EXAMINATION	NATIONAL SENIOR CERTIFICATE
GRADE	12
DATE	JUNE 2024
SUBJECT	INFORMATION TECHNOLOGY
PAPER	1
MARK TOTAL	150
DURATION (HOURS)	3
NUMBER OF PAGES	26 plus 2 planning pages

SOUTH AFRICAN COMPREHENSIVE ASSESSMENT INSTITUTE

SUID-AFRIKAANSE KOMPREHENSIEWE ASSESSERINGSINSTITUUT

INSTRUCTIONS TO CANDIDATES

1. In your examination folder, DATAJUN2024 you'll find the folders and incomplete programs for four questions.
2. Rename the folder DATAJUN2024 by adding on (appending) your final examination number e.g. DataJUN2024_62493.
3. You may make backup copies of the folder in a backup folder of your own, **but you should do all your examination work in your renamed** DataJUN2024_XXX folder.
4. YOU ARE REQUIRED TO ANSWER ALL FOUR QUESTIONS.
5. Two blank pages are included at the back of the paper for planning purposes.
6. During the control session you should check all your units, namely **Question1_U, Question2_U, clsQuestion3_U, Question3_U** and **Question4_U.**
    - Can the project open and run?
    - Is your exam number entered as a comment in the unit?
7. You may comment out a specific part of your code if that part doesn'tcompile. All commented code will be marked if there is no alternative code.
8. Notes on question 2:
    - Field names for the tables are included as comments, you may copy and paste these field names in your answers for question 2.
    - Question 2.2.3 is problem solving – you may consider to do this question later.

9. The files that you need to complete this question paper have been provided to you on the disk space allocated to you.

   Once extracted, the following list of files will be available in the folder named **DATAJUN2024:**

Question1:
Fish[1..15].txt
Init_U.pas
Question1_P.dpr
Question1_P.dproj
Question1_P.res
Question1_U.dfm
Question1_U.pas

Question2:
Pets.mdb
PetsBK.mdb
Question2_P.dpr
Question2_P.dproj
Question2_P.res
Question2_U.dfm
Question2_U.pas
TDBConnection_U.pas

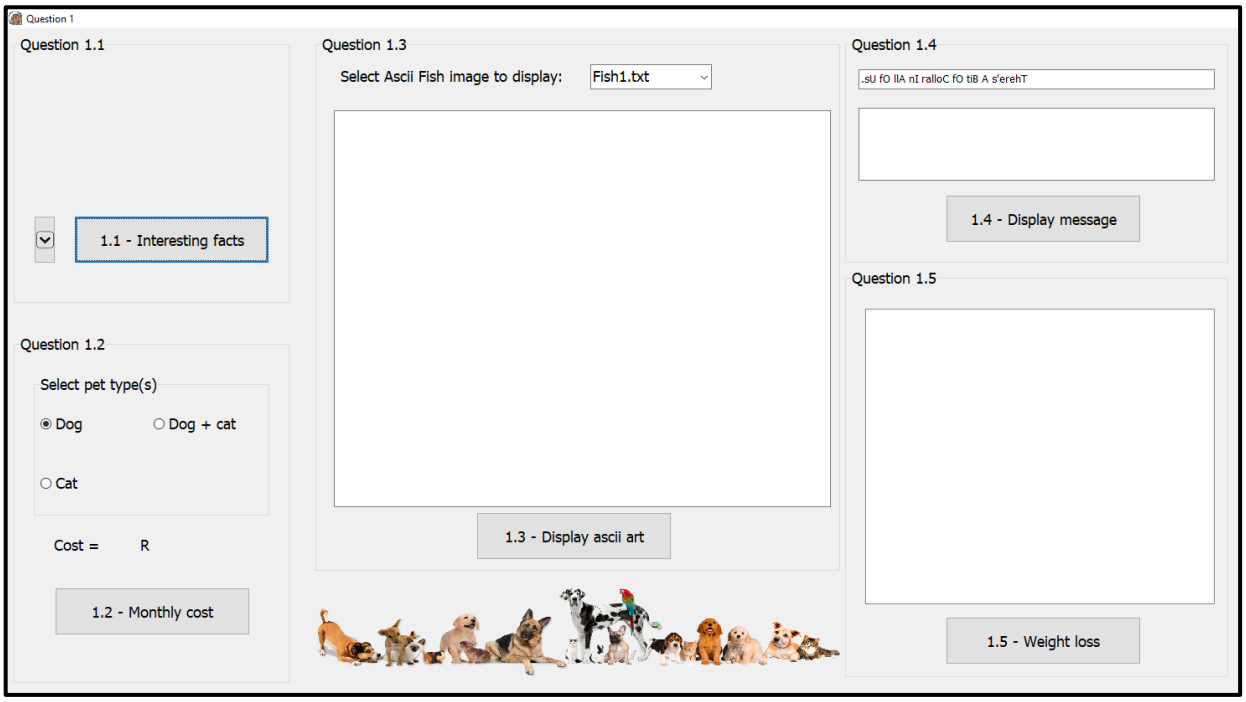
Question3:
clsQuestion3_U.pas
Question3_P.dpr
Question3_P.dproj
Question3_P.res
Question3_U.dfm
Question3_U.pas

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

SECTION A

QUESTION 1: GENERAL PROGRAMMING

Open the incomplete **Question1_p.dpr** program in your **DATAJUN2024/ Question 1** folder. The following screen shot shows the interface:



<u>Question 1.1</u> btnQ1_1 [1.1 - Interesting facts]

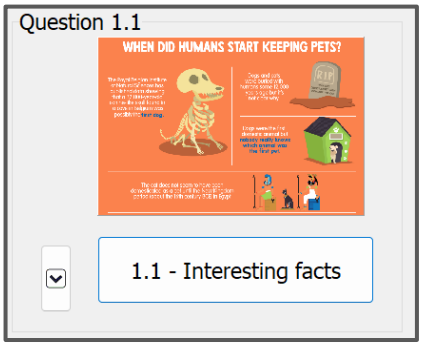
Interesting facts about humans and pets are displayed. The display of the information is very small.

Write code to do the following:

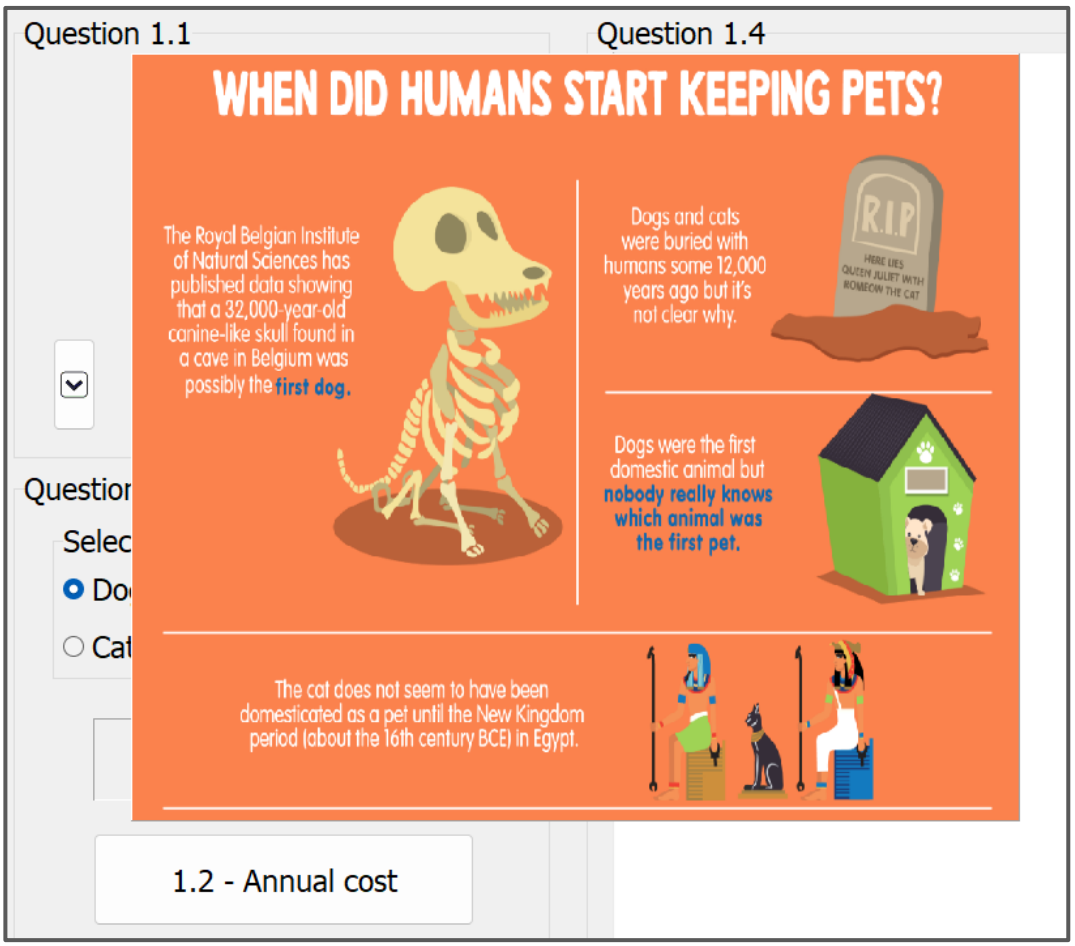
1.1.1 Display the **pnlQ1_1** panel when the **btnQ1_1** is clicked. (1)

1.1.2 Increase the height and width of the **pnlQ1_1** panel with 100 units each time the user clicks on the panel (i.e. the **imgQ1_1** click event handler). (2)

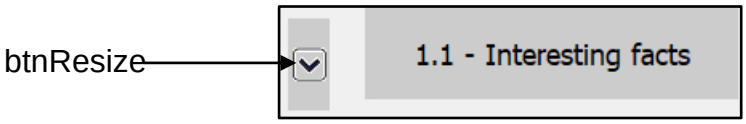
Example of output when the **btnQ1_1** button is clicked:



Example of output when the **imgQ1_1** image is clicked a few times:



NOTE: Code is provided in the **btnResize** event handler to reset the panel to its original size.



Question 1.2 btnQ1_2 [1.2 – Annual cost]

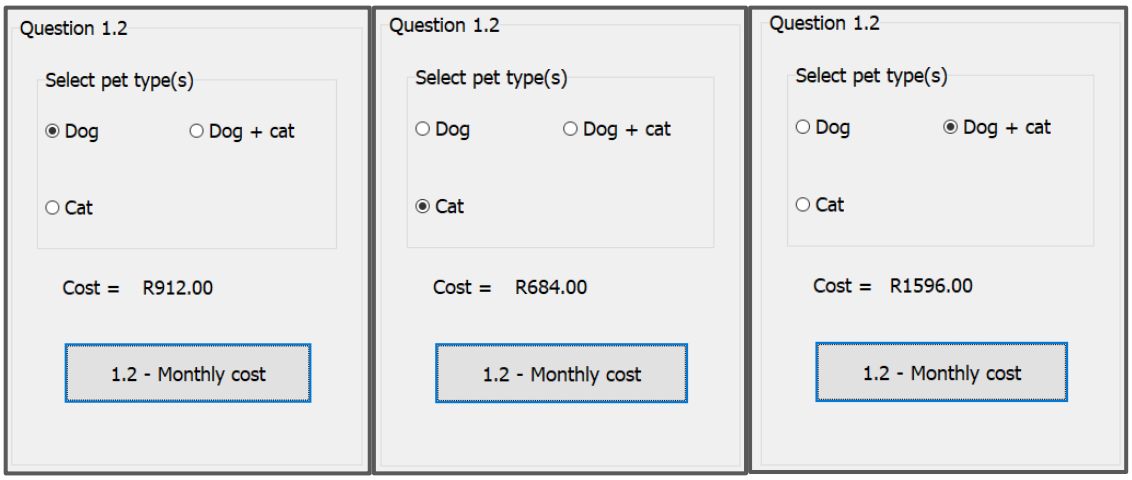
Costs to keep pets can be very expensive. The current estimated cost, to keep a medium sized dog is estimated at R912.00 per month. The cost for a cat is 25% less than that of a dog.

A constant variable to store the cost to keep a dog has been declared.

```
const
  costDog = 912;
```

Use the provided constant variable and complete the code for the **onClick event** of the **btnQ1_2** button to display the monthly cost of the pet selection from the **rgpQ1_2** radio group, on the **lblQ1_2** label.

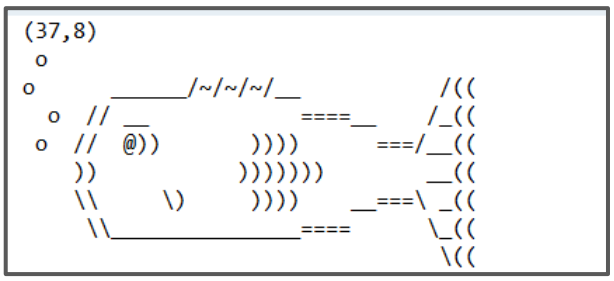
Examples of output:



(6)

Question 1.3 btnQ1_3 [1.3 – Display ascii art]

ASCII art was used to draw pictures of fish. Fifteen (15) pictures are saved in text files namely, `Fish1.txt, Fish2.txt, Fish3.txt` and so forth. The first line in the text file represents the size of the picture only for display purposes. The following screenshot displays the content of the first text file (**Fish1.txt**):

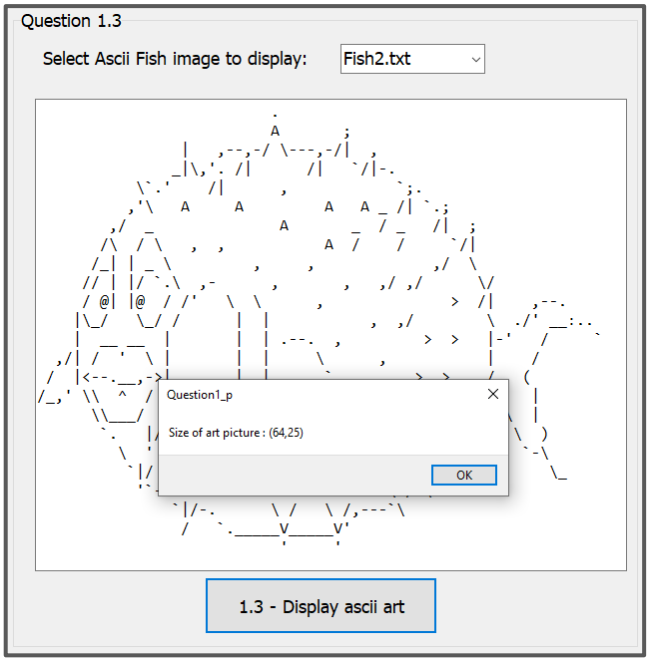


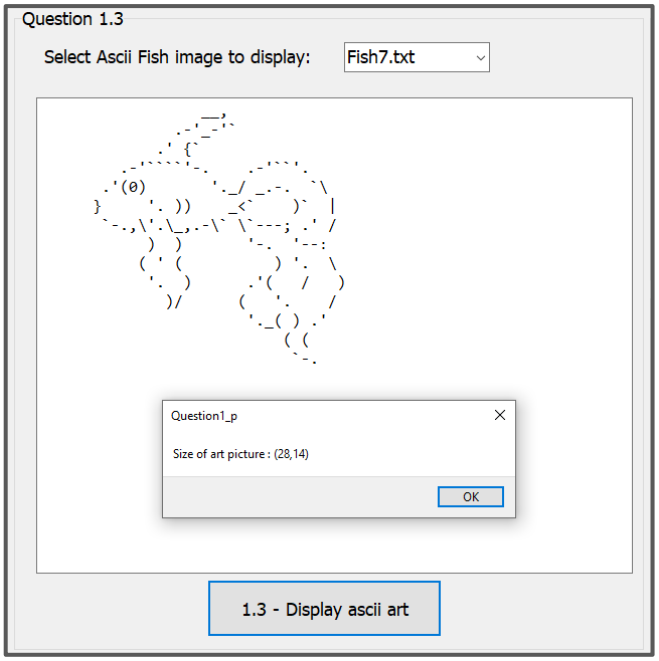
Write code to do the following for **any** of the fifteen pictures:

1.3.1 Clear the rich edit.

1.3.2
- Retrieve the name of the selected file from the **cmbQ1_3** combo box and assign the file name to a variable **sFileName.**
- **TAKE NOTE**: You may assume that the file exists.
- Read and save the first line in the file in the provided variable **sSize**.
- Read and display the rest of the lines in the file in the rich edit.
- Display the saved size (sSize)of the picture using a message box.

Examples of some of the output:





(11)

<u>Question 1.4</u> btnQ1_4 [1.4 – Display message]

An “encrypted” message is displayed in the **edtQ1_4** edit box.

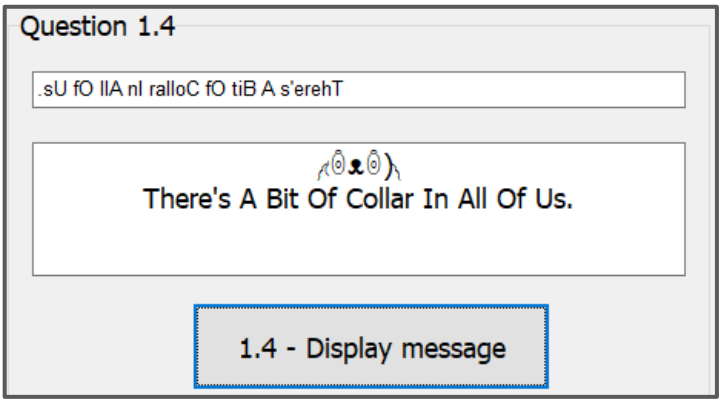
Study the given declarations and code supplied as part of the event.

```
#sPicture : string;  // Given
begin
//Question 1.4
sPicture :=  '┌(ಠ‿ಠ)┐'; // Given
```

Write code to do the following:

1.4.1	Change the font of the edit box **edtQ1_4** to ‘Arial’.	(1)
1.4.2	Retrieve the message from the edit box.	(1)
1.4.3	Reverse the message to read from left to right.	(5)
1.4.4	Set the alignment for the **memQ1_4** memo component to centre.	(1)
1.4.5	Display the provided ‘picture’ ┌(ಠ‿ಠ)┐ in the memo component.	(1)
1.4.6	Display the message in the memo component.	(1)

Example of output:



1.5 **<u>Question 1.5</u> btnQ1_5 [1.5 – Weight loss]**

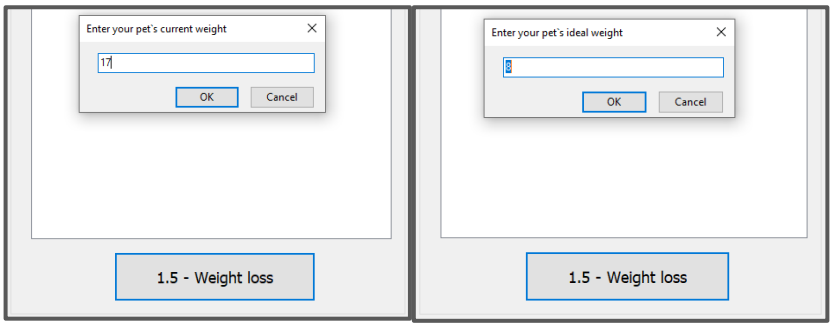
The user's pet is overweight, and the veterinarian said that the pet can lose 0.5 kg in a week if the pet's food intake changes.

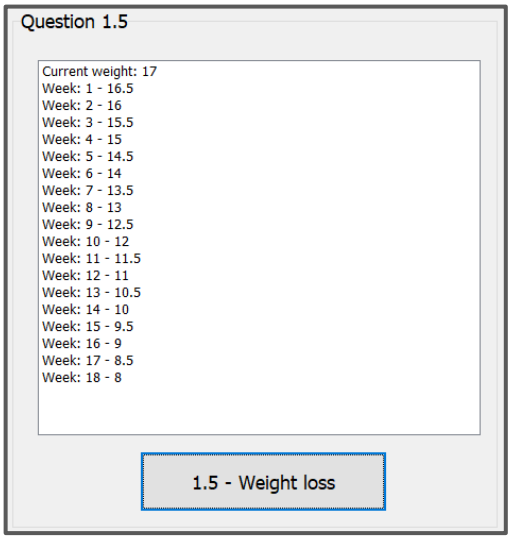
Code has been provided to obtain the value of the current weight of a pet stored in the variable sCurrWeight and the ideal weight stored in the variable sIdealWeight using input boxes.

Complete the code for the onClick event of the **btnQ1_5** button to do the following, based on the given pseudocode below:

```
Initialize a counter to count the number of weeks to get to the ideal weight.
Display the current weight in the lstQ1_6 listbox.
While current weight is greater than the ideal weight do:
        Increment the week counter by 1.
        Reduce the current weight by 0.5.
        Display the week and current weight in the list box.
```

Example of output if the current weight is 17 and the ideal weight is 8:





(10)

- Enter your examination number as a comment in the first line of the program file.
- Save your program.

TOTAL QUESTION 1: [40]

SECTION B

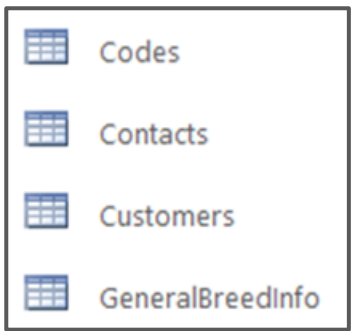
QUESTION 2: DELPHI PROGRAMMING AND DATABASE MANIPULATION

Information on different dog types, dog club contact details and customers, are saved in the Access database **Pets.** Screen shots of the layout and content of the database are shown in Appendix A.

Open the incomplete **Question2_p.dpr** program in your **DATAJUN2024/Question 2** folder.

TAKE NOTE: Field and table names are provided in the code to use in your answer.

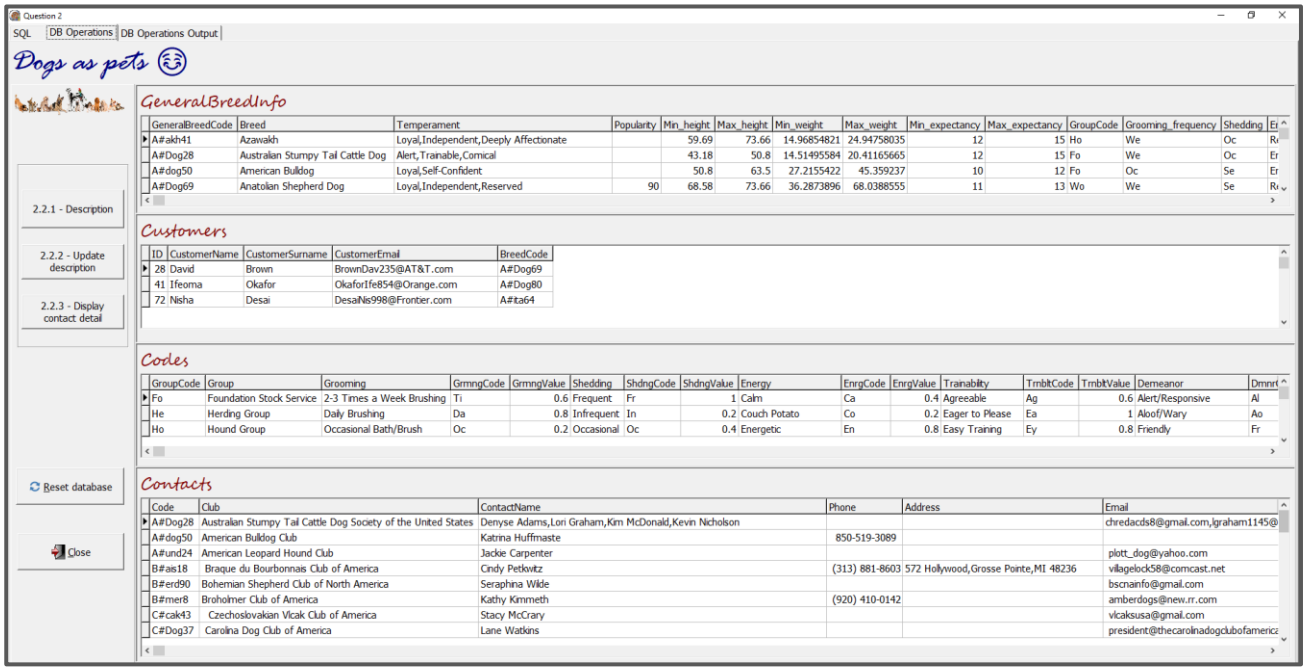
The tables in the DB Pets are listed below. The **same names are used** for reference for the corresponding ADO table component as part of Question 2.2.



E.g.

```
sGroup := Codes['Group'];
// Codes refer to the table and Group to the applicable field.
```

The following screen shot show the interface:



WORK ON THE *tbsQuestion2*_1 TABSHEET

Complete the **SQL** statements for each button of QUESTION 2, as described in QUESTION 2.1.1 to QUESTION 2.1.4.

2.1.1 btnQ2_1_1 [2.1.1 – Clubs]

Write a **SQL** statement to display the club, contact name and email for all the clubs in the Contacts table sorted by the club field in descending order.

An example of output for first five records:

Club	ContactName	Email
United States Mountain Cur Club	Mandy Middleton	walnutridgemtncurs@yahoo.com
The United States Lancashire Heeler Club	Sheryl Bradbury	sherylbradbury@windstream.net
Taiwan Dog Club of America	Atticus Sterling	info@taiwandog.org
Pyrenean Mastiff USA	Valentina Hart	pmusa@pyreneanmastiffclubs.org
Lapponian Herder USA	Heidi Christenson	girln2pooches@yahoo.com

(4)

2.1.2 btnQ2_1_2 [2.1.2 - Popularity]

Write an SQL statement to display the breed and popularity of all the breeds that have a loyal temperament and a popularity index of more than 100.

An example of output for first five records:

Breed	Popularity
Afghan Hound	113
Berger Picard	144
Bedlington Terrier	141
Borzoi	103
Entlebucher Mountain Dog	157

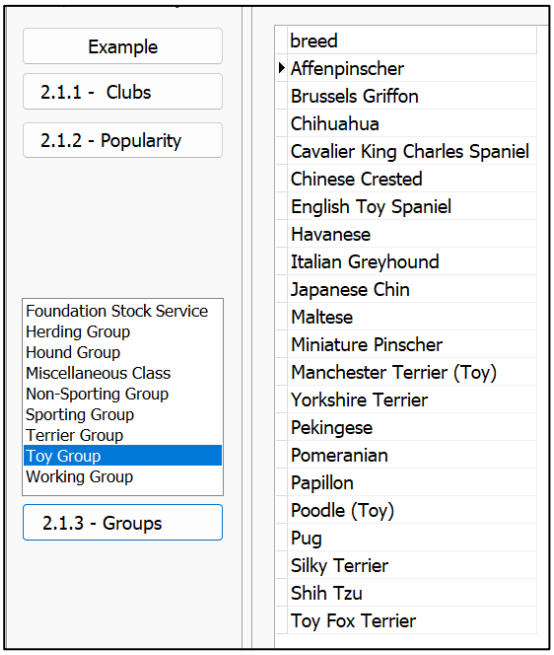
(5)

2.1.3 **btnQ2_1_3 [Groups]**

Write a SQL statement to display the different breeds for the specific characteristics selected in the list box named **lstQ2**.

TAKE NOTE: For this query you must use the GeneralBreedInfo and the Codes table as part of your SQL statement.

An example of the first 17 output records for the “Toy Group”:



(5)

2.1.4 **btnQ2_1_4 [2.1.4 – Update code]**

Write an **SQL** statement to **update** all the American Bulldog’s popularity index to 121 where the code is “A#dog50”.

A message dialogue is provided to confirm the change in the database:

(3)

WORK ON THE *tbsQUESTION2_2* TABSHEET

Use database table statements for each button of QUESTION 2, as described in **QUESTION 2.2.1** and **QUESTION 2.2.3.**

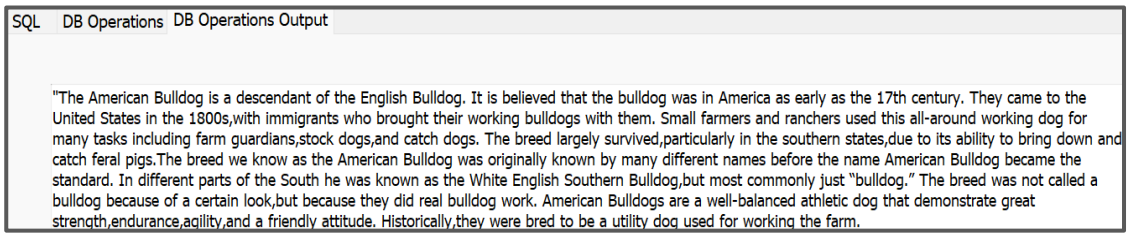
2.2.1 **btnQ2_2_1 [2.2.1 - Description]**

A general description of each dog type is included in the database and stored in the 'Description' field.

Use a database statement to display the description for the dog breed record currently selected (or the current active record) in the GeneralBreedInfo table.

NOTE: Code is provided to display the **DB Operations Output** tabsheet.

Example of output if the American Bulldog is selected in the GeneralBreedInfo table:



(1)

2.2.2 **btnQ2_2_2 [2.2.2 – Update Customer Email]**

The CustomerEmail field in the customers table stores the corresponding email for the customer.

All customers with VerizonFios as an email provider should change to Verizon only.

Use database statements to change the email addresses accordingly.

NOTE:
Code is provided to display:

- The **DB Operations Output** tabsheet.
- A conformation message that the applicable email addresses were updated.

(11)

2.2.3 **btnQ2_2_3 [2.2.3 – Display contact detail]**

Use database statements to display the contact persons and contact details for all the clubs in the database table Contacts. The club name must be bold.

Example of output for first five clubs is shown below:

NOTE: There can be more than one contact person for a specific club. All contact details are stored as a single record per club.

```
Contact details for clubs:
===========================
Australian Stumpy Tail Cattle Dog
Denyse Adams                    chredacds8@gmail.com
Lori Graham                     lgraham1145@gmail.com OR lgraham1145@yahoo.com
Kim McDonald                    breezyacresstumpies@gmail.com
Kevin Nicholson                 nicholson.klnicholson@gmail.com
American Bulldog
Katrina Huffmaste
American Leopard Hound
Jackie Carpenter                 plott_dog@yahoo.com
Braque du Bourbonnais
Cindy Petkwitz                   villagelock58@comcast.net
Bohemian Shepherd
Seraphina Wilde                  bscnainfo@gmail.com
```

(20)

- Enter your examination number as a comment in the first line of the program file.
- Save your program.

TOTAL QUESTION 2: [49]

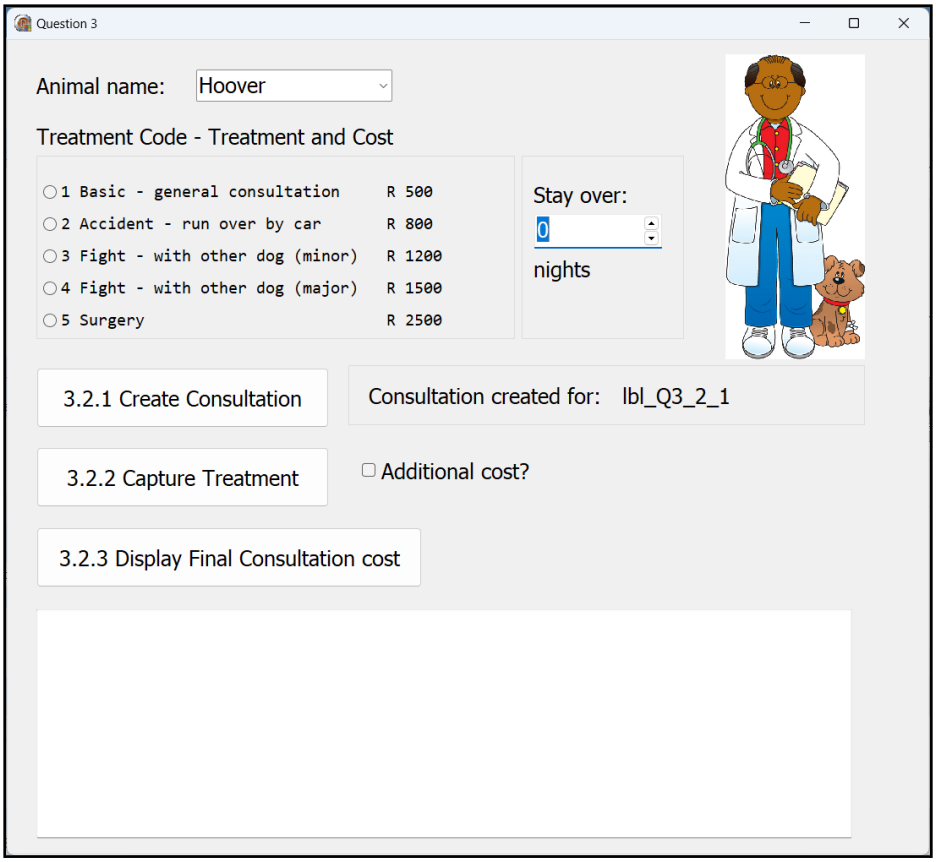
SECTION C

QUESTION 3 (Object orientated programming)

Open the incomplete **Question3_p.dpr** program in your **DATAJUN2024/Question 3** folder.

Visits to the veterinarian is recorded and the costs for an animal patient is calculated based on the treatment, additional costs (if any) and the number of nights of stay.

The following screen shot show the user interface (GUI):



3.1 An incomplete object class is saved in the **clsQuestion3_u** unit.
The following UML class diagram represents the incomplete **TAnimalConsultation** class:

TAnimalConsultation
- fAnimalName: string - fTreatmentDate: string - fTreatmentCode : Integer - fAmountDue: Real
+ create() + getAnimalName: string + setTreatmentCode(Integer) + CalculateBalance(Integer Boolean) + getAmountDue: Real + toString: string

Some notes on the attributes:

The **fAnimalName** field stores the name of the animal.

The **fTreatmentDate** field stores the system date as date of appointment.

The **fTreatmentCode** field stores the choice for the type of treatment as an integer.

```
1 Basic - general consultation     R 500
2 Accident - run over by car       R 800
3 Fight - with other dog (minor)   R 1200
4 Fight - with other dog (major)   R 1500
5 Surgery                          R 2500
```

The given array:

```
arrTreatmentCost: array [1..5] of
Real=(500,800,1200,1500,2500);
```

stores the cost for a type of a treatment.

The **fAmountDue** field stores the amount due after the consultation and treatment have been completed.

Complete the code in the given class (**TAnimalConsultation)** as part of the **clsQuestion3_u** unit as described in QUESTION 3.1.1 to QUESTION 3.1.5.

3.1.1 Change and complete the code for the constructor named **Create** that will receive the animal's name as a parameter and:

(a) Assign the passed name value to the corresponding name field.
(b) Set the **fTreatmentDate** field to the current date.
(c) Set the **fAmountDue** field to 0. (4)

3.1.2 Write the code to create an accessor method named **getAnimalName** that will return the **fAnimalName** field. (2)

3.1.3 Write the code for a mutator method named **setTreatmentCode** that receives an integer value as a parameter to be assigned to the **fTreatmentCode** field. (3)

3.1.4 Write the code to create a method named **CalculateBalance**, to calculate and assign the amount due to the **fAmountDue** field.

This method receives:

- The number of nights, @ R350 per night, that the animal was kept at the clinic. (NumberOfNights)
- A Boolean parameter to indicate that additional costs must be charged.

Use an input box to enter any additional costs as a single amount.
Use the following formula to calculate the amount due:

```
(TreatmentCost (from array) + additional cost + overnight cost)
+ VAT
```

NOTE: The percentage for VAT is provided as a constant variable (rVat = 15). (7)

3.1.5 Write the code to complete the method named **ToString** to return a string in the following format:

```
Animal name: <animal name >
Appointment date: <appointment date>
Amount due: R<amount due>
=============================================
```

TAKE NOTE: The amount due should be formatted as a currency value. (4)

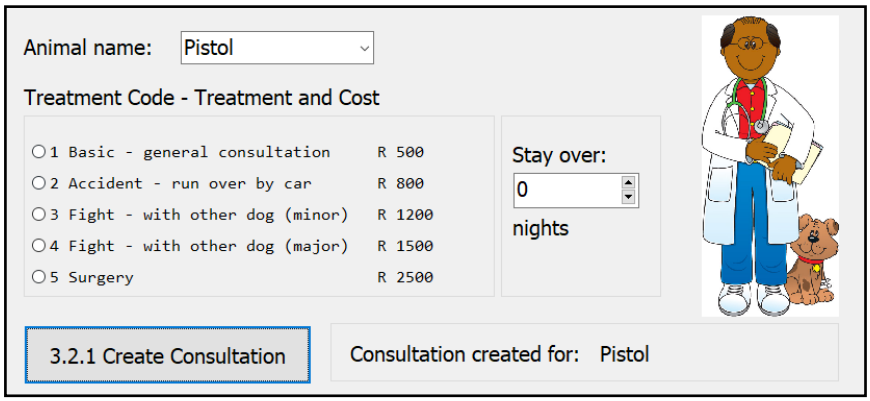
3.2 Complete and code the following for the **Question3_u** main unit:

3.2.1 **btnQ3_2_1 [3.2.1 – Create Consultation]**

Write code to:

- Create an instance of the object **objConsultation** with the animal's name selected in the **cmbQ3_2** combo box.
- Call the appropriate method to return and display the name of the animal on the **lbl_Q3_2_1** label.

Example of output:



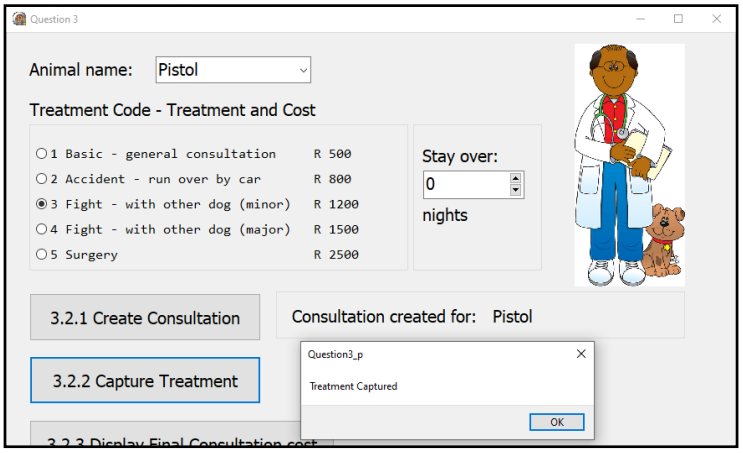
(3)

3.2.2 btnQ3_2_2 [3.2.2 – Capture Treatment]

Write code to:

- Call the applicable method to receive and assign the selected treatment code (1 – 5) from the radio group box.
- Show a message that the treatment has been captured.

Example of output:



(3)

3.2.3 btnQ3_2_3 [3.2.3 – Display Final Consultation cost]

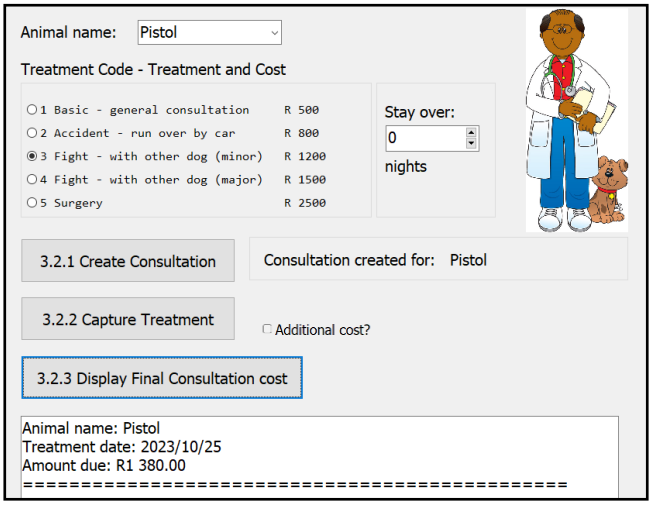
Write code to do the following:

- Call the CalculateBalance method with the status of the check box (checked or not) as argument.
- Populate the redQ3 rich edit with output by calling the appropriate methods.

Study the examples with input and corresponding output.

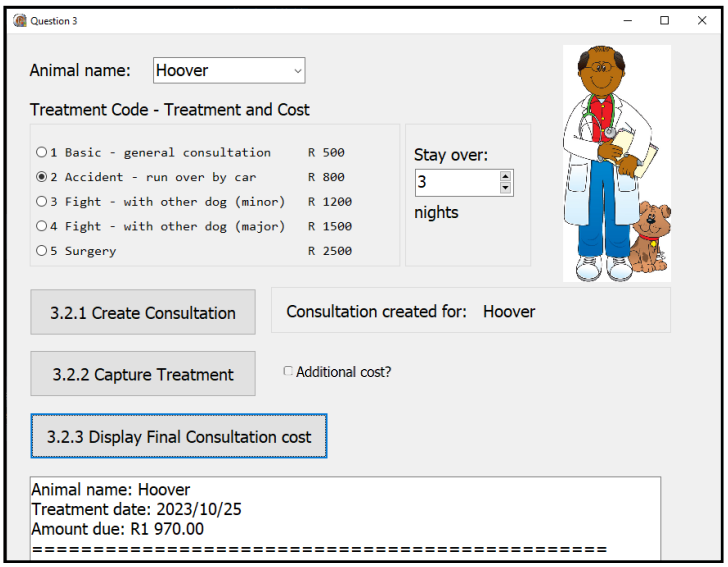
Example 1 - INPUT and OUTPUT:

In this example there was no extra cost, and the animal did not have to stay over.



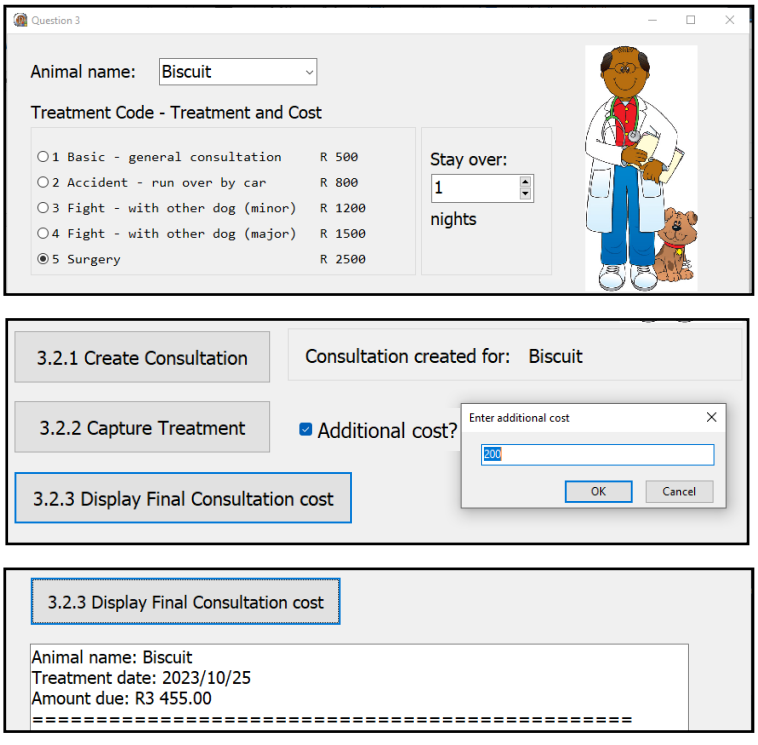
Example 2 - INPUT and OUTPUT:

In this example there was no extra cost, but the animal had to stay over for three nights.



Example 3 - INPUT and OUTPUT:

In this example there was R200 in extra cost, and the animal had to stay over for one night.



(5)

- Enter your examination number as a comment in the first line of the object class and the form class.
- Save your program.

TOTAL QUESTION 3: [31]

SECTION D

QUESTION 4 (Problem solving and arrays)

The *Vigenère* cipher is a polyalphabetic encryption algorithm based on a shift cipher with a keyword.

The Vigenère cipher uses a 26×26 table with A to Z as the row heading and column heading. The column and row headings are used as part of the encryption and decrytion process.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

The following declarations are provided:

```
arrVigenere: array [1 .. 26, 1 .. 26] of char;

arrLetters: array [1 .. 26] of char =
          ('A','B','C','D','E','F','G','H','I','J','K','L','M'
          ,'N','O','P','Q','R','S', 'T','U','V','W','X','Y','Z');
```

A keyword is used to decipher text.

Open the incomplete **Question4_p.dpr** program in your **DATAJUN2024/ Question 4** folder.

4.1 btnQ4_1 [4.1 – Display]

Write the code to display the content of the **arrVigenere** array in the **stgQ4** string grid.

NOTE: Code has been provided for the row and column headings.

Example of output:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

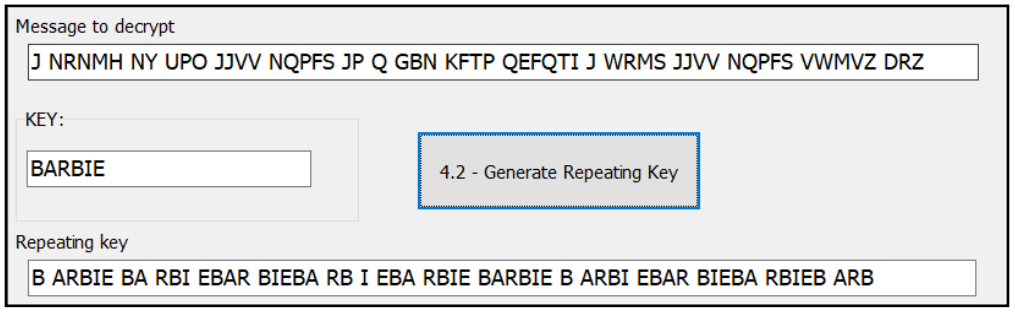
(4)

4.2 btnQ4_2 - [4.2 – Generate Repeating Key]

Code has been provided to display the selected text from the **lstQ4** list box in the **edtQ4_4_1** edit box.

Write the code that will use the keyword in the **edtQ4Key** edit box, to generate and display a repeating key in **edt4_2_2** with the same word length and spaces as the selected text.

Example of output:



(13)

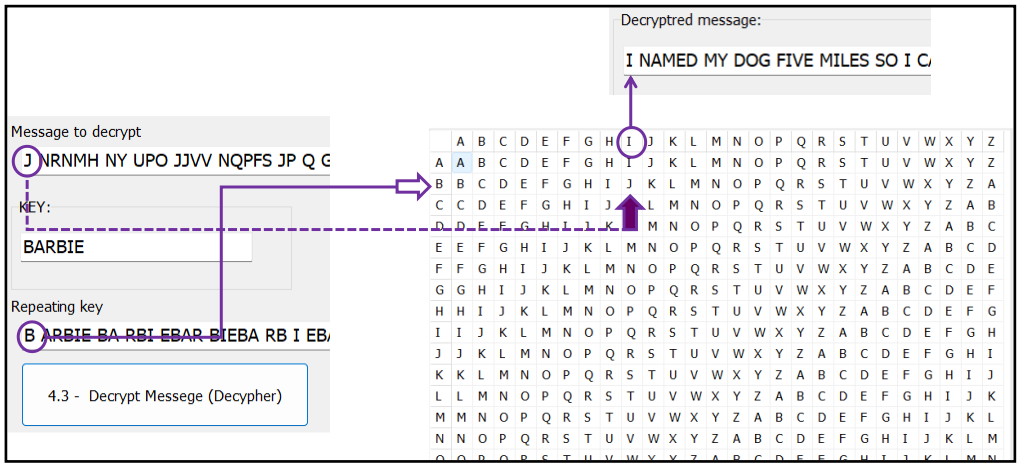
4.3 **btnQ4_3 [4.3 – Decrypt Message (Decipher)]**

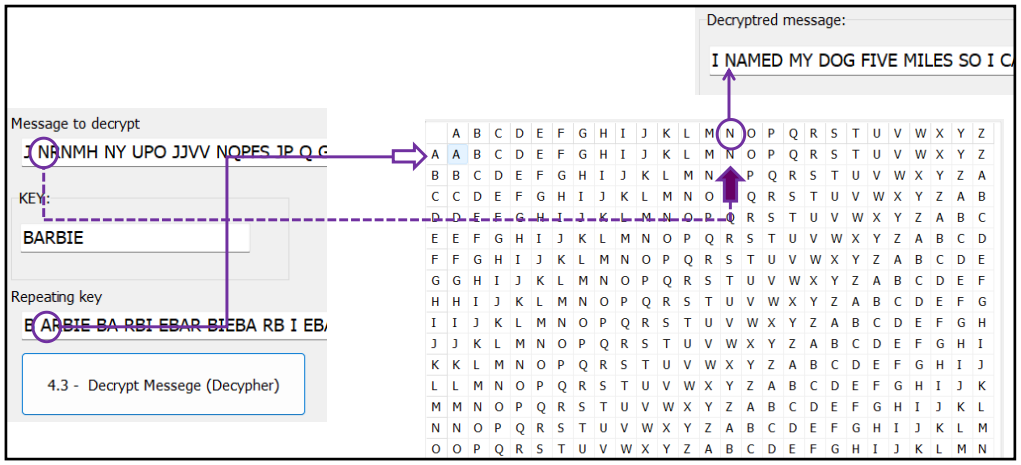
Write the code that will use the repeating key, the selected text and the **arrVigenere** and **arrLetters** arrays to decrypt and display the deciphered text in the **edtQ4_4_3** edit box.

The Vigenére method to decrypt the text is as follows:

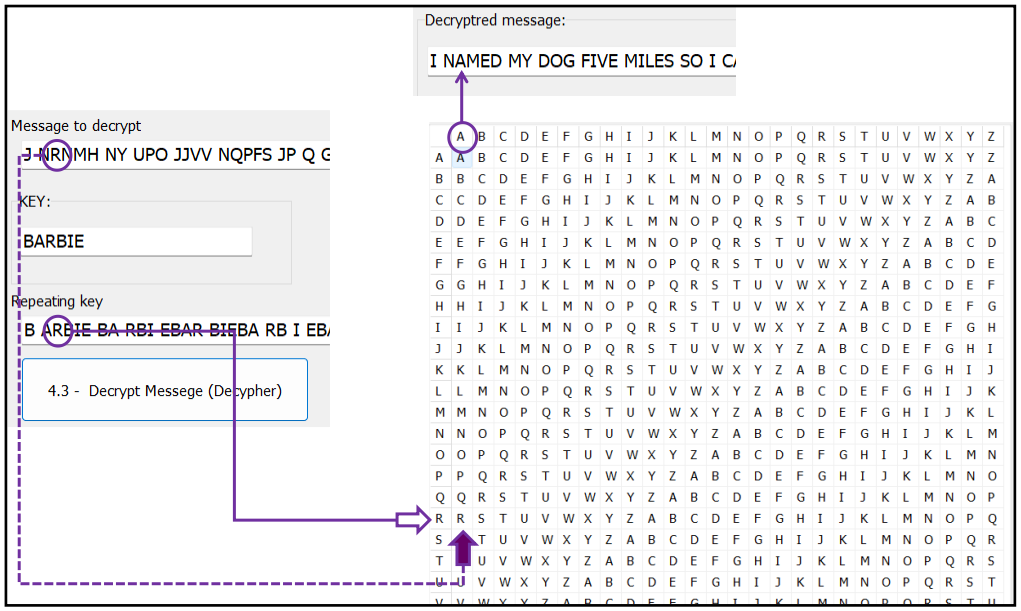
- The repeating key letter represents the row to use.
- Find the corresponding letter of the encrypted text in the specific row.
- The column heading is the decrypted letter.

The following three figures illustrates how the first three characters are decrypted based on the method described above.

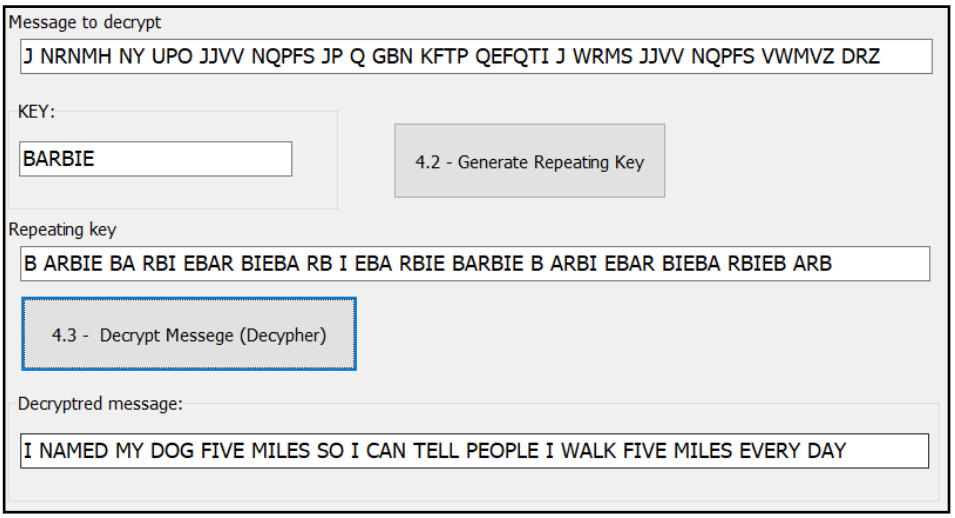




(13)



Example of output:



NOTE: The ASCII value for 'A' is 65, but you may use any solution.

For the second and third messages in the **lstQ4** listbox the key, CHARMANDER and ROOKIE must be used as keywords respectively.

- Enter your examination number as a comment in the first line of the program file.
- Save your program.

TOTAL QUESTION 4: [30]

GRAND TOTAL: [150]

INFORMATION TECHNOLOGY P1

DATABASE INFORMATION OF PETS.MDB FOR QUESTION 2:

The design of the database tables is as follows:

The Pets database consists of four tables i.e. Codes, Contacts, Customers and GeneralBreedInfo.

Design view of the Codes table:

Codes

Field Name	Data Type
GroupCode	Short Text
Group	Short Text
Grooming	Short Text
GrmngCode	Short Text
GrmngValue	Number
Shedding	Short Text
ShdngCode	Short Text
ShdngValue	Number
Energy	Short Text
EnrgCode	Short Text
EnrgValue	Number
Trainability	Short Text
TrnbltCode	Short Text
TrnbltValue	Number
Demeanor	Short Text
DmnrCode	Short Text
DmnrValue	Number

Partial content of the Codes table:

Codes

GroupCode	Group	Grooming	GrmngCode	GrmngValue	Shedding	ShdngCode	ShdngValue	Energy	EnrgCode	EnrgValue	Trainability	TrnbltCode	TrnbltValue	Demeanor
Fo	Foundation Stock Service	2-3 Times a We	Ti	0.60	Frequent	Fr	1.0	Calm	Ca	0.4	Agreeable	Ag	0.60	Alert/Responsiv
He	Herding Group	Daily Brushing	Da	0.80	Infrequent	In	0.2	Couch Potato	Co	0.2	Eager to Please	Ea	1.00	Aloof/Wary
Ho	Hound Group	Occasional Batl	Oc	0.20	Occasional	Oc	0.4	Energetic	En	0.8	Easy Training	Ey	0.80	Friendly
Mi	Miscellaneous Class	Specialty/Profe	Sp	1.00	Regularly	Re	0.8	Needs Lots of A	Ne	1	Independent	In	0.40	Outgoing
No	Non-Sporting Group	Weekly Brushin	We	0.40	Seasonal	Se	0.6	Regular Exercis	Re	0.6	May be Stubbor	Ma	0.20	Reserved with S
Sp	Sporting Group													
Te	Terrier Group													
To	Toy Group													

Design view of the Contacts table:

Contacts

Field Name	Data Type
Code	Short Text
Club	Short Text
ContactName	Short Text
Phone	Short Text
Address	Short Text
Email	Short Text

Partial content of the Contacts table:

Contacts

Code	Club	ContactName	Phone	Address	Email
A#Dog28	Australian Stum	Denyse Adams,Lori Graham,Kim McDonald,Kevin Nicholson			chredacds8@gmail.com,lgraham1145@gmail.com OR lgraham1145@yahoo.com,breezyacresstu
A#dog50	American Bulld	Katrina Huffmaste	850-519-3089		
A#und24	American Leop	Jackie Carpenter			plott_dog@yahoo.com
B#ais18	Braque du Bou	Cindy Petkwitz	(313) 881-8603	572 Hollywood,	villagelock58@comcast.net
B#erd90	Bohemian Shep	Seraphina Wilde			bscnainfo@gmail.com
B#mer8	Broholmer Club	Kathy Kimmeth	(920) 410-0142		amberdogs@new.rr.com
C#cak43	Czechoslovaki	Stacy McCrary			vlcaksusa@gmail.com
C#Dog37	Carolina Dog C	Lane Watkins			president@thecarolinadogclubofamerica.com
D#dog2	Danish/Swedis	Dorothy Herman			InfoFarmdogs.org
D#ond35	Drentsche Patr	Brian O'Connor	208-590-0027		dpcna.gundogs@gmail.com

Design view of the GeneralBreedInfo table:

GeneralBreedInfo

Field Name	Data Type
GeneralBreedCode	Short Text
Breed	Short Text
Temperament	Short Text
Popularity	Number
Min_height	Number
Max_height	Number
Min_weight	Number
Max_weight	Number
Min_expectancy	Number
Max_expectancy	Number
GroupCode	Short Text
Grooming_frequency	Short Text
Shedding	Short Text
Energy_level	Short Text
Trainability	Short Text
Demeanor	Short Text
Description	Long Text

Partial content of the GeneralBreedInfo table:

GeneralBreedInfo

GeneralBreed	Breed	Temperamer	Popularity	Min_height	Max_height	Min_weight	Max_weight	Min_expecta	Max_expecta	GroupCode	Grooming_fr	Shedding	Energy_level	Trainability	Demeanor
A#akh41	Azawakh	Loyal,Independ		59.69	73.66	14.97	24.95	12	15	Ho	We	Oc	Re	In	Fr
A#Dog28	Australian Stumpy Tail Cattl	Alert,Trainable,		43.18	50.80	14.51	20.41	12	15	Fo	We	Oc	En	Ey	Al
A#dog50	American Bulldog	Loyal,Self-Confi		50.80	63.50	27.22	45.36	10	12	Fo	Oc	Se	En	Ag	Al
A#Dog69	Anatolian Shepherd Dog	Loyal,Independ	90	68.58	73.66	36.29	68.04	11	13	Wo	We	Se	Re	In	Re
A#Dog70	American Eskimo Dog	Playful,Perky,Sn	122	22.86	48.26	2.72	15.88	13	15	No	We	Se	En	Ea	Ou
A#Dog80	Australian Cattle Dog	Alert,Curious,Pl	55	43.18	50.80	15.88	22.68	12	16	He	Oc	Re	Ne	Ma	Re
A#erd54	Australian Shepherd	Smart,Work-Ori	17	45.72	58.42	18.14	29.48	12	15	He	We	Se	En	Ea	Al
A#her27	Affenpinscher	Confident,Famc	148	22.86	29.21	3.18	4.54	12	15	To	Ti	Se	Re	Ey	Ou
A#iel20	American Water Spaniel	Eager,Happy,Ch	166	38.10	45.72	11.34	20.41	10	14	Sp	Ti	In	Re	Ea	Fr
A#ier1	Australian Terrier	Affectionate,Cc	140	25.40	27.94	6.80	9.07	11	15	Te	We	In	En	Ag	Fr

Design view of the Curstomers table:

Customers

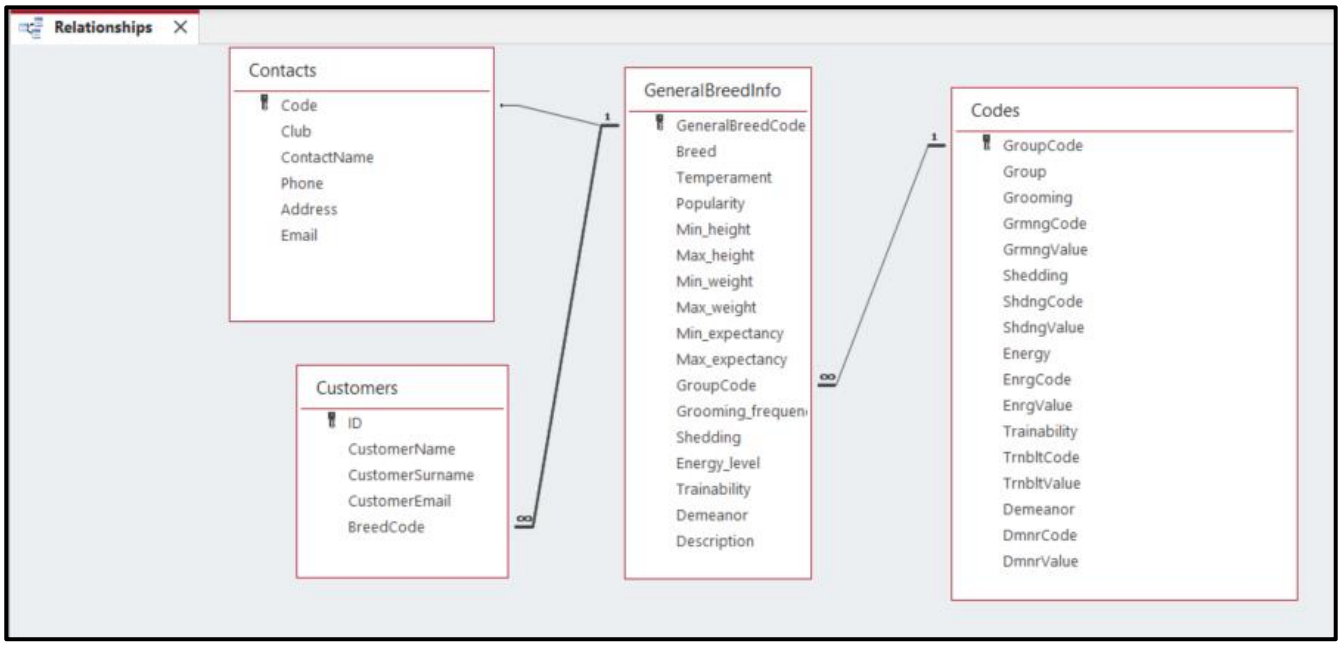
Field Name	Data Type
ID	AutoNumber
CustomerName	Short Text
CustomerSurname	Short Text
CustomerEmail	Short Text
BreedCode	Short Text

Partial content of the Customers table:

Customers

ID	CustomerNa	CustomerSur	CustomerEmail	BreedCode
1	Aarav	Patel	PatelAar951@Gmail.com	S#iel6
2	Abigail	Bell	BellAbi702@Verizon.com	W#sla91
3	Abigail	Hall	HallAbi66@VerizonFios.com	K#Ken28
4	Aditya	Sharma	SharmaAdi99@Optimum.com	M#Cur43
5	Ahmed	Khalid	KhalidAhm907@CenturyLink.com	T#osa71
6	Aisha	Keita	KeitaAis208@Vodafone.com	Y#ika91
7	Aisha	Mansour	MansourAis614@Spectrum.com	I#und68
8	Aisha	Verma	VermaAis885@Rogers.com	E#Dog42
9	Alexander	Clark	ClarkAle841@Shaw.com	T#ier99
10	Ali	Farouk	FaroukAli505@TPG_Tel.com	E#und87

Relationship between tables:



PLANNING PAGE

PLANNING PAGE