basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

INFORMATION TECHNOLOGY P1

NOVEMBER 2018

MARKS: 150

TIME: 3 hours

This question paper consists of 19 pages and 2 data pages.

INSTRUCTIONS AND INFORMATION

1.   This question paper is divided into THREE sections. Candidates must answer ALL the questions in ALL THREE sections.

2.   The duration of this examination is three hours. 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.

3.   This question paper is set with programming terms that are specific to the Delphi programming language.

4.   Make sure that you answer the questions according to the specifications that are given in each question. Marks will be awarded according to the set requirements.

5.   Answer only what is asked in each question. For example, if the question does not ask for data validation, then no marks will be awarded for data validation.

6.   Your programs must be coded in such a way that they will work with any data and not just the sample data supplied or any data extracts that appear in the question paper.

7.   Routines, such as search, sort and selection, must be developed from first principles. You may NOT use the built-in features of Delphi for any of these routines.

8.   All data structures must be defined by you, the programmer, unless the data structures are supplied.

9.   You must save your work regularly on the disk/CD/DVD/flash disk you have been given, or on the disk space allocated to you for this examination session.

10.   Make sure that your examination number appears as a comment in every program that you code, as well as on every event indicated.

11.   If required, print the programming code of all the programs/classes that you completed. You will be given half an hour printing time after the examination session.

12.   At the end of this examination session you must hand in a disk/CD/DVD/ flash disk with all your work saved on it OR you must make sure that all your work has been saved on the disk space allocated to you for this examination session. Ensure that all files can be read.

13.   The files that you need to complete this question paper have been given to you on the disk/CD/DVD/flash disk or on the disk space allocated to you. The files are provided in the form of password-protected executable files.

NOTE: Candidates must use the file **DataENGNov2018.exe**.

Do the following:

- Double click on the password-protected executable file.
- Click on the extract button.
- Enter the following password: **Rest2Rant*&**

Once extracted, the following list of files will be available in the folder **DataENGNov2018**:

SUPPLIED FILES:

Question1:
Question1_P.dpr
Question1_P.dproj
Question1_P.res
Question1_U.dfm
Question1_U.pas

Question2:
ConnectDB_U.pas
PaymentsDB.mdb
PaymentsDBBackup.mdb
Question2_P.dpr
Question2_P.dproj
Question2_P.res
Question2_U.dfm
Question2_U.pas

Question3:
Question3_P.dpr
Question3_P.dproj
Question3_P.res
Question3_U.dfm
Question3_U.pas
Restaurant_U.pas

Question4:
Question4_P.dpr
Question4_P.dproj
Question4_P.res
Question4_U.dfm
Question4_U.pas
Visitors.txt

SECTION A

QUESTION 1: GENERAL PROGRAMMING SKILLS

Do the following:

- Open the incomplete program in the **Question1** folder.
- Enter your examination number as a comment in the first line of the **Question1_U.pas** file.
- Compile and execute the program. The user interface displays FOUR tab sheets labelled QUESTION 1.1 to QUESTION 1.4. The program has no functionality currently.
- Follow the instructions below to complete the code for EACH section of QUESTION 1, as described in QUESTION 1.1 to QUESTION 1.4.

1.1　　**Tab sheet [Question 1.1]**

　　　　Panel [1.1 – Display heading]

　　　　Write code to set the properties of panel **pnlQ1_1** as follows:

- Background colour to lime
- Font colour to red
- Font size to 20 pt
- Caption to 'Information Technology Paper 1'
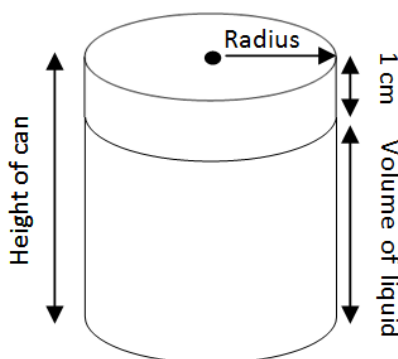
　　　　Example of output:

<div align="center">

Information Technology Paper 1

</div>

　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　(4)

1.2　　**Tab sheet [Question 1.2]**

　　　　Button [1.2 – Volume]

　　　　A supplier of cooldrink cans must determine the volume of a can before it is filled with liquid.

　　　　All cans are filled with liquid, leaving a space of **1 cm** from the top.

Write code to do the following:

- Declare variables for the height and radius of the can.

- Calculate the <u>volume</u> of the liquid required (according to the specifications) to fill a can if the height and radius of the can are provided as input from the user.

  The volume of a cylinder (can) is calculated as follows:

  $$\text{Volume} = \text{pi} \times \text{radius}^2 \times \text{height of cylinder}$$

- Use a dialog box to display the volume of the liquid in the can, calculated to ONE decimal place.

Example of input and output if the height of the can is 5.4 cm and the radius is 1.2 cm:

| Enter height | 5.4 | cm |
| Enter radius | 1.2 | cm |

The volume is 19.9.

    OK

(9)

1.3 **Tab sheet [Question 1.3]**

Button [1.3 – Display factors and check if it is a prime number]

Write code to do the following:

- Declare suitable variables.
- Clear the rich edit component **redQ1_3**.
- Generate a random integer in the range 5 to 50 (inclusive).
- Determine and display the factors of the generated number in the rich edit component **redQ1_3**.
- If the number generated is a prime number, display the number and a message indicating that it is a prime number.

NOTE:
- A factor is a number that is divisible by another number with no remainder.
- A prime number has only two factors, namely the value of 1 and the number itself.

Example of output if the number generated is 6:

```
1
2
3
6
```

Example of output if the number generated is 13:

```
1
13

13 is a prime number.
```

(13)

1.4 **Tab sheet [Question 1.4]**

Button [1.4 – Enter line of instructions and display commands]

A robot must receive a line of instructions to move out of a maze using no more than 10 steps forward in total. The robot reacts to the characters S, R and L, which represent the following commands:

S: One step forward
R: Turn right
L: Turn left

Example of a line of instructions: **SSSRSLSLLSSR**

Code is provided that does the following:

- Convert the line of instructions entered into capital letters and assign the line of instructions to the variable **sInstructions**.
- Clear the output area **redQ1_4**.

Write code to do the following:

- Display the line of instructions that has been entered in the output area **redQ1_4**, followed by a blank line.
- Decode the line of instructions into short descriptive commands indicating the movements of the robot.
- Display the descriptive commands in the rich edit component **redQ1_4**, as shown in the example that follows.
- As soon as the number of forward steps exceeds the value of 10, the message 'Number of forward steps exceeds 10' must be displayed and none of the remaining commands contained in the line of instructions must be decoded.

NOTE: Your code must be able to decode any line of instructions and not only the line of instructions shown in the examples that follow.

Example of output if SSSRSLSLLSSR was entered as the line of instructions:

```
SSSRSLSLLSSR

Step forward
Step forward
Step forward
Turn right
Step forward
Turn left
Step forward
Turn left
Turn left
Step forward
Step forward
Turn right
```

Example of output if SSSRSLSLLSSRSSSS was entered as the line of instructions:

```
SSSRSLSLLSSRSSSS

Step forward
Step forward
Step forward
Turn right
Step forward
Turn left
Step forward
Turn left
Turn left
Step forward
Step forward
Turn right
Step forward
Step forward
Step forward
Number of forward steps exceeds 10.
```

(14)

- Ensure that your examination number has been entered as a comment in the first line of the program file.
- Save your program.
- Print the code if required.

TOTAL SECTION A: 40

SECTION B

QUESTION 2: DATABASE PROGRAMMING

The database **PaymentsDB** contains the information of the staff members of a restaurant. The database contains two tables, namely **tblEmployees** and **tblPayments**, with data specifically related to the year 2017.

The data pages attached at the end of the question paper provide information on the design of the database and its contents.

Do the following:

- Open the incomplete project file called **Question2_P.dpr** in the **Question2** folder.
- Enter your examination number as a comment in the first line of the **Question2_U.pas** unit file.
- Compile and execute the program. The program has no functionality currently.

  The following user interface is displayed:

tblEmployees

EmployeeNumber	Surname	FirstName	IDNumber	Permanent	Children
3	van Staden	Johan	8708191623067	True	6
4	Walker	Ann	8708206699090	True	4
5	Cilliers	Lize	8710016190088	False	2
6	Powell	Brian	8710183187027	True	1
7	Nel	Riana	8710243529025	False	4

tblPayments

PaymentNumber	EmployeeNumber	PaymentDate	GrossSalary	Deductions
109	20	2017/12/11	R 3892.00	R 194.60
110	18	2017/05/24	R 4210.00	R 126.30
111	11	2017/04/26	R 3774.00	R 188.70
112	21	2017/04/08	R 5239.00	R 261.95
113	15	2017/05/24	R 5445.00	R 163.35

Question 2.1 - SQL | Question 2.2 - Delphi code

Results:

2.1.1 - Alphabetical list	2.1.4 - Delete payment
2.1.2 - Number of children of permanent employees	2.1.5 - Total net salaries per month
2.1.3 - Employees paid on selected date	

Restore database | Close

- Follow the instructions that follow to complete the code for each section, as described in QUESTION 2.1 and QUESTION 2.2.
- Use SQL code to answer QUESTION 2.1 and Delphi code to answer QUESTION 2.2.

NOTE:

- The **[Restore database]** button is provided to restore the data contained in the database to the original content. If you need to test your code on the original data, you may click this button to restore the data.
- The content of the database is password protected. Therefore you will not be able to gain access to the content of the database with Microsoft Access.
- Do NOT change any of the code provided.
- Code is provided to link the GUI components to the database.
- TWO variables are declared as public variables, as described in the table below.

Variable	Data type	Description
tblEmployees	TADOTable	Refers to the table tblEmployees
tblPayments	TADOTable	Refers to the table tblPayments

2.1   **Tab sheet [Question 2.1 – SQL]**

In this section you may use ONLY SQL statements to answer QUESTION 2.1.1 to QUESTION 2.1.5.

Code to execute the SQL statements and display the results of the queries is provided. The SQL statements are incomplete.

The following user interface is displayed:

Do the following to complete the incomplete SQL statements assigned to the variables **sSQL1**, **sSQL2**, **sSQL3**, **sSQL4** and **sSQL5** per question respectively:

2.1.1   **Button [2.1.1 – Alphabetical list]**

Display ALL details of employees in the **tblEmployees** table, sorted alphabetically according to the surname field.

Example of output of the first four records:

EmployeeNumber	Surname	Firstname	IDNumber	Permanent	Children
11	Adams	Megan	8804108271062	True	6
18	Ajam	Amy	8808023964093	False	1
22	Barth	Luke	8811045557049	False	5
23	Boreham	Olwethu	8812163974057	False	5

(3)

2.1.2   **Button [2.1.2 – Number of children of permanent employees]**

Display the surname, first name and number of children of all the permanent employees with more than three children.

Example of output of the first four records:

Surname	FirstName	Children
van Staden	Johan	6
Walker	Ann	4
Botha	Rina	5
Adams	Megan	6

(5)

2.1.3   **Button [2.1.3 – Employees paid on selected date]**

Display the payment number and ID number of all the employees who were paid on 2017/01/17.

Example of output:

PaymentNumber	IDNumber
131	8708191623067
149	8804101172067

(6)

2.1.4   **Button [2.1.4 – Delete payment]**

Delete the record with the payment number 110 from the **tblPayments** table.

Code has been provided to display a message to indicate that the content of the database was changed.

(3)

2.1.5   **Button [2.1.5 – Total net salaries per month]**

The net salary of an employee is calculated by subtracting the deductions from the gross salary.

Calculate and display the total amount that the restaurant has paid towards net salaries per month. Display the month using the field name **MonthNum** and display the calculated amount using the field name **TotalAmountPaid**. Format the amount as currency.

Example of output for the first three months if the original data was used:

MonthNum	TotalAmountPaid
1	R 48 979.69
2	R 45 462.89
3	R 39 069.39

(8)

2.2   **Tab sheet [Question 2.2 – Delphi code]**

In this section only Delphi programming code may be used to answer QUESTION 2.2.1 to QUESTION 2.2.3.

NO marks will be awarded for SQL statements in QUESTION 2.2.

The user interface for QUESTION 2.2 is shown below.

2.2.1   **Button [2.2.1 – Temporary employees]**

Write code to display the surname, first name and number of children of all temporary employees from the **tblEmployees** table in the rich edit component **redQ2**.

NOTE: Code for headings and columns are provided.

Example of output for first four records:

Temporary employees		
Surname	**FirstName**	**Children**
Cilliers	Lize	2
Nel	Riana	4
Magwa	Megan	4
Cerfontein	Lwazi	3

(6)

2.2.2　　**Button [2.2.2 – Add an employee]**

Write code to add a record to the **tblEmployees** table. The data of the employee to be added is provided below.

Surname:　Zwelini
First name:　Lungile
ID number:　7601050179081
Permanent:　Yes
Children:　3

Example of the last few records in the **tblEmployees** table after adding the record for Lungile:

EmployeeNumber	Surname	FirstName	IDNumber	Permanent	Children
26	Makuleni	Nicola	8906138880020	False	6
27	Crawford	Bulumko	9007082259024	True	2
28	Plaatjies	Bulumko	9007082259029	True	2
29	Bosch	Tegan	9007262590039	True	3
30	Zwelini	Lungile	7601050179081	True	3

(5)

2.2.3　　**Button [2.2.3 – Update deductions]**

The deductions amount for a specific payment was captured incorrectly. The user must select a record from the DBGrid **dbgPayments**.

Write code to update the deductions field of the record that has been selected by increasing the deductions amount by 1% of the gross salary.

Example of content of record if payment number 112 was selected:

PaymentNumber	EmployeeNumber	PaymentDate	GrossSalary	Deductions
112	21	2017/04/08	R 5239.00	R 261.95

Example of updated content of record with payment number 112:

PaymentNumber	EmployeeNumber	PaymentDate	GrossSalary	Deductions
112	21	2017/04/08	R 5239.00	R 314.34

(4)

- Ensure that your examination number has been entered as a comment in the first line of the program file.
- Save your program.
- Print the code if required.

TOTAL SECTION B:　40

SECTION C

QUESTION 3: OBJECT-ORIENTATED PROGRAMMING

A new restaurant requires software to compile an identification code and to manage staff numbers.

Do the following:

- Open the incomplete program in the **Question3** folder.
- Open the incomplete object class **Restaurant_U.pas**.
- Enter your examination number as a comment in the first line of both the **Question3_U.pas** file and the **Restaurant_U.pas** file.
- Compile and execute the program. The program has no functionality currently.

    The following user interface is displayed:

- Complete the code as specified in QUESTION 3.1 for the **Restaurant_U** object class and in QUESTION 3.2 for the **Question3_U** form class.

3.1   The incomplete object class (**TRestaurant**) provided contains the following:

- Declarations of three attributes that define a **Restaurant** object
- A completed **toString** method

    The attributes for the **Restaurant** object have been declared as follows:

Names of attributes	Description
fName	Name of the restaurant
fYearOpened	Year the restaurant opened in the format YYYY
fNumEmployees	The number of employees at the restaurant

3.1.1 Write code for a constructor method that will receive the name of the restaurant, the year the restaurant opened and the number of employees as parameter values. Assign these values to the respective attributes. (5)

3.1.2 Write code for a method called **getNumEmployees** that will return the number of employees. (2)

3.1.3 Write code for a method called **increaseNumEmployees** to receive an integer value as a parameter. The number of employees must be increased by the value received as a parameter. (3)

3.1.4 Write code for a method called **compileCode** that receives the full name of the owner as a parameter and compiles and returns an identification code in the following format:

%$YYYY

where:
% represents the first letter of the name of the restaurant
$ represents the last two letters of the full name of the owner
YYYY represents the year the restaurant opened

Example of an identification code for the following input values:

Restaurant name: Simply Fabulous Food
Year opened: 2018
Owner's full name: Pieter van Wyk

Identification code: Syk2018 (7)

3.2 The incomplete unit **Question3_U** provided contains code for the object class to be accessible and an object variable **objRestaurant**.

Follow the instructions below to code the solution.

3.2.1 **Button [3.2.1 – Instantiate and display object]**

The user must enter the name of the restaurant, the year the restaurant opened and the number of employees using the components provided.

The following input data has been provided to test the program:

Restaurant name: Simply Fabulous Food
Year opened: 2018
Number of employees: 25

Write code to do the following:

- Use the data that has been entered to instantiate a restaurant object.
- Use the toString method to display the details of the object in the rich edit component **redQ3**. (8)

3.2.2   **Button [3.2.2 – Identification code]**

The user must enter the full name of the owner in the edit box provided for input.

Write code to do the following:

- Call the **compileCode** method using the full name of the owner as an argument.
- Display the compiled identification code in the edit box **edtIDCode**.

Example of output if the test data, provided as input values in QUESTION 3.2.1, was used and Pieter van Wyk was used as the full name of the owner:

<div align="center">Syk2018</div>

(3)

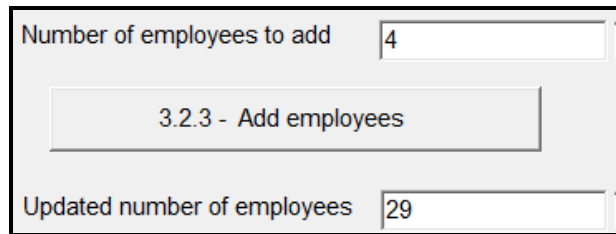3.2.3   **Button [3.2.3 – Add employees]**

The user must enter the number of employees to be added to the edit box **edtAdd**.

Code has been provided to declare a constant variable called **iMaxEmployees** that represents the maximum number of employees at the restaurant and set its value to 40.

Write code to do the following:

- Extract the number of employees entered from the edit box.
- Determine if the extracted number of employees can be added without exceeding the maximum number of employees at the restaurant.
- If the number of employees can be added:
  - Call the relevant method to increase the number of employees.
  - Display the total number of employees in the **edtUpdated** edit box.
- If the number of employees entered cannot be added (exceeds the maximum number of employees), display a suitable message in the **edtUpdated** edit box.

Example of output for the test data used in QUESTION 3.2.1 and the value 4 is now entered for the employees to be added:
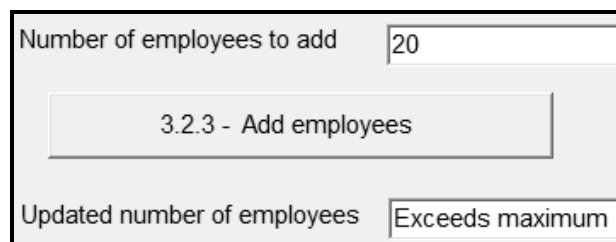
Number of employees to add	4
3.2.3 - Add employees	
Updated number of employees	29

Example of output for the test data used in QUESTION 3.2.1 and the value 20 is now entered for the employees to be added:

Number of employees to add	20
3.2.3 - Add employees	
Updated number of employees	Exceeds maximum

(10)

- Ensure that your examination number has been entered as a comment in the first line of the object class and the form class.
- Save all the files.
- Print the code if required.

TOTAL SECTION C: **38**

SECTION D

QUESTION 4: PROBLEM-SOLVING PROGRAMMING

SCENARIO
A local restaurant kept a record of the number of customers who visited the restaurant per day during the past year.

Do the following:

- Open the incomplete program in the **Question4** folder.
- Enter your examination number as a comment in the first line of the **Question4_U.pas** file.
- Compile and execute the program. The program has no functionality currently.

  The following user interface is displayed:

The program contains the code shown below for the declaration of three arrays called **arrDays**, **arrTempCustomers** and **arrCustomers**.

- **arrDays** is a constant array that contains abbreviations of the seven days of the week.
- **arrTempCustomers** is a constant array that must ONLY be used to answer QUESTION 4.2 if your code to populate **arrCustomers** in QUESTION 4.1 was NOT successful.
- **arrCustomers** is declared with a maximum size of 31 elements.

```
const
  arrDays : array[1..7] of String = ('Sun', 'Mon', 'Tue', 'Wed',
                                     'Thu', 'Fri', 'Sat');
  arrTempCustomers : array[1..31] of integer = (248, 81, 189,
    141, 163, 163,233, 64, 145, 188, 108, 124, 120, 130, 57, 64,
    131, 54, 138, 71, 75, 152, 126, 170, 56, 157, 230, 82, 199,
    119, 136);
var
  arrCustomers : array[1..31] of integer;
```

A text file called **Visitors.txt** is provided and contains 365 lines of data representing the number of customers who visited the restaurant from 1 January 2017 to 31 December 2017. Each line of data contains a date (day–month), a hash character (#) and a number indicating the number of customers who visited the restaurant on that date. The data is saved in the following format:

```
<date>#<number of customers>
```

Example of lines of data in the text file:

```
1-Jan#177
2-Jan#96
3-Jan#152
:
31-Jan#152
1-Feb#174
:
31-Dec#250
```

Explanation of the first two lines of data in the **Visitors.txt** text file:

    On 1 January 177 customers visited the restaurant.
    On 2 January 96 customers visited the restaurant.

NOTE:

- Do NOT change the code provided.
- Apply good programming techniques and modular design in your solution.

Complete the code as described in QUESTION 4.1 and QUESTION 4.2 below.

4.1   **Button [4.1 – Populate customer array]**

The user must select a month from the **cmbMonths** combo box.

Use the text file **Visitors.txt** to populate the **arrCustomers** array with the data of the selected month.

Example of the content in the array if May was selected:

	arrCustomers
1	248
2	81
3	189
:	:
31	136

A message must be displayed to indicate that the array has been populated successfully.   (14)

4.2   **Button [4.2 – Display]**

Use the **cmbDays** combo box to select a day of the week as the first day of the month.

Use the selected day and the populated array from QUESTION 4.1 and write code to display a calendar, as shown below.

The number of customers who visited the restaurant each day must be displayed in brackets.

Example of output if Friday has been selected as the first day of May:

```
Calendar for May

Sun         Mon         Tue         Wed         Thu         Fri         Sat
                                                            1 (248)     2 (81)
3 (189)     4 (141)     5 (163)     6 (163)     7 (233)     8 (64)      9 (145)
10 (188)    11 (108)    12 (124)    13 (120)    14 (130)    15 (57)     16 (64)
17 (131)    18 (54)     19 (138)    20 (71)     21 (75)     22 (152)    23 (126)
24 (170)    25 (56)     26 (157)    27 (230)    28 (82)     29 (199)    30 (119)
31 (136)
```

NOTE: Code to display the heading and subheadings has been provided.

Example of output if Wednesday has been selected as the first day of February:

```
Calendar for Feb

Sun         Mon         Tue         Wed         Thu         Fri         Sat
                                    1 (174)     2 (115)     3 (68)      4 (217)
5 (187)     6 (202)     7 (201)     8 (98)      9 (240)     10 (162)    11 (105)
12 (220)    13 (194)    14 (248)    15 (149)    16 (237)    17 (116)    18 (78)
19 (140)    20 (92)     21 (226)    22 (75)     23 (124)    24 (121)    25 (58)
26 (97)     27 (115)    28 (202)
```

(18)

- Ensure that your examination number has been entered as a comment in the first line of the program file.
- Save your program.
- Print the code if required.

TOTAL SECTION D: 32
GRAND TOTAL: 150

INFORMATION TECHNOLOGY P1

QUESTION 2: DATABASE INFORMATION

DESIGN OF DATABASE TABLES:

Table: **tblEmployees**

This table contains the personal information of the employees working at the restaurant.

Field name	Data type	Description
EmployeeNumber	AutoNumber	A unique number assigned to each employee
Surname	Text (20)	The surname of the employee
FirstName	Text (20)	The first name of the employee
IDNumber	Text (15)	The South African ID number of the employee
Permanent	Boolean	Boolean field indicating whether the employee is part of the permanent staff or not
Children	Number	Number of child dependants of the employee

Example of data of the first ten records:

EmployeeNumber	Surname	FirstName	IDNumber	Permanent	Children
3	van Staden	Johan	8708191623067	☑	6
4	Walker	Ann	8708206699090	☑	4
5	Cilliers	Lize	8710016190088	☐	2
6	Powell	Brian	8710183187027	☑	1
7	Nel	Riana	8710243529025	☐	4
8	Matthews	John	8712018430042	☑	1
9	Velze	Gerda	8802053234052	☑	2
10	Botha	Rina	8803267958013	☑	5
11	Adams	Megan	8804108271062	☑	6
12	Magwa	Megan	8808028315088	☐	4

Table: **tblPayments**

This table contains the records of all the payments made to employees during the past year (2017).

Field name	Data type	Description
PaymentNumber	AutoNumber	A unique number assigned to a payment made to an employee
EmployeeNumber	Number	A number that identifies the employee who received the payment
PaymentDate	Date/Time	The date on which the payment was made
GrossSalary	Currency	The salary before deductions
Deductions	Currency	The total amount that will be subtracted from the gross salary

Example of data of the first ten records:

PaymentNumber	EmployeeNumber	PaymentDate	GrossSalary	Deductions
109	20	2017/12/11	R 3892.00	R 155.68
110	18	2017/05/24	R 4210.00	R 126.30
111	11	2017/04/26	R 3774.00	R 188.70
112	21	2017/04/08	R 5239.00	R 261.95
113	15	2017/05/24	R 5445.00	R 163.35
114	29	2017/06/23	R 6484.00	R 194.52
115	17	2017/03/28	R 3638.00	R 145.52
116	8	2017/07/05	R 4706.00	R 141.18
117	27	2017/06/25	R 5950.00	R 238.00
118	16	2017/09/25	R 4460.00	R 133.80

The following one-to-many relationship with referential integrity exists between the two tables in the database:

tblEmployees
- EmployeeNumber
- Surname
- FirstName
- IDNumber
- Permanent
- Children

tblPayments
- PaymentNumber
- EmployeeNumber
- PaymentDate
- GrossSalary
- Deductions