SCENARIO

The Africa Zoo houses a large variety of animals. The animals are kept in different enclosures within specific areas of the zoo. The zoo also has an aquarium and a reptile park.

The zoo requires new software for the administration of the various activities at the zoo.

You are required to complete the following THREE questions using the programming language you have studied.

QUESTION 1: PROGRAMMING AND DATABASE

A Microsoft Office Access database named **ZooDB.mdb**, two text files (**tblCarnivores.txt** and **tblVetVisits.txt**) and an incomplete program are provided in the folder named **Question1_XXXX** where XXXX refers to the programming language you have studied.

The design of the tables in the **ZooDB** database and sample data for each table can be found in **ANNEXURE A**.

Do the following:

- Make a backup copy of the **ZooDB** database **BEFORE** you start answering QUESTION 1. You will need a copy of the original database to be able to test your program thoroughly.
- Rename the given folder for QUESTION 1 by replacing the name of the programming language you have studied with your examination number.
- Open the given incomplete program for QUESTION 1.
- Add your examination number as a comment in the first line of the program file.
- Compile and execute the program. The interface displays eight menu options as indicated in the section labelled **QUESTION 1** in **ANNEXURE B (Delphi)/ANNEXURE C (Java)**.

  NOTE:
  - An error message will be displayed if any of the options A–G are selected, due to the incomplete SQL statements.
  - If you experience any problems using the database or connecting to the database, refer to **ANNEXURE D (Delphi)/ANNEXURE E (Java)** for troubleshooting hints.
  - If you still experience database problems, you must nevertheless do the SQL code and submit it for marking. **Marks will only be awarded for the programming code that contains the SQL statements.**

- Complete the code for each menu option by formulating an appropriate SQL statement to display the results for the respective query as described in QUESTIONS 1.1 to 1.7 below.

NOTE: The code to some input statements and the code to execute the SQL statements and display the results of the queries have already been written as part of the given code.

1.1 **Menu Option A**

Display all the details of the animals stored in the **tblCarnivores** table, sorted firstly by the **FamilyName** field and secondly by the **ScientificName** field in **alphabetical order**.

Example of the output of the first five records:

EnclosureNo	FamilyName	ScientificName	GeneralName	NumAdults	NumYoung	EnclosureSize	Endangered
ZE7	Canidae	Canis adustus	Side-striped jackal	6	3	46	LE
ZE8	Canidae	Canis mesomelas	Black-backed jackal	3	3	52	LE
ZF1	Canidae	Lycaon pictus	African wild dog	2	1	60	EN
ZE9	Canidae	Otocyon megalotis	Bat-eared fox	2	3	52	LE
ZE6	Canidae	Vulpes chama	Cape fox	7	4	46	LE

:

(3)

1.2 **Menu Option B**

The user enters the family name of an animal via the keyboard. Display the scientific name, general name, enclosure number and enclosure size for animals belonging to the family name entered by the user, and housed in the ZE area of the zoo.

The **EnclosureNo** field indicates the specific enclosure within an area where the animals are housed, for example in enclosure number ZE5. (ZE represents the area of the zoo and 5 the specific enclosure in that area).

Example of the output of the animals in the *Canidae* family in the ZE area:

ScientificName	GeneralName	EnclosureNo	EnclosureSize
Vulpes chama	Cape fox	ZE6	46
Canis adustus	Side-striped jackal	ZE7	46
Canis mesomelas	Black-backed jackal	ZE8	52
Otocyon megalotis	Bat-eared fox	ZE9	52

(5)

1.3 Menu Option C

Display the categories of endangered species and the total number of animal species in each category housed at the zoo. Use a calculated field with the heading **CountAnimals** for the calculation.

Example of the output:

Endangered	CountAnimals
EN	1
LE	37
NE	1
VU	3

(4)

1.4 Menu Option D

Display the enclosure number and a calculated field showing the space available for each animal of the different mongoose species indicated in the **GeneralName** field.

Calculate the space available per animal by creating a formula which divides the enclosure size by the total number of animals housed in the enclosure. The calculated values should be displayed with a maximum of two decimal digits. Display the calculated field with the heading **SpacePerAnimal**.

HINT: Use the **NumAdults**, **NumYoung** and **EnclosureSize** fields as part of the formula.

Example of the output of the first five records:

EnclosureNo	SpacePerAnimal
ZD1	6.8
ZD2	5.5
ZD3	3.09
ZD4	10.5
ZD5	3.25

:

NOTE: The format of the **SpacePerAnimal** field may differ from the given example. (6)

1.5 Menu Option E

Increase the number of young animals in the ZF1 enclosure by 3.

If the updating of the record was done successfully, an output message, stating that the record was successfully processed, will be displayed. (4)

1.6 **Menu Option F**

The user is requested to enter the day of the month when the veterinarian (vet) visited the animals, for example 23.

Display the enclosure number, the general name, the date of the visit, the IDs of the particular animals visited and the reason why each of the animals was visited on the specific day of the month entered by the user.

HINT: Use the **DAY()** SQL function in the SQL statement.

Example of the output of the first five records for 23 September 2012:

EnclosureNo	GeneralName	VisitDate	Animal_ID	ReasonForVisit
ZF1	African wild dog	2012-09-23	ZF1_5	Assisted with birth
ZD3	Cape grey mongoose	2012-09-23	ZD3_3	Skin problem
ZD8	Selous' mongoose	2012-09-23	ZD8_1	Routine check-up
ZE6	Cape fox	2012-09-23	ZE6_5	Routine check-up
ZE4	Brown hyena	2012-09-23	ZE4_5	Routine check-up

:

NOTE: The format of the dates in the **VisitDate** field may differ from the given example. (7)

1.7 **Menu Option G**

On 25 September 2012 the veterinarian examined the animal with the ID ZD5_3 in enclosure ZD5 for an ear infection. He indicated that a follow-up visit would be required. Add this data as a new record into the **tblVetVisits** table.

If the record was added to the table successfully, an output message stating that the record was processed successfully will be displayed.

HINT: Use option F to verify the adding of the record to the database. Use 25 September 2012 as input data. (6)

- Enter your examination number as a comment in the first line of the file containing the SQL statements.
- Save your program.
- A printout of the code will be required.

[35]