Province of the
EASTERN CAPE
EDUCATION

NATIONAL SENIOR CERTIFICATE

GRADE 12

SEPTEMBER 2019

INFORMATION TECHNOLOGY P1 MARKING GUIDELINE

MARKS: 150

This marking guideline consists of 24 pages.

NAME OF LEARNER:				
TOTAL QUESTION 1:	**TOTAL QUESTION 2:**	**TOTAL QUESTION 3:**	**TOTAL QUESTION 4:**	**TOTAL**
/37	**/41**	**/36**	**/36**	**/150**

QUESTION 1: GENERAL PROGRAMMING SKILLS		**Max. MARKS**	**MARKS ACHIEVED**
1.1 1.1.1	**BUTTON: [1.1 Join]** Declaring string variables✓ and age as an integer variable✓ Get E-mail and password as string variables✓ Get Age✓ stored as integer Get the Grade✓	**5**	
1.1.2	Initialise counters for capitals and numbers✓ Loop for the length of the password✓ If a character is a capital letter✓ then increment the correct counter✓ If a character is a number from 0 to 9✓ then increment the correct counter✓ check if all conditions are true: ✓ If length of password >= 8✓ and counter for capital letters >=1✓ and counter for numbers >=2✓ and age >= 16✓ and grade >= 10✓ and email contains '@' symbol✓ then set pnlQuestion1_2 to enabled✓ else✓ set pnlQuestion1_2 to disabled✓	**16**	
1.2 1.2.1	**BUTTON: [1.2 Upload]** Obtain the category name from the combobox ✓ Obtain the speed and the size, both converted to a number ✓	**2**	
1.2.2	If length of title > 23✓ then clear the edit box edtTitle✓ and exit ✓	**3**	

1.2.3	Calculate time to upload in seconds (size/speed) ✓ Test for a whole number (using frac and trunc or any other successful method) ✓ If time to upload is not a whole number (not an integer) Then round up to next whole number✓ Calculate minutes (time to upload in seconds / 60) ✓	**4**	
1.2.4	Compile the video code✓ using: A random number from 10 to 100✓ last character of the title✓ length of the title✓	**4**	
1.2.5	Display the Title, Video code, Category and Minutes in the richedit (redUploads) in correct order✓ Minutes formatted to 1 decimal place ✓ using spacing (#9) ✓	**3**	
	TOTAL QUESTION 1	**37**	

QUESTION 2: OBJECT-ORIENTATED PROGRAMMING		Max. MARKS	MARKS ACHIEVED
2.1.1	**Constructor Create:** Correct name✓ with string parameter✓ Set fvideoname to parameter value✓ Set flikes and fdislikes to 0✓	**4**	
2.1.2	**Procedure AddNewComment** Correct method - procedure✓ 2 string parameters✓ Joined to attribute fcomments✓ *first empty line (#13)* Comment parameter added✓ Tab space (#9)✓ Date parameter added✓ *last empty line (#13)* Empty lines added (#13) in the correct places✓	**7**	
2.1.3	**Procedure SetRating;** Correct method – procedure ✓ Calculate the difference fLikes - fDislikes✓ Use case or nested IF statements✓ Set attribute fRating to ✓ 2, if difference is 1 to 3✓ 5, if difference is > 3 ✓ If frating <= 0 then frating = 0✓	**7**	
2.1.4	**Function GetRating: string;** correct method , a string function or a procedure with a variable string parameter✓ Initialise a local variable to an empty string✓ Loop from 1 to the attribute value, frating✓ Join stars (* symbol) to local string variable✓ OR If frating = 2 then star string = '**' If frating = 5 then star string = '*****" If frating = 0 then star string = empty string (*3 marks*) Return the local string variable✓	**5**	
	2.1 Subtotal: Object class	**23**	

2.2.1	**TQuestion2.lbxVideosClick** Instantiate the object Object name = ✓ classname.create✓ with argument name of video✓	**3**	
2.2.2	**Button: [2.2.2 Like] and** Call the addLike method with correct object name✓ **Button: [2.2.2 Dislike]** Call the addDislike method with correct object name✓	**2**	
2.2.3	**Button: [2.2.3 Watch this video]** Inputbox or messagedlg for Y or N response from user✓ If "Y" then bitbtnLike.click✓ Else bitbtnDislike.click✓ Call the setrating procedure with correct object name✓ Call the getRating function with correct object name✓ and display in the panel pnlRatings✓	**6**	
2.2.4	**Button: [2.2.4 Submit]** Join the name and comment from the input components✓ Comment on a separate line✓ Call the AddNewComment method with correct object name✓ with arguments : joined name and comment compiled into one string and the SystemDate ✓ Clear the richedit ✓ Richedit component for display✓ Object name/ toString✓	**7**	
	2.2 Subtotal: Form class	**18**	
	TOTAL QUESTION 2	**41**	

QUESTION 3: DATABASE PROGRAMMING		**MAX. MARKS**	**MARKS ACHIEVED**
3.1.1	**Button: [3.1.1]**	**4**	
	SQL: Select DateCreated, ProfileName from Profile order by DateCreated DESC		

	Concepts:
	SELECT all correct fields✓
	FROM correct table✓
	ORDER BY correct field✓ DESC ✓		

3.1.2	**Button: [3.1.2]**	**9**	
	SQL: Select DatePublished, format((Likes - Dislikes + Comments)/Views, "Currency",2) AS [Earnings] from Video, Profile where Video.ProfileID = Profile.ProfileID and ProfileName = ‘ + QUOTEDSTR(SLINE)		
	Concepts: SELECT correct field (DatePublished) ✓ Calculation of 4 correct fields✓ format to currency✓ AS specified field name✓ FROM ✓ Video Profile (2 tables) ✓ WHERE clause to link tables✓ AND ✓correct condition and correct syntax✓		
3.1.3	**Button: [3.1.3]**	**4**	
	SQL: Select VideoTitle from Video where Views >= 1000 and Dislikes <= 10		

	Concepts:
	SELECT correct field✓
	FROM correct table✓
	WHERE Views >= 1000✓
	AND Dislikes <=10✓		

3.1.4	**Button: [3.1.4]**	**4**	
	SQL: Delete from Video where YEAR(DatePublished) = 2008		

	Concepts:
	DELETE✓
	FROM correct table✓
	WHERE Year function✓ DatePublished = 2008✓		

3.1.5	**Button: [3.1.5]**	**3**	
	SQL: Update Profile set ProfileName = "Tim Horton" where ProfileName = "TimmyTom"		

	Concepts:
	UPDATE correct table✓
	SET correct field = Tim Horton✓
	WHERE correct field = TimmyTom✓		

	3.1 Subtotal: SQL	**24**	

3.2.1	**Button: [3.2.1]**	**6**	

```
Move to first record of tblVideos ✓
Loop while not end of table✓
    If uppercase✓ VideoTitle field contains 'THE'✓
      Then display VideoTitle field in richedit✓
    Move to next record before end of loop✓
```

3.2.2	**Button: [3.2.2]**	**6**	

```
Move to first record of tblVideos ✓
Loop while not end of table✓
 If Dislikes field >= 9✓
   Then delete✓
       Else✓
     Move to next record before end of loop✓
```

	3.2 Subtotal: Code constructs	**12**	
	TOTAL QUESTION 3	**36**	

QUESTION 4: PROBLEM-SOLVING		MAX. MARKS	MARKS ACHIEVED
4.1	**MENU OPTION [4.1 Sort and Remove]**	17	

```
Outer loop from 1 to array count – 1✓
  Inner loop from outer loop counter to array count✓
  If arrprofiles[outerloop] < arrprofiles[innerloop]  ✓
  Then Set temp veriable to arrprofiles[outerloop] ✓
     Set arrprofiles[outerloop] = arrprofiles[innerloop] ✓
     Set arrprofiles[innerloop] = temp variable✓
     perform the same swap alogrithm for arrminutes ✓

outer loop from 1 to 7  (1st 7 elements to remove) ✓
  inner loop from 1 to array count - 1✓
    replace element in arrprofiles with the next index✓
    replace element in arrminutes with the next index✓
subtract 7 from array counter after nested loops✓

Assignfile (text file name, profile name + '.txt') ✓
Rewrite(text file name); ✓
Loop to array counter ✓
     Writeln✓(text file name, arrprofiles and arrminutes
                                     converted to string) ✓
(closefile statement)
```

4.2	**MENU OPTION [4.3 Advertisements]**	19	

```
Extract name from combo box✓
Using 2 nested loops✓
Set the 2D array contents to 0 ✓
  (subtract 1 mark if a procedure to reset the array
                                        was not written)
Set counter = 0✓
Loop from 1 to array counter✓
   If name = string in array arrprofiles (correct index)✓
     then get minutes from arrminutes(correct index) ✓
       increment counter✓
       Set space = minutes divided by 4✓

 Use Case or nested If statements:✓
 For 2 adverts:
    Add space to 2D array at row=counter, column=1✓
    set space = 3 x space✓
    Add space to 2D array at row=counter, column=3✓
For 4 adverts:
    Set temp variable = space✓
    Loop from 1 to 4 ✓ (4 entries made to 2darray)
          Add temp to 2D array at row=counter, column=
                                          loop variable ✓
          Set temp = temp + space✓
End of if statement in correct position✓
Call the procedure Display in the correct place✓
```

TOTAL QUESTION 4	**36**	

SAMPLE SOLUTIONS

QUESTION 1

```
unit Question1_u;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ComCtrls, ExtCtrls, Spin, Math;

type
  TQuestion_1 = class(TForm)
    pnlQuestion1_1: TPanel;
    Label10: TLabel;
    edtEmail: TEdit;
    Label1: TLabel;
    edtPassword: TEdit;
    Label2: TLabel;
    rdgGrade: TRadioGroup;
    sedAge: TSpinEdit;
    Label5: TLabel;
    Q1_1btnJoin: TButton;
    pnlQuestion1_2: TPanel;
    Label11: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    edtSpeed: TEdit;
    edtSize: TEdit;
    edtTitle: TEdit;
    Q1_2btnUpload: TButton;
    Label9: TLabel;
    redUploads: TRichEdit;
    Label4: TLabel;
    cmbCategory: TComboBox;
    Label3: TLabel;
    Label12: TLabel;
    lblOutput: TLabel;
    procedure Q1_1btnJoinClick(Sender: TObject);
    procedure Q1_2btnUploadClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Question_1: TQuestion_1;
implementation
{$R *.dfm}
```

```
procedure TQuestion_1.Q1_1btnJoinClick(Sender: TObject);
var semail, spass, sline : string;
iage, igrade, k, icountnumbers, icountcaps, icountother  : integer;
bvalpass, bvalemail, bvalgrage : boolean;
begin
//=====================================================================
// Question 1.1.1   5 marks
//=====================================================================
  semail := edtemail.Text;
  spass := edtpassword.Text;
  iage := sedage.value;
  igrade := rgpgrade.ItemIndex + 8;

//=====================================================================
// Question 1.1.2   16 marks
//=====================================================================
 bvalpass := false;
 bvalemail := false;
 bvalgrage := false;
  if pos('@', semail) > 0 then
        bvalemail := true;

  /// checking the password
   icountcaps := 0;
   icountnumbers := 0;
     for k := 1 to length(spass) do
        begin
          if spass[k] in ['0'..'9'] then
             inc(icountnumbers);
          if spass[k] in ['A'..'Z'] then
             inc(icountcaps);
        end;
if (length(spass) >= 8) and (icountcaps >=1) and (icountnumbers >= 2) then
       bvalpass := true;

if (iage >= 16) and (igrade >=10) then
      bvalgrage := true;

if bvalemail  and bvalpass and bvalgrage then
  pnlQuestion1_2.enabled := true;
else
  pnlQuestion1_2.enabled := false;

 end;
```

```
procedure TQuestion_1.Q1_2btnUploadClick(Sender: TObject);
var scat, stitle,  svidid : string;
rspeed, rsize, rtime, rremaining, rminutes : real;
imaxtimetoupload, iseconds, irandomcharacter : integer;
begin
//==================================================================
// Question 1.2.1   2 marks
//==================================================================
 scat := cmbcategory.Items[cmbcategory.itemindex];
 stitle := edttitle.Text;
 rspeed := strtofloat(edtspeed.Text);
 rsize := strtofloat(edtsize.Text);
//==================================================================
// Question 1.2.2   3 marks
//==================================================================
  if length(stitle) > 23 then
      begin
        edttitle.Clear;
        exit;
      end;
//==================================================================
// Question 1.2.3   4 marks
//==================================================================
rtime := rsize/rspeed;  //calculate minutes to upload
iseconds := trunc(rtime);
rremaining := frac(rtime);
if rremaining > 0 then
  inc(iseconds);
rminutes := iseconds/60;
//==================================================================
// Question 1.2.4   4 marks
//==================================================================
svidid := inttostr(randomrange(10,101)) + stitle[length(stitle)] + inttostr(length(stitle));
//==================================================================
// Question 1.2.5   3 marks
//==================================================================
 reduploads.Lines.Add(stitle + #9 +svidid + #9 + scat + #9 +
floattostrf(rminutes,fffixed,8,1));
end;

procedure TQuestion_1.FormActivate(Sender: TObject);
begin

///Provided code do not delete///
reduploads.Paragraph.TabCount := 3;
reduploads.Paragraph.tab[0] := 95;
reduploads.Paragraph.tab[1] := 130;
reduploads.Paragraph.tab[2]  := 200;
reduploads.Lines.Add('Title of video' + #9 + 'Code' + #9 + 'Category' + #9 + 'Minutes');
reduploads.Lines.Add('-------------------------------------------------------------------------------')
///Provided code do not delete///
end;
```

```
end.
```

QUESTION 2

Class Unit:

```
unit VideoClass;

interface
uses sysutils, dialogs, math;

type
TVideo = class
  private
  fvideoname : string;
  fcomments : string;
  flikes : integer;
  fdislikes : integer;
  frating : integer;
 public
 ///Enter your code here//
    constructor create(sname : string);
    procedure setrating;
    function getrating : string;
    procedure addnewcomment(scomm, sdate : string);

//Provided code // Do NOT change or delete//
    procedure addLike;
    procedure addDislike;
    function tostring : string;
end;

implementation

{ TVideo }
//==================================================================
//  Question 2.1.1   4 marks
//==================================================================
constructor tvideo.create(sname: string);
begin
fvideoname := sname;
flikes := 0;

fdislikes := 0;
end;

//==================================================================
//  Question 2.1.2   7 marks
//==================================================================
procedure tvideo.addnewcomment(scomm, sdate: string);
begin
  fcomments := fcomments + #13+ scomm +#9 + sdate  + #13;
end;
```

```
//======================================================================
//  Question 2.1.3   7 marks
//======================================================================
procedure tvideo.setrating;
var idiff : integer;
begin
idiff := flikes - fdislikes ;
if idiff > 0 then
 begin
case idiff of
 1..3 : frating := 2;
 else
    frating := 5;
  end;
 end;

if idiff <= 0 then frating := 0;
end;

//======================================================================
//  Question 2.1.4   5 marks
//======================================================================
function tvideo.getrating: string;
var k : integer;
sstar : string;
begin
  sstar := '';
  for k := 1 to frating do
     sstar := sstar + '*';
result := sstar;
end;

//Provided code // Do NOT change or delete//
procedure tvideo.addDislike;
begin

inc(fdislikes);
end;

//Provided code // Do NOT change or delete//
procedure tvideo.addLike;
begin
inc(flikes);
end;

//Provided code // Do NOT change or delete//
function tvideo.tostring: string;
begin
  result := fvideoname+ #13 + fcomments;
end;

end.
```

Main Unit:

```
unit Question2_u;

interface

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

type
  TQuestion2 = class(TForm)
    redComments: TRichEdit;
    imgVideo: TImage;
    Label2: TLabel;
    Label3: TLabel;
    Q2_2_3btnView: TButton;
    pnlStats: TPanel;
    pnlRatings: TPanel;
    bitbtnDislike: TBitBtn;
    bitbtnLike: TBitBtn;
    pnlComment: TPanel;
    Q2_2_4btnSubmit: TButton;
    edtcomment: TEdit;
    Label4: TLabel;
    edtname: TEdit;
    Label1: TLabel;
    Label6: TLabel;
    lblvideo: TLabel;
    lbxVideos: TListBox;
    procedure Q2_2_3btnViewClick(Sender: TObject);
    procedure Q2_2_4btnSubmitClick(Sender: TObject);
    procedure bitbtnLikeClick(Sender: TObject);
    procedure bitbtnDislikeClick(Sender: TObject);
    procedure lbxVideosClick(Sender: TObject);

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

var
//Provided code // Do NOT change or delete//
  Question2: TQuestion2;
  objVideo : tvideo;
  Systemdate : string;

implementation

{$R *.dfm}
```

```
procedure TQuestion2.lbxVideosClick(Sender: TObject);
begin
//Provided code // Do NOT change or delete//
 pnlstats.visible := false;
 pnlcomment.visible := false;
 redcomments.Clear;
 edtcomment.clear;
 edtname.Clear;
 pnlratings.Caption := '';
// add your code below for Question 2.2.1
//=====================================================================
//  Question 2.2.1   3 marks
//=====================================================================

objvideo := tVideo.create(lbxvideos.Items[lbxvideos.ItemIndex]);

end;

//=====================================================================
//  Question 2.2.2   1 mark
//=====================================================================
procedure TQuestion2.bitbtnLikeClick(Sender: TObject);
begin
objvideo.addLike;
end;

//=====================================================================
//  Question 2.2.2   1 mark
//=====================================================================
procedure TQuestion2.bitbtnDislikeClick(Sender: TObject);

begin
objvideo.addDislike;
end;

procedure TQuestion2.Q2_2_3btnViewClick(Sender: TObject);
begin
//Provided code // Do NOT change or delete//
 pnlstats.visible := true;
 pnlcomment.visible := true;
 imgvideo.picture.loadfromfile(lbxvideos.Items[lbxvideos.ItemIndex] + '.jpg');
 lblvideo.Caption :=  lbxvideos.Items[lbxvideos.ItemIndex];
 systemdate := datetostr(date);

///Add your code below///
//=====================================================================
//  Question 2.2.3   6 marks
//=====================================================================
 if uppercase(inputbox('Do you like this video?', 'Answer Y or N','')) = 'Y' then
    bitbtnlike.Click
  else
    bitbtndislike.Click;
```

```
objvideo.setrating;
pnlratings.Caption := objvideo.getrating;
end;

//==============================================================
//  Question 2.2.4   7 marks
//==============================================================
procedure TQuestion2.Q2_2_4btnSubmitClick(Sender: TObject);
var scomment : string;
begin
scomment := edtname.Text + #13 + edtcomment.Text;
objvideo.addnewcomment(scomment,systemdate);
redcomments.Clear;
redcomments.Lines.add(objvideo.tostring);
end;

end.
```

QUESTION 3

```
unit Question3_u;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Grids, DBGrids, StdCtrls, ComCtrls, ExtCtrls, dbConnection_u, DB,
  ADODB;
type
  TQuestion_3 = class(TForm)
    dbgVideos: TDBGrid;
    btnRestoreDB: TButton;
    dbgProfiles: TDBGrid;
    dbgResults: TDBGrid;
    redDisplay: TRichEdit;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    btnQuestion3_1_1: TButton;
    btnQuestion3_1_4: TButton;
    btnQuestion3_2_1: TButton;
    btnQuestion3_2_2: TButton;
    btnQuestion3_1_5: TButton;
    btnQuestion3_1_2: TButton;
    btnQuestion3_1_3: TButton;
    procedure btnQuestion3_1_1Click(Sender: TObject);
    procedure btnRestoreDBClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure btnQuestion3_1_2Click(Sender: TObject);
    procedure btnQuestion3_1_3Click(Sender: TObject);
    procedure btnQuestion3_1_4Click(Sender: TObject);
    procedure btnQuestion3_1_5Click(Sender: TObject);
    procedure btnQuestion3_2_2Click(Sender: TObject);
    procedure btnQuestion3_2_1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Question_3: TQuestion_3;
  // --- Global variables provided ---
  // Provided code - do not change
  dbCONN: TConnection;
  qryDB: TADOQuery;
  tblProfiles, tblVideos: TADOTable;
  // Provided code - do not change
implementation
{$R *.dfm}
{$R+}
```

```
//==============================================================
//  Question 3.1.1   4 marks
//==============================================================
procedure TQuestion_3.btnQuestion3_1_1Click(Sender: TObject);
var
  sSQL1: String;
begin

  sSQL1 := 'select DateCreated, ProfileName from Profile order by DateCreated DESC';

  // Provided code - do not change
  dbCONN.runSQL(sSQL1);
   // Provided code - do not change
end;
//==============================================================
//  Question 3.1.2   9 marks
//==============================================================
procedure TQuestion_3.btnQuestion3_1_2Click(Sender: TObject);
// Provided code - do not change/
var
  sline : string;
  sSQL2: String;
// Provided code - do not change/
begin
  // Provided code - do not change//////////////////////
  sline := inputbox('Enter Name of Profile','','DonDoe');
  // Provided code - do not change//////////////////////

  sSQL2 := 'Select DatePublished, format((Likes - Dislikes + Comments)/Views,
"Currency",2) AS [Earnings] ' +
                  'from Video, Profile where Video.ProfileID = Profile.ProfileID and ProfileName
= '  + QUOTEDSTR(SLINE);

  // Provided code - do not change
  dbCONN.runSQL(sSQL2);
  // Provided code - do not change
end;
//==============================================================
//  Question 3.1.3   4 marks
//==============================================================
procedure TQuestion_3.btnQuestion3_1_3Click(Sender: TObject);
// Provided code - do not change
var
  sSQL3: String;
// Provided code - do not change
begin

  sSQL3 := 'Select VideoTitle from Video where Views >= 1000 and Dislikes <= 10';

  // Provided code - do not change
  dbCONN.runSQL(sSQL3);
  // Provided code - do not change

end;
```

```
//=====================================================================
//  Question 3.1.4   4 marks
//=====================================================================
procedure TQuestion_3.btnQuestion3_1_4Click(Sender: TObject);
// Provided code - do not change
var
  sSQL4: String;
// Provided code - do not change
begin

  sSQL4 := 'Delete from Video where YEAR(DatePublished) = 2008';

  // Provided code - do not change
  dbCONN.executeSQL(sSQL4,dbgProfiles,dbgVideos,dbgResults);
  // Provided code - do not change
end;

//=====================================================================
//  Question 3.1.5   3 marks
//=====================================================================
procedure TQuestion_3.btnQuestion3_1_5Click(Sender: TObject);
// Provided code - do not change
var
  sSQL5: String;
// Provided code - do not change
begin

  sSQL5 := 'Update Profile set ProfileName = "Tim Horton" where ProfileName =
"TimmyTom"';

 // Provided code - do not change
 dbCONN.executeSQL(sSQL5, dbgProfiles,dbgVideos, dbgResults);
 // Provided code - do not change
end;

//=====================================================================
//  Question 3.2.1   6 marks
//=====================================================================
procedure TQuestion_3.btnQuestion3_2_1Click(Sender: TObject);

begin
/// enter your code below//
reddisplay.Clear;
tblvideos.First;
while not tblvideos.eof do
begin
  if pos('THE ',uppercase(tblvideos['Videotitle'])) <> 0 then
  reddisplay.Lines.Add(tblvideos['VideoTitle']);
   tblvideos.Next;
end;end;

//=====================================================================
//  Question 3.2.2   6 marks
//=====================================================================
procedure TQuestion_3.btnQuestion3_2_2Click(Sender: TObject);
begin
/// enter your code below//
tblvideos.First;
while not tblvideos.eof do
begin
  if tblvideos['Dislikes'] >= 9 then
  tblvideos.Delete
  else
  tblvideos.Next;
end;
end;

// Provided code - do not change
{$REGION DB CONNECTION}
 // Provided code - do not change
//Setup DB connections - DO NOT CHANGE!
procedure TQuestion_3.btnRestoreDBClick(Sender: TObject);
begin
  // Restores the Database
  dbCONN.RestoreDatabase(dbgProfiles, dbgVideos, dbgResults);
  reddisplay.Clear;
  // Formatting field datatypes
  tblProfiles := dbCONN.tblOne;
  tblVideos := dbCONN.tblMany;
  qryDB := dbCONN.qrymovies;
end;
 // Provided code - do not change
 // Provided code - do not change
procedure TQuestion_3.FormActivate(Sender: TObject);
var
  AnsW: string;
begin
  // Sets up the connection to database and opens the tables.
  dbCONN := TConnection.Create;
  dbCONN.dbConnect;

  tblProfiles := dbCONN.tblOne;
  tblVideos := dbCONN.tblMany;
  qryDB := dbCONN.qrymovies;
  dbCONN.setupGrids(dbgProfiles, dbgVideos, dbgResults);
 end;
// Provided code - do not change
procedure TQuestion_3.FormClose(Sender: TObject; var Action: TCloseAction);
begin // Disconnects from database and closes all open connections
  dbCONN.dbDisconnect;
end;
 // Provided code - do not change
{$ENDREGION}
// Provided code - do not change   end.
```

QUESTION 4

```
unit Question4_u;

interface

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

type
  TQuestion_4 = class(TForm)
    redOutput: TRichEdit;
    Menu: TMainMenu;
    Q4_1SortandRemove: TMenuItem;
    Q4_2Advertisements: TMenuItem;
    cmbProfiles: TComboBox;
    stgAdverts: TStringGrid;
    Label1: TLabel;
    Label2: TLabel;
    procedure resetarray;
    ///Provided code. Do NOT delete////////
    procedure display;
    ///Provided code. Do NOT delete////////
    procedure FormActivate(Sender: TObject);
    procedure Q4_1SortandRemoveClick(Sender: TObject);
    procedure Q4_2AdvertisementsClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Question_4: TQuestion_4;
   ///Provided code. Do NOT delete////////
   icountarr : integer;
   arrprofiles : array[1..1000] of string;

   arrminutes : array[1..1000] of integer;
   arr2D : array[1..8,1..4] of real;
   ///Provided code. Do NOT delete////////

implementation

{$R *.dfm}

//=====================================================================
//  Question 4.1   17 marks
//=====================================================================
procedure TQuestion_4.Q4_1SortandRemoveClick(Sender: TObject);
var
```

```
myfile : textfile;
stempname: string;
itempmins : integer;
k,l, x,m : integer;
begin
for k := 1 to icountarr - 1 do
   for l := k + 1 to icountarr do
      begin
        if arrprofiles[k] > arrprofiles[l] then
            begin
              stempname := arrprofiles[k];
              arrprofiles[k] := arrprofiles[l];
              arrprofiles[l] := stempname;

              itempmins := arrminutes[k];
              arrminutes[k] := arrminutes[l];
              arrminutes[l] := itempmins;
            end;
      end;

for x := 1 to 7  do
begin
 for m := 1 to icountarr - 1 do
  begin
    arrprofiles[m] := arrprofiles[m + 1];
    arrminutes[m] := arrminutes[m + 1];
  end;
end;
icountarr := icountarr - 7;

display; // not required

assignfile(myfile,'Revised.txt');
rewrite(myfile);
for k := 1 to icountarr do
  writeln(myfile, arrprofiles[k]+ ';' + inttostr(arrminutes[k]));

closefile(myfile);
end;

//=====================================================================
//  Question 4.2   19 marks
//=====================================================================
procedure TQuestion_4.resetarray;
var irow, icol : integer;
begin
  for irow := 1 to 8 do
      for icol := 1 to 4 do
         arr2d[irow,icol] := 0;
end;
procedure TQuestion_4.Q4_2AdvertisementsClick(Sender: TObject);
var
```

```
//Provided code. Do NOT delete////
inumber : integer;
//Provided code. Do NOT delete////
  k,x, icount: integer;
  sname: string;
  rmins, rspace, rnew : real;
 begin
///Provided code. Do NOT
delete//////////////////////////////////////////////////////////////////////////////////////////////////////////////
inumber := strtoint(inputbox('How many advertisements?','Only 2 or 4 allowed',''));
////Provided code. Do NOT
delete//////////////////////////////////////////////////////////////////////////////////////////////

resetarray;
 icount := 0;
 sname := cmbprofiles.Text;
 for k := 1 to icountarr do
  begin
    if arrprofiles[k] = sname then
      begin
      rmins := arrminutes[k];
       inc(icount);
       rspace := rmins/4;
        case inumber of
            2 : begin
                 arr2D[icount,1] := rspace;
                 rspace:=  rspace + rspace + rspace;
                 arr2D[icount,3] := rspace;
                 end;
           4 : begin
                rnew := rspace;
                for x := 1 to 4  do
                 begin
                   arr2D[icount,x] := rnew;
                   rnew := rnew + rspace;

                 end;
              end
       end;
    end; // end of if
  end;  // end of for loop
 display;
end;

////provided code Do not delete//////////////////////////////////
procedure TQuestion_4.display;
var irow, icol, k: integer;
begin
  stgadverts.cells[0,0] := 'Videos';
  for irow := 1 to 8 do
     stgadverts.cells[0,irow] := inttostr(irow);
     stgadverts.cells[1,0] := '1st Position';
```

```
   stgadverts.cells[2,0] :=  '2nd Position';
   stgadverts.cells[3,0] :=  '3rd Position';
     stgadverts.cells[4,0] :=  'Last Position';
 for irow := 1 to 8 do
    for icol := 1 to 4 do
      begin
       if arr2d[irow,icol] = 0 then
       stgadverts.cells[icol,irow] := 'No advert'
        else
       stgadverts.cells[icol,irow] := floattostrf(arr2d[irow,icol], fffixed,8,1);
      end;
redoutput.Paragraph.TabCount := 1;
redoutput.Paragraph.tab[0] := 50;
redoutput.Clear;
redoutput.Lines.Add('Profile Name' + #9 + 'Length of video in minutes');
redoutput.Lines.Add('-------------------------------------------------------');
for k := 1 to icountarr do
  redoutput.Lines.Add(arrprofiles[k]  + #9 + floattostr(arrminutes[k]));
end;
////provided code Do not delete/////////////////////////////////////
procedure TQuestion_4.FormActivate(Sender: TObject);
var  myfile: textfile;
sline : string;
k : integer;
begin
///provided code Do not delete///////////////////////////////////////////
icountarr  := 0;
assignfile(myfile,'Videos.txt');
reset(myfile);
 while not eof(myfile) do
   begin
     readln(myfile, sline);
     inc(icountarr);
     arrprofiles[icountarr] := copy(sline,1,pos(';',sline) - 1);
     delete(sline,1,pos(';',sline));
     delete(sline,1,pos(';',sline));

     arrminutes[icountarr] := round(strtofloat(sline)/60);
   end;
display;
////provided code Do not delete////////////////////////////////////////////////////
end;
end.
```

TOTAL: 150