



Province of the
EASTERN CAPE
EDUCATION

**NATIONAL
SENIOR CERTIFICATE**

GRADE 11

NOVEMBER 2012

INFORMATION TECHNOLOGY P1

MARKS: 120

TIME: 3 hours



This question paper consists of 11 pages.

INSTRUCTIONS AND INFORMATION

1. This is a three-hour examination. Because of the nature of this examination it is important to note that you will NOT be permitted to leave the examination room before the end of the examination session.
2. You require the files listed below in order to answer the questions. They are EITHER on a stiffy disk OR CD issued to you, OR the invigilator/educator will tell you where to find them on the hard drive of the workstation you are using OR in which network folder it is. If the files are issued to you on a CD, you need to copy them onto your hard disk.

QUESTION 1

Tuckshop.mdb
Suppliers.txt
Stock.txt
Question1_UX.pas
Question1_UX.dfm
Question1_PX.dpr

QUESTION 2

Stock.txt
Question2_UX.pas
Question2_UX.dfm
Question2_PX.dpr

QUESTION 3

Question3_UX.pas
Question3_UX.dfm
Question3_PX.dpr

If a disk or CD containing the files was issued to you, write your surname on the label.

3. Save your work at regular intervals as a precaution against power failures.
4. Read ALL the questions carefully. Do only what is required by the question.
5. During the examination you may use the manuals originally supplied with the hardware and software. You may also use the HELP functions of the software. You may NOT refer to any other resource material.
6. At the end of this examination session you will be required to hand in the stiffy or CD given to you by the invigilator with your work saved on it, or you must make sure that all your work has been saved on the network as explained to you by the invigilator/educator. Ensure that all files can be read.
7. You also have to hand in printouts of the programming code for all the questions you have done.
8. All printing of programming questions will take place within an hour of the completion of the examination.

SCENARIO

The school tuck-shop is a very lucrative business and demands excellent management. The choices of stock, as well as which suppliers to use, are important decisions that have to be made on a regular basis. As you are providing a service to learners the prices have to be quite reasonable and making a profit is not always the main aim, rather healthy food choices.

QUESTION 1 DATABASE AND DELPHI

The database, **Tuckshop.mdb**, which contains data related to this topic, has been supplied to you in a folder named **Question 1**.

Two text files have been supplied as well. If you cannot use the database provided, use the text files named **Suppliers.txt** and **Stock.txt** to create your own database named **Tuckshop.mdb** containing two tables named **Suppliers** and **Stock**. Change the data types of the fields of the tables to the specifications given below. Create a one-to-one relationship between the two tables.

The **Suppliers** table stores data about the suppliers of stock for the tuck-shop. The fields in this table are defined as follows:

Suppliers	
Field Name	Data Type
SupplierNo	Text
Supplier	Text
Address	Text
ContactNo	Text

The following table is an example of the data contained in the table named **Suppliers** in the database named **Tuckshop.mdb**.

Suppliers			
SupplierNo	Supplier	Address	ContactNo
BETBAT	Better Batters	61 Highveld Ro	0837128781
FFGROU	Fridge Foods G	71-75 Caxton S	0437220666
PUDFUD	Pudge Fudge	54 Kings Way, C	0437261442
QUEDIS	Quenera Distri	2 Strelitzia Stre	0437221981
RJSALE	RJ SALES	21 Union Ave	0437211895
SIMBA	Simba (PTY) LTI	8 Belgravia Cre	0437361424
W2GEL	Way-2-Go Sale	8 Ridge Road, E	0828216340
WILBUT	Wilson's Butch	57 Caxton Stree	0437222246

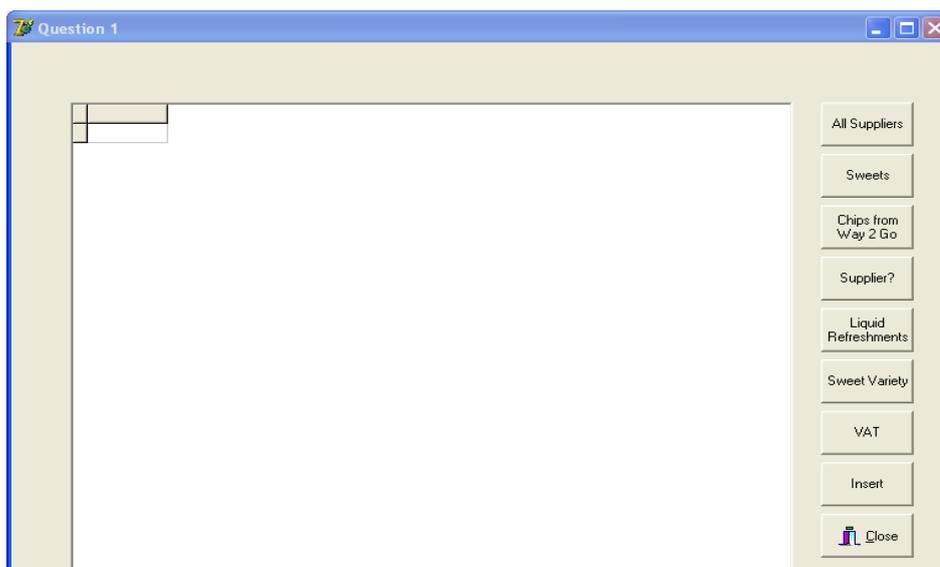
The **Stock** table stores data about the stock that is either sold or used in the tuck-shop. The fields in this table are defined as follows:

Field Name	Data Type
StockID	AutoNumber
Type	Text
Description	Text
Units	Number
UnitPrice	Currency
SupplierID	Text

The following table is an example of the data contained in the table named **Stock** in the database named **Tuckshop.mdb**.

StockID	Type	Description	Units	UnitPrice	SupplierID
1	Chips	Doritos Sweet	48	R 118.00	SIMBA
2	Chips	Doritos Supren	48	R 118.00	SIMBA
3	Chips	Simba Creamy	48	R 118.00	SIMBA
4	Chips	Simba Mrs Ball	48	R 118.00	SIMBA
5	Chips	Lays Lightly Sal	48	R 118.00	SIMBA
6	Chips	Lays Sour Crear	48	R 118.00	SIMBA
7	Chips	Lays Thai Swee	48	R 118.00	SIMBA
8	Sweets	Beacon Maynai	24	R 78.94	W2GEL
9	Sweets	Beacon Sour Ba	24	R 78.94	W2GEL
10	Sweets	Beacon Fruit Je	24	R 78.94	W2GEL
11	Sweets	Beacon Sour Je	24	R 78.94	W2GEL
12	Sweets	Beacon Teddie	24	R 78.94	W2GEL
13	Flavoured Wat	Tsitsikamma Be	6	R 20.88	W2GEL
14	Flavoured Wat	Tsitsikamma Le	6	R 20.88	W2GFI

You have also been supplied with an incomplete Delphi program with a unit named **Question1_UX** and a project named **Question1_PX** in the folder named **Question 1**. Open the incomplete program.



- The program should be able to connect to the database named **Tuckshop.mdb**.
- When you do QUESTION 1.1 and you find that the connectivity is not in place, use the following steps to establish connection with the database:
 - Click on the ADOQuery component named **qryTuckshop**.
 - Click on the Ellipse button (three dots) to the right of the ConnectionString property in the Object Inspector.
 - Click on the Build button which takes you to the Data Link Properties dialogue box.
 - Select Microsoft Jet 4.0 OLE DB Provider and click on Next.
 - The first option on the Connection tab sheet allows you to browse and find the **Tuckshop.mdb** file.
 - Remove the user name Admin.
 - Click on the Test Connection button.
 - Click OK on each one of the open dialogue windows.

NOTE: If you cannot establish connectivity with the database at all when you execute the program you must still do and submit the programming code for marking.

Marks will only be awarded for the programming code that contains the SQL statements in the unit named Question1_UX as well as code that makes use of an inputbox.

- 1.1 Complete the code in the **All Suppliers** button by formulating an SQL statement to display all information from the Suppliers table, sorted according to the Supplier.

Example of output:

SupplierNo	Supplier	Address	ContactNo
BETBAT	Better Batters	61 Highveld Road, Berea	0837128781
FFGROU	Fridge Foods Group	71-75 Caxton Street, Quigney	0437220666
PUDFUD	Pudge Fudge	54 Kings Way, Cambridge	0437261442
QUEDIS	Quenera Distributing (PTY) LTD	2 Strelitzia Street, Braelyn	0437221981
RJSALE	RJ SALES	21 Union Ave	0437211895
SIMBA	Simba (PTY) LTD	8 Belgravia Crescent, Wilsonia	0437361424
W2GEL	Way-2-Go Sales East London CC	8 Ridge Road, Beacon Bay	0828216340
WILBUT	Wilson's Butchery	57 Caxton Street, Quigney	0437222246

(3)

- 1.2 Complete the code in the **Sweets** button by formulating an SQL statement to display the Description as well as UnitPrice of all the sweets in the tuck-shop.

Example of output of the first few records:

Description	UnitPrice
▶ Beacon Maynards Jelly Babies	78.94
Beacon Sour Babies	78.94
Beacon Fruit Jelly Beans	78.94
Beacon Sour Jelly Beans	78.94
Beacon Teddies	78.94
Beacon Liquorope Sour	60.72
Beacon Mallow Mice	46.96
Beacon Mini Babies	78.94

(3)

- 1.3 Complete the code in the **Chips from Way 2 Go** button by formulating an SQL statement to display all the different chips that is received from the Supplier 'Way 2 Go'.

Example of output:

StockID	Type	Description	Units	UnitPrice	SupplierID
▶ 28	Chips	Chippanaks Beef	50	23.68	W2GEL
29	Chips	Chippanaks Tomato	50	23.68	W2GEL
30	Chips	Chippanaks Chutney	50	23.68	W2GEL
31	Chips	Chippanaks Cheese	50	23.68	W2GEL

(4)

- 1.4 Complete the code in the **Supplier?** button by formulating an SQL statement to display all the information about the supplier as requested by the user. Make use of an InputBox component to enter the SupplierNo.

Example of output if the SupplierNo **SIMBA** was entered:

SupplierNo	Supplier	Address	ContactNo
▶ SIMBA	Simba (PTY) LTD	8 Belgravia Crescent, Wilsonia	0437361424

(4)

- 1.5 Complete the code in the **Liquid Refreshments** button by formulating an SQL statement to display the Supplier, Description as well as the UnitPrice of all the liquid refreshments containing the words drink/water/juice. NB: You will need to link the tables with an appropriate **where** clause to be able to do this.

Example of output:

supplier	description	unitprice
▶ Quenera Distributing (PTY) LTD	Aquella Lemon	22.2
Quenera Distributing (PTY) LTD	Aquella Litchi	22.2
Quenera Distributing (PTY) LTD	Aquella Marula	22.2
Quenera Distributing (PTY) LTD	Aquella Melon Honey	22.2
Quenera Distributing (PTY) LTD	Aquella Naartjie	22.2
Quenera Distributing (PTY) LTD	Aquella Strawberry	22.2
Quenera Distributing (PTY) LTD	Aquella Still Sports Bottle	25.2
Quenera Distributing (PTY) LTD	Lahha Pineapple	42.2

(7)

- 1.6 Complete the code in the **Sweet Variety** button by formulating an SQL statement to count how many different sweets the tuck-shop stocks. Display the output with a suitable heading.

Example of output:

Number of Different Sweets	
▶	20

(4)

- 1.7 Complete the code in the **VAT** button by formulating an SQL statement that will determine the amount of VAT (14%) paid on the UnitPrice. Display the StockID, Type, Description, UnitPrice as well as the calculated field VAT.

Example of output of the first few records:

Stockid	Type	description	unitprice	VAT
▶ 1	Chips	Doritos Sweet Chilli	118	R 16.52
2	Chips	Doritos Supreme Cheese	118	R 16.52
3	Chips	Simba Creamy Cheddar	118	R 16.52
4	Chips	Simba Mrs Balls Chutney	118	R 16.52
5	Chips	Lays Lightly Salted	118	R 16.52
6	Chips	Lays Sour Cream and Cheese	118	R 16.52
7	Chips	Lays Thai Sweet Chilli	118	R 16.52
8	Sweets	Beacon Maynards Jelly Babies	78.94	R 11.05
9	Sweets	Beacon Sour Babies	78.94	R 11.05
10	Sweets	Beacon Fruit Jelly Beans	78.94	R 11.05
11	Sweets	Beacon Sour Jelly Beans	78.94	R 11.05
12	Sweets	Beacon Teddies	78.94	R 11.05

(6)

- 1.8 Complete the code in the **Insert** button by formulating an SQL statement that will insert a new supplier. Once the new supplier has been added, display all the information in the Suppliers table.

New Supplier's Information:

SupplierNo: RJSALE

Supplier: RJ SALES

Address: 21 Union Ave

ContactNo: 0437211895

(5)

- Enter your name and surname as a comment line in the first line of the file named **Question1_UX.pas** containing the SQL statements.
- Save the unit **Question1_UX** and the project **Question1_PX** (File|Save All).
- A printout of the code of the **Question1_UX.pas** file will be required.

[36]

QUESTION 2 DELPHI PROGRAMMING

The Grade 11 Council members have to assist in the tuck-shop during break. They rotate their duty on a weekly basis. In this program you will work out the duties for a period of a month as well as many calculations regarding the stock. The stock information is stored in a text file and the Council members' names are stored in an array **arrLearners**. The code to get the information from the text file has already been written and information is stored in an array called **arrItems**. Do not delete or alter the code.

arrItems contents are stored as follows: **Name of Item,Units,UnitPrice**

- 2.1 Write a procedure named **Items** which will separate the contents stored in **arrItems**. The array must be separated into the Name of the Item (**arrName**), units (**arrUnits**), as well as the price per item (**arrPrice**). (10)
- 2.2 Write code for the **Calculate Costs** button which will call procedure **Items**. The quantities will be generated randomly (values between 1 and 20). The amount excluding VAT must be calculated for each item based on the quantity ordered, as well as the amount including VAT (14%). Once the amount including VAT has been determined, the selling price per unit can then be calculated. The profit is calculated if all the stock is sold.

Display the information and values formatted to two decimals neatly in columns in the **richedit** component.

Quantity	Description	Excl VAT	Incl VAT	Selling Price Per Unit
1	Doritos Supreme Cheese	118.00	134.52	5.75
4	Simba Creamy Cheddar	472.00	538.08	5.75
17	Lays Lightly Salted	2006.00	2286.84	5.75
9	Beacon Maynards Jelly Babies	710.46	809.92	7.70
1	Beacon Sour Babies	78.94	89.99	7.70
12	Twizza Orange	420.96	479.89	6.84
2	Twizza Grape	70.16	79.98	6.84
3	Beacon Original Winegums	235.44	268.40	7.65
17	Aquella Lemon	377.40	430.24	8.66
20	Aquella Naartjie	444.00	506.16	8.66
8	Aquella Strawberry	177.60	202.46	8.66
13	Aquella Still Sports Bottle	327.60	373.46	9.83
8	Muffins	403.20	459.65	9.83
Profit if ALL stock sold: 7010.11				

(17)

2.3 Write code for the **Duties** button which will sort the Council members' names in alphabetical order. Once the names are sorted they must be put in groups of five to work over a period of four weeks.

```
Week 1  
Andrea  
Chantel  
Gabriella  
Henry  
Jack  
  
Week 2  
John  
Jonathan  
Lisa  
Liz
```

(13)

- Enter your name and surname as a comment line in the first line of the file named **Question2_UX.pas**.
- Save the unit **Question2_UX** and the project **Question2_PX** (File|Save All).
- Make a printout of the code of the **Question2_UX.pas**.

[40]

QUESTION 3 DELPHI PROGRAMMING

The Department of Education required information about Grade Eleven learners in order to start the registration process before the Grade Twelve final exams the following year. Grade Eleven learners are requested to provide their ID numbers. Many times the ID numbers are recorded incorrectly and with this program those errors can be eliminated.

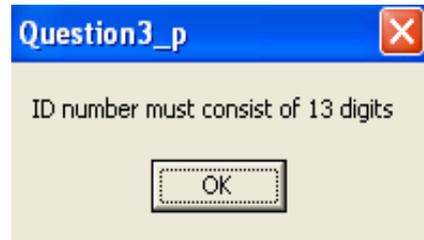
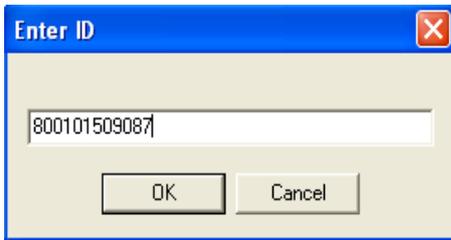
- 3.1 3.1.1 Write a function called **LengthID** which will receive the ID number as a parameter and will return a boolean value indicating that the ID number is the correct length. All South African ID numbers must contain 13 digits. (4)
- 3.1.2 Write a procedure called **DateOfBirth** which will receive the ID number as a parameter and will return the date of birth in the following format: YYYY/MM/DD. The first six digits of an ID number represent the YYMMDD. (8)
- 3.1.3 Write a function called **Gender** which will receive the ID number as a parameter and will return a value M or F indicating the gender. The seventh digit of an ID number is the gender digit. If it is less than 5 (0 to 4) the person is female and if the digit is between 5 and 9 (inclusive) then the person is male. (5)
- 3.1.4 Write a function called **Verify** which will receive the ID number as a parameter and will return a boolean value. An ID number is verified as follows:
ID Number 8001015009087 as an example:
- Add all the digits in the odd positions (excluding the last digit)
 $8 + 0 + 0 + 5 + 0 + 0 = 13$
 - The digits in the even positions must be multiplied by 2
 $011098 * 2 = 22196$
 - Add all the digits after it has been multiplied
 $2 + 2 + 1 + 9 + 6 = 20$
 - Add the answers of step 1 and step 3
 $13 + 20 = 33$
 - Subtract the second digit from 10.
 $10 - 3 = 7$
 - If the answer is the same as the last digit of the ID number then the ID number is correct else it is invalid.
 $7 = 7 \rightarrow$ ID NUMBER IS VALID
 - Example output on next page. (5)
- 3.2 Write code for the **Enter ID number** EventHandler which will ask the user to enter an ID number by making use of an InputBox component. If the ID number does not consist of 13 digits then a message must be displayed informing the user and must also enable the user to enter the ID number again until it consists of 13 digits.
The ID number must be verified and if it is correct it must display a suitable message in the richedit component and also the display the gender and date of birth. If the ID number is not correct a message must be displayed in the richedit component. (14)

- Enter your name and surname as a comment line in the first line of the file named **Question3_UX.pas**.
- Save the unit **Question3_UX** and the project **Question3_PX** (File|Save All).
- Make a printout of the code of the **Question3_UX.pas** file.

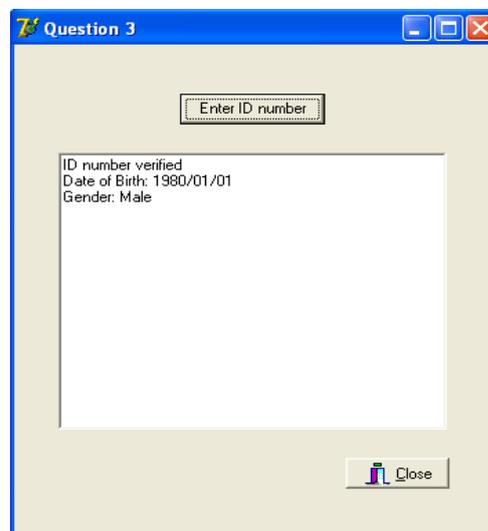
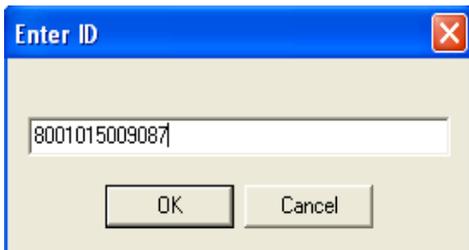
[44]

Example:

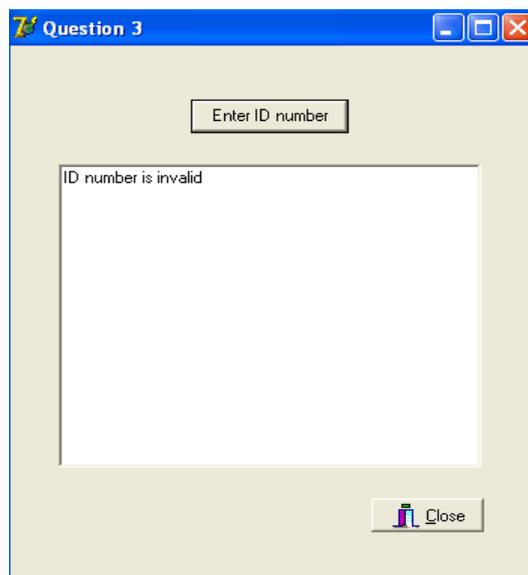
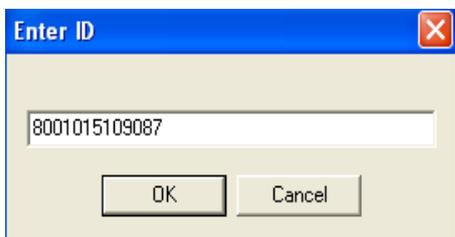
INCORRECT NUMBER OF DIGITS:



CORRECT ID NUMBER:



INVALID ID NUMBER:



TOTAL: 120