStrF✓ (rPrice,ffCurrency✓,10,2);	**17**	
4.3	1. Kry die bestemming ✓ 2. Lus deur ar2Bespreking en vermeerder die aantal passasiers: Besigheidsklas Lus deur Rye 0 to 1 ✓ Lus deur kolomme 0 to 4 ✓ if ar2Bespreking[Row,Col] = 'B' then ✓ vermeerder(besigheidsklas teller) ✓ en Ekonomiese Klas Lus deur rye 2 tot 14 Lus deur kolomme 0 tot 4 } ✓ if ar2Bespreking[Row,Col] = 'B' then ✓ vermeerder(ekonomiese klas teller) 3. Bepaal koste vir besigheidsklas ✓✓ en ekonomiese klas ✓✓ 4. Vertoon besprekinskaartjie-inligting geformateer 'Passasiers' + #13 + #9 + 'Besigheidsklas: ' + IntToStr(iBusClass) + #13 + ✓ #9 + 'Ekonomiese klas: ' + IntToStr(iEcoClass) + #13 + ✓ #9 + 'Totaal: ' + IntToStr(iBusClass + iEcoClass) + #13#13 + ✓ 'Koste' + #13 + #9 + 'Besigheidsklas: ' + FloatToStrF(rBusPrice,ffCurrency,10,2) + #13 + ✓ #9 + 'Ekonomiese klas: ' + FloatToStrF(rEcoPrice,ffCurrency,10,2) + #13 +✓ #9 + 'Totale Koste: ' + FloatToStrF(rBusPrice + rEcoPrice,ffCurrency,10,2);✓	**17**	
	Vraag 4 Totaal	**40**	

VOORBEELDE EN OPLOSSINGS

VRAAG 1

```
//////////     40 punte     //////////

//////////     Vraag 1.1  –  4 punte     //////////
procedure TfrmQuestion1.btnQ1_1_CostClick(Sender: TObject);
begin
  case rgpQ1_1_License.ItemIndex of
    0 : lblQ1_1_Cost.Caption := 'Microlight Pilot License = R37 000';
    1 : lblQ1_1_Cost.Caption := 'Private Pilot License = R110 451';
    2 : lblQ1_1_Cost.Caption := 'Commercial Pilot License = R761 379';
  end;
end;


//////////     Vraag 1.2.1  –  6 punte     //////////
procedure TfrmQuestion1.btnQ1_2_1Click(Sender: TObject);
begin
  shpFrontWheel := TShape.Create(frmQuestion1);
  shpFrontWheel.Parent := pnlQ1_2_Lights;

  with shpFrontWheel do
    begin
      Shape := stCircle;
      Top := 25;
      Left := 35;
      Height := 50;
      Width := 50;
      Brush.Color := clWhite;
    end;
end;


//////////     Vraag 1.2.2  –  1 punt     //////////
procedure TfrmQuestion1.btnQ1_2_2_UpClick(Sender: TObject);
begin
  btnQ1_2_1.Click; //Provided code, DO NOT DELETE
  shpFrontWheel.Brush.Color := clRed;
  shpLeftWheel.Brush.Color  := clRed;
  shpRightWheel.Brush.Color := clRed;
end;


//////////     Vraag 1.2.3  –  1 Punt     //////////
procedure TfrmQuestion1.btnQ1_2_3_DownClick(Sender: TObject);
begin
  btnQ1_2_1.Click; //Provided code, DO NOT DELETE
  shpFrontWheel.Brush.Color := clGreen;
  shpLeftWheel.Brush.Color  := clGreen;
  shpRightWheel.Brush.Color := clGreen;
end;
```

```
//////////     Vraag 1.2.4  –  8 punte     //////////
procedure TfrmQuestion1.btnQ1_2_4_TestClick(Sender: TObject);
begin
  btnQ1_2_1.Click;   //Provided code, DO NOT DELETE
  RandomColours;    //Provided code, DO NOT DELETE

  if (shpFrontWheel.Brush.Color = clRed) AND
     (shpLeftWheel.Brush.Color = clRed) AND
     (shpRightWheel.Brush.Color = clRed) then
        MessageDLG('Catastrophic failure!',MTError,[MBOk],0)
  else
  if (shpFrontWheel.Brush.Color = clGreen) AND
     (shpLeftWheel.Brush.Color = clGreen) AND
     (shpRightWheel.Brush.Color = clGreen) then
        MessageDLG('Safe to land',MTInformation,[MBOk],0)
  else
     MessageDLG('Caution, not safe to land',MTInformation,[MBOk],0);

ALTERNATIEWE OPLOSSING
var
  iFront, iLeft, iRight, iTotal : Integer;

  iFront := 0;
  iLeft  := 0;
  iRight := 0;
  iTotal := 0;
  if shpFrontWheel.Brush.Color = clGreen then
    iFront := 1;
  if shpLeftWheel.Brush.Color = clGreen then
    iLeft := 1;
  if shpRightWheel.Brush.Color = clGreen then
    iRight := 1;

  iTotal := iFront + iLeft + iRight;
  case iTotal of
    0      : MessageDLG('Catastrophic failure!',MTError,[MBOk],0);
    1,2    : MessageDLG('Caution, not safe to land',MTInformation,[MBOk],0);
    3      : MessageDLG('Safe to land',MTInformation,[MBOk],0);
  end;
end;
```

```
//////////    Vraag 1.3  -   20 Punte     //////////
procedure TfrmQuestion1.Q1_3_TopOfDescentClick(Sender: TObject);

const DEG_TO_RAD = 0.0174532925;
const NM_TO_FT = 6076.11549;
const FT_TO_NM = 1 / NM_TO_FT;

var
  rAverageSpeed, rStartSpeed, rEndSpeed, rVerticalSpeed, rWind   : Real;
  rGlideSlope, rVerticalSpeed_NM_P_Min                           : Real;
  rStartAltitude, rEndAltitude                                   : Real;
  rDistance, rDeltaAltitude, rDeltaSpeed                         : Real;
  rEstimatedTime                                                 : Real;

begin
//Vertical Speed
  rStartSpeed := StrToFloat(edtQ1_3_StartSpeed.Text);
  rEndSpeed := StrToFloat(edtQ1_3_EndSpeed.Text);
  rWind := StrToFloat(edtQ1_3_Wind.Text);
  rGlideSlope := StrToFloat(edtQ1_3_GlideSlope.Text);

  rAverageSpeed := (rStartSpeed + rEndSpeed) / 2 + rWind;
  rVerticalSpeed_NM_P_Min := (tan(rGlideSlope * DEG_TO_RAD) * rAverageSpeed) / 60;
  rVerticalSpeed := rVerticalSpeed_NM_P_Min * NM_TO_FT;

//Distance
  rStartAltitude := StrToFloat(edtQ1_3_StartAltitude.Text);
  rEndAltitude := StrToFloat(edtQ1_3_EndAltitude.Text);

  rDeltaAltitude := rStartAltitude - rEndAltitude;
  rDistance := (rDeltaAltitude * FT_TO_NM) / tan(rGlideSlope * DEG_TO_RAD);
  rDeltaSpeed := rStartSpeed - rEndSpeed;
  rDistance := rDistance + ceil(rDeltaSpeed / 10);
  rDistance := rDistance + ceil(rWind / 10);

//Time
  rEstimatedTime := (rDistance / rAverageSpeed) * 60;

  redQ1_3.Clear;
  redQ1_3.Lines.Add('Vertical Speed: ' + FloatToStrF(rVerticalSpeed,ffFixed,10,2) + ' (fpm)' +
                    #13 +
                   'Distance: ' + FloatToStrF(rDistance, ffFixed, 10, 2) + ' (nm)' + #13 +
                   'Estimated Time: ' + FloatToStrF(rEstimatedTime, ffFixed, 10, 2) + ' (min)');

end;
```

VRAAG 2

```
//////////     35 punte     //////////

//////////    Vraag 2.1.1  –  4 punte     //////////
procedure TfrmQuestion2.btnQuestion2_1_1Click(Sender: TObject);
// Provided code - DO NOT DELETE OR ALTER //
var
  sSQL1: String;
begin
  /// Enter your code below ///
  sSQL1 := 'SELECT FirstName, Surname, Age, Email ' +
           'FROM tblPilots ' +
           'WHERE Age > 35 AND Age < 43 ' +
           'ORDER BY Age DESC';

  // Provided code - DO NOT DELETE OR ALTER //
  dbCONN.runSQL(sSQL1);
  if length(sSQL1) <> 0 then
    SetGridColumnWidths(dbgSQL);
end;


//////////    Vraag 2.1.2  -   4 punte     //////////
procedure TfrmQuestion2.btnQuestion2_1_2Click(Sender: TObject);
// Provided code - DO NOT DELETE OR ALTER //
var
   sSQL2: String;
   sLine : String;
begin
  /// Enter your code below ///
  sLine := InputBox('Destination','Enter the destination','');

  sSQL2 := 'SELECT * ' +
           'FROM tblFlights ' +
           'WHERE Destination LIKE ' + QuotedStr('%' + sLine + '%');

  // Provided code - DO NOT DELETE OR ALTER //
  dbCONN.runSQL(sSQL2);
  if length(sSQL2) <> 0 then
    SetGridColumnWidths(dbgSQL);
end;


//////////    Question 2.1.3  -   3 Marks     //////////
procedure TfrmQuestion2.btnQuestion2_1_3Click(Sender: TObject);
// Provided code - DO NOT DELETE OR ALTER //
var
  sSQL3 : String;
begin
  /// Enter your code below ///
  sSQL3 := 'SELECT count(*) AS [Flights in September] ' +
           'FROM tblFlights ' +
```

```
        'WHERE Month(DepartureDate) = 9';

  // Provided code - DO NOT DELETE OR ALTER //
  dbCONN.runSQL(sSQL3);
  if length(sSQL3) <> 0 then
    SetGridColumnWidths(dbgSQL);
end;


//////////    Vraag 2.1.4  -   12 punte     //////////
procedure TfrmQuestion2.btnQuestion2_1_4Click(Sender: TObject);
// Provided code - DO NOT DELETE OR ALTER //
var
  sSQL4: String;
begin
  /// Enter your code below ///
  sSQL4 := 'SELECT Destination, ' +
           'Format(sum(PilotCostPerFlight), "Currency") AS [Pilot Cost], ' +
           'Format(sum(FlightCost), "Currency") AS [Flight Cost], ' +
           'Format(sum(PilotCostPerFlight) + sum(FlightCost), "Currency") AS [Total Cost] ' +
           'FROM tblPilots, tblFlights ' +
           'WHERE tblPilots.PilotID = tblFlights.PilotID ' +
           'GROUP BY Destination';

  // Provided code - DO NOT DELETE OR ALTER //
  dbCONN.runSQL(sSQL4);
  if length(sSQL4) <> 0 then
    SetGridColumnWidths(dbgSQL);
end;


//////////    Vraag 2.1.5  -   3 punte     //////////
procedure TfrmQuestion2.btnQuestion2_1_5Click(Sender: TObject);
// Provided code - DO NOT DELETE OR ALTER //
var
  sSQL5: String;
begin
  /// Enter your code below ///
  sSQL5 := 'UPDATE tblPilots ' +
           'SET PilotCostPerFlight = PilotCostPerFlight * 1.07 ' +
           'WHERE LicenseType = "CPL"';

 // Provided code - DO NOT DELETE OR ALTER //
 dbCONN.executeSQL(sSQL5,dbgPilots,dbgFlights,dbgSQL);
 if length(sSQL5) <> 0 then
   SetGridColumnWidths(dbgSQL);
end;
```

```
//////////    Vraag 2.2.1  -   3 punte     //////////
procedure TfrmQuestion2.btnQuestion2_2_1Click(Sender: TObject);
begin
  // Provided code - DO NOT DELETE OR ALTER //
  redQ2_Output.Clear;

with tblPilot do
    begin
      Open;
      redQ2_Output.Lines.Add('Pilots before regulation change: ' + IntToStr(RecordCount));
      First;
      while not (eof) do
        begin
          /// Enter your code below ///
          if FieldByName('Age').AsInteger < 21 then
            Delete
          else
            Next;
        end;
      redQ2_Output.Lines.Add('Pilots after regulation change: ' + IntToStr(RecordCount));
    end;
end;


//////////    Vraag 2.2.2  -   6 punte     //////////
procedure TfrmQuestion2.btnQuestion2_2_2Click(Sender: TObject);

// Provided code - DO NOT DELETE OR ALTER //
var
  sDestination : string;
  iCPL, iPPL, iMPL, iCoPilot : Integer;

begin

// Provided code - DO NOT DELETE OR ALTER //
  redQ2_Output.Clear;
  iCPL := 0;
  iPPL := 0;
  iMPL := 0;
  iCoPilot := 0;

 with tblFlight do
    begin
      Open;
      sDestination := cmbQ2_2_2_Destination.Text;
      redQ2_Output.SelAttributes.Style := [fsBold];
      redQ2_Output.Lines.Add(sDestination);
      redQ2_Output.Lines.Add('-------------------------');
      First;
```

```
 while not (eof) do
       begin
         /// Enter your code below ///
         if FieldByName('Destination').AsString = sDestination then
           begin
             if FieldByName('LicenseRequired').AsString = 'CPL' then
               inc(iCPL)
               else
               if FieldByName('LicenseRequired').AsString = 'PPL' then
                 inc(iPPL)
                 else
                 if FieldByName('LicenseRequired').AsString = 'MPL' then
                   inc(iMPL);
             if (FieldByName('CoPilotRequired').AsBoolean = True) then
               inc(iCoPilot);
           end;
         Next;
       end;
     redQ2_Output.Lines.Add('CPL: ' + IntToStr(iCPL) + #13 +
                            'PPL: ' + IntToStr(iPPL) + #13 +
                            'MPL: ' + IntToStr(iMPL) + #13 +
                            'Co-Pilots required: ' + IntToStr(iCoPilot));
   end;
end;
```

VRAAG 3

```
//////////     35 punte     //////////

//////////     Vraag 3.1.1 – 5 punte     //////////
constructor TAircraft.Create(sName, sManufacturer   : String;
                     rSpeed, rHeight, rRange, rWeight     : Real;
                     rWingspan                            : Real;
                     iFirepower                           : Integer;
                     sImageName, sCountry                 : String;
                     sDescription                         : WideString);
begin
   fName         := sName;
   fManufacturer := sManufacturer;
   fSpeed        := rSpeed;
   fHeight       := rHeight;
   fRange        := rRange;
   fWeight       := rWeight;
   fWingspan     := rWingspan;
   fFirepower    := iFirepower;
   fImageName    := sImageName;
   fCountry      := sCountry;
   fDescription  := sDescription;
end;


//////////     Vraag 3.1.2  –  5 punte     //////////
procedure TAircraft.setValues;
begin
   fSpeed      := fSpeed * 1.852;          //Knots converted to KM/h
   fHeight     := fHeight * 0.3048;        //Feet converted to Meters
   fRange      := fRange * 1.60934;        //Miles converted to KM
   fWeight     := fWeight * 0.45359;       //Pounds converted to KG
   fWingspan   := fWingspan * 0.3048       //Feet converted to Meters
end;


//////////     Question 3.2.1 - 20 marks     //////////
procedure TfrmQuestion3.cmbQ3_SelectAircraftChange(Sender: TObject);
var
  sName, sManufacturer, sImageName, sCountry: String;
  rSpeed, rHeight, rRange, rWeight          : Real;
  rWingspan                                 : Real;
  iFirepower                                : Integer;
  sDescription                              : Widestring;
  MyFile                                    : Textfile;
  sOneLine                                  : String;
  sSearch                                   : String;
  iPos                                      : Integer;
  bFound                                    : Boolean;
```

```
begin
  sSearch := cmbQ3_SelectAircraft.Text;

  AssignFile(MyFile, 'Aircraft_List.csv');
  try
    Reset(MyFile);
  except
    ShowMessage('File not found');
    Exit;
  end;

  bFound := False;
  Readln(MyFile, sOneLine); //Skip the heading line in the textfile
  while (not eof(MyFile)) AND (bFound = False) do
    begin
      Readln(MyFile, sOneLine);
      if pos(UpperCase(sSearch),UpperCase(sOneline)) <> 0 then
        begin
        //Aircraft Name
          iPos := pos(',',sOneLine);
          sName := copy(sOneLine,1,iPos-1);
          delete(sOneLine,1,iPos);
        //Manufacturer Name
          iPos := pos(',',sOneLine);
          sManufacturer := copy(sOneLine,1,iPos-1);
          delete(sOneLine,1,iPos);
        //Speed
          iPos := pos(',',sOneLine);
          rSpeed := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
        //Range
          iPos := pos(',',sOneLine);
          rRange := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
        //Weight
          iPos := pos(',',sOneLine);
          rWeight := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
        //Height
          iPos := pos(',',sOneLine);
          rHeight := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
        //Wingspan
          iPos := pos(',',sOneLine);
          rWingspan := StrToFloat(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
        //Firepower
          iPos := pos(',',sOneLine);
          iFirepower := StrToInt(copy(sOneLine,1,iPos-1));
          delete(sOneLine,1,iPos);
        //Image Name
          iPos := pos(',',sOneLine);
          sImageName := copy(sOneLine,1,iPos-1);
```

```
        delete(sOneLine,1,iPos);
      //Country
        iPos := pos(',',sOneLine);
        sCountry := copy(sOneLine,1,iPos-1);
        delete(sOneLine,1,iPos);
      //Description
        sDescription := sOneLine;
      objAircraft := TAircraft.Create(sName,sManufacturer,rSpeed,rHeight,rRange,
                          rWeight,rWingspan,iFirepower,sImageName,sCountry,sDescription);
        bFound := True;
   end;
  end;

  if bFound then
    begin
      objAircraft.SetValues;
     lblQ3_AircraftName.Caption := objAircraft.getName;
     lblQ3_Manufacturer.Caption := objAircraft.getManufacturer;
     imgQ3_CountryFlag.Picture.LoadFromFile('Images\Flags\' + objAircraft.getCountry);
     imgQ3_AircraftImage.Picture.LoadFromFile('Images\Aircraft\' +
            objAircraft.getImageName);
     lblQ3_AircraftDescription.Caption := objAircraft.getDescription;
     lblQ3_MaxSpeed.Caption := FloatToStrF(objAircraft.getSpeed,ffFixed,10,0);
     lblQ3_MaxHeight.Caption := FloatToStrF(objAircraft.getHeight,ffFixed,10,0);
     lblQ3_Range.Caption := FloatToStrF(objAircraft.getRange,ffFixed,10,0);
     lblQ3_MaxTakeoffWeight.Caption := FloatToStrF(objAircraft.getWeight,ffFixed,10,0);
     lblQ3_Wingspan.Caption := FloatToStrF(objAircraft.getWingspan,ffFixed,10,2);
     lblQ3_Firepower.Caption := IntToStr(objAircraft.getFirepower);
    end;
end;
```


VRAAG 4

```
//////////     40 punte     //////////

//////////     Vraag 4.1 – 6 punte     //////////
procedure TfrmQuestion4.cmbQ4_1_DestinationChange(Sender: TObject);
var
  iRow, iCol : Integer;
begin
  case cmbQ4_1_Destination.ItemIndex of //Case or If Statement
    0 : ar2Booking := ar2Bloemfontein;
    1 : ar2Booking := ar2CapeTown;
    2 : ar2Booking := ar2Durban;
    3 : ar2Booking := ar2EastLondon;
    4 : ar2Booking := ar2Johannesburg;
  end;

//  //Alternate Solution
//  case cmbQ4_Destination.ItemIndex of  //Case or If Statement
//    0 : for iRow := 0 to 14 do
//          for iCol := 0 to 4 do
//            ar2Booking[iRow,iCol] := ar2Bloemfontein[iRow,iCol];
//    1 : for iRow := 0 to 14 do
//          for iCol := 0 to 4 do
//            ar2Booking[iRow,iCol] := ar2CapeTown[iRow,iCol];
//    2 : for iRow := 0 to 14 do
//          for iCol := 0 to 4 do
//            ar2Booking[iRow,iCol] := ar2Durban[iRow,iCol];
//    3 : for iRow := 0 to 14 do
//          for iCol := 0 to 4 do
//            ar2Booking[iRow,iCol] := ar2EastLondon[iRow,iCol];
//    4 : for iRow := 0 to 14 do
//          for iCol := 0 to 4 do
//            ar2Booking[iRow,iCol] := ar2Johannesburg[iRow,iCol];
//  end;

  //Provide Code - DO NOT DELETE
  PaintColour;
end;


//////////     Question 4.2 - 17 marks     //////////
procedure TfrmQuestion4.pnlQ4_2_BookClick(Sender: TObject);
var
  sDestination, sDate, sNameSur, sTime : String;
  sClass, sLine : String;
  rPrice : Real;
  iRow, iCol : Integer;
  cCol : Char;
begin
```

```
  redQ4_Output.Clear;
  iRow := sedQ4_RowNumber.Value;

 case cmbQ4_ColNumber.ItemIndex of
    0 : begin
          iCol := 0;
          cCol := 'A';
        end;
    1 : begin
          iCol := 1;
          cCol := 'B';
        end;
    2 : begin
          iCol := 3;
          cCol := 'C';
        end;
    3 : begin
          iCol := 4;
          cCol := 'D';
        end;
  end;

  if ar2Booking[iRow-1,iCol] = 'B' then
    MessageDLG('Seat has already been booked',MTInformation,[MBOK],0)
  else
  begin
    ar2Booking[iRow-1,iCol] := 'B';
    sDestination := cmbQ4_1_Destination.Text;
    sDate := DateToStr(dtpQ4_Date.Date);
    sTime := cmbQ4_Time.Text;
    sNameSur := edtQ4_NameSur.Text;
    sClass := 'Economy Class';
    rPrice := arrPrice[cmbQ4_1_Destination.ItemIndex];
    case iRow of
      1,2 : begin
              rPrice := rPrice * 1.95;
              sClass := 'Business Class';
            end;
    end;
    sLine := 'Name and Surname: ' + #13#9 + sNameSur + #13 +
             'Destination: ' + #13#9 + sDestination + #13 +
             'Date and Time: ' + #13#9 + sDate + #13 +
                #9 + sTime  + ' flight' + #13 +
             'Cabin: ' + #13#9 + sClass + #13 +
             'Seat Number: ' + #13#9 + cCol + IntToStr(iRow + 1) + #13 +
             'Price: ' + #13#9 + FloatToStrF(rPrice,ffCurrency,10,2);
    redQ4_Output.SelAttributes.Style := [fsBold];
    redQ4_Output.Lines.Add('Booking Information' + #13);
    redQ4_Output.Lines.Add(sLine);
  end;
PaintColour;
end;
```

```
//////////     Vraag 4.3 – 17 punte     //////////
procedure TfrmQuestion4.pnlQ4_3_StatsClick(Sender: TObject);
var
  sLine : String;
  iRow, iCol : Integer;
  iBusClass, iEcoClass : Integer;
  rBusPrice, rEcoPrice : Real;
begin
  redQ4_Output.Clear;
  redQ4_Output.Paragraph.TabCount := 1;
  redQ4_Output.Paragraph.Tab[0] := 10;
  iBusClass := 0;
  iEcoClass := 0;
  rBusPrice := 0;
  rEcoPrice := 0;
  for iRow := 0 to 1 do
    for iCol := 0 to 4 do
      begin
        if ar2Booking[iRow,iCol] = 'B' then
          inc(iBusClass);
      end;

  for iRow := 2 to 14 do
    for iCol := 0 to 4 do
      begin
        if ar2Booking[iRow,iCol] = 'B' then
          inc(iEcoClass);
      end;
  rBusPrice := iBusClass * (arrPrice[cmbQ4_1_Destination.ItemIndex] * 1.95);
  rEcoPrice := iEcoClass * (arrPrice[cmbQ4_1_Destination.ItemIndex]);
  redQ4_Output.SelAttributes.Style := [fsBold];

  sLine := 'Passengers' + #13 +
           #9 + 'Business Class: ' + IntToStr(iBusClass) + #13 +
           #9 + 'Economy Class: ' + IntToStr(iEcoClass) + #13 +
           #9 + 'Total: ' + IntToStr(iBusClass + iEcoClass) + #13#13 +
           'Cost' + #13 +
           #9 + 'Business Class: ' + FloatToStrF(rBusPrice,ffCurrency,10,2) + #13 +
           #9 + 'Economy Class: ' + FloatToStrF(rEcoPrice,ffCurrency,10,2) + #13 +
           #9 + 'Total Cost: ' + FloatToStrF(rBusPrice + rEcoPrice,ffCurrency,10,2);

  redQ4_Output.Lines.Add('Statistics of Flight' + #13);
  redQ4_Output.Lines.Add(sLine);
end;
```

TOTAAL: 150