Cambridge

OL- IGCSE

Computer science

CODE: (0478)

Chapter 09

Databases

9.1 Databases

9.1.1 Single-table databases

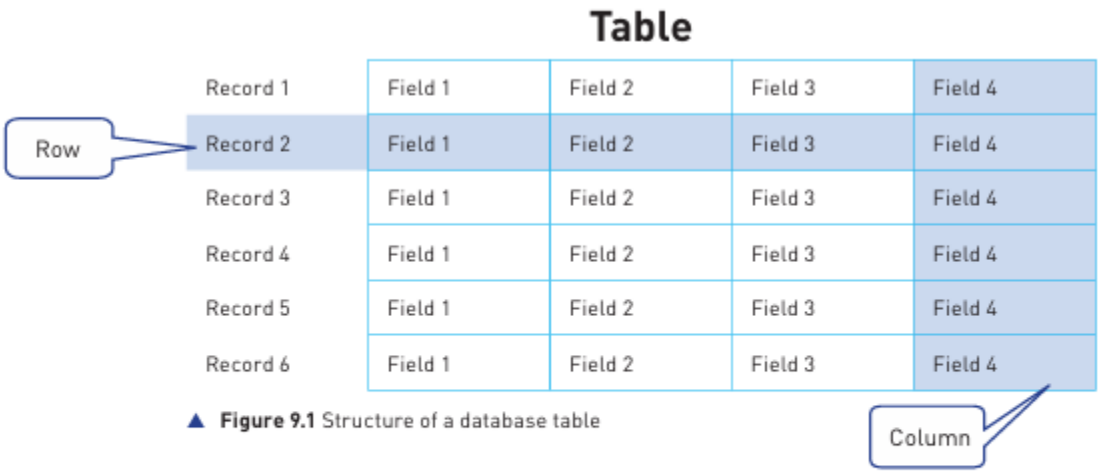
A **database** is a structured collection of data that allows people to extract information in a way that meets their needs. The data can include text, numbers, pictures; anything that can be stored in a computer. Relational databases will be studied at A Level but for IGCSE only single-table databases will be studied. A **single-table database** contains only one table.

Why are databases useful? Databases prevent problems occurring because:
» If any changes or additions are made it only has to be done once – data is consistent
» The same data is used by everyone
» Data is only stored once in relational databases which means no data duplication.

What are databases used for? To store information about people, for instance:
» Patients in a hospital
» Pupils at a school. To store information about things, for instance:
» Cars to be sold
» Books in a library. To store information about events, for instance:
» Hotel bookings
» Results of races.

Fields and records – the building blocks for any database

Inside a database, data is stored in tables, which consists of many records. Each record consists of several fields. The number of records in a table will vary as new records can be added and deleted from a table as required.



▲ **Figure 9.1** Structure of a database table

A table contains data about one type of item or person or event, and will be given a meaningful name, for example:

- a table of patients called **PATIENT**
- a table of books called **BOOK**
- a table of doctor's appointments called **APPOINTMENT**.

As every record contains the same number of fields, each field in a record contains a specific piece of information about the single item, person or event stored in that record. Each field will have a meaningful name to identify the data stored in it.

Validation

The role of validation was discussed in Section 7.5. It may be worth the reader revisiting this part of the book before continuing with this chapter. Some validation checks will be automatically provided by the database management software that is used to construct and maintain the database.

The PATIENT table structure could look like this:

PATIENT Table

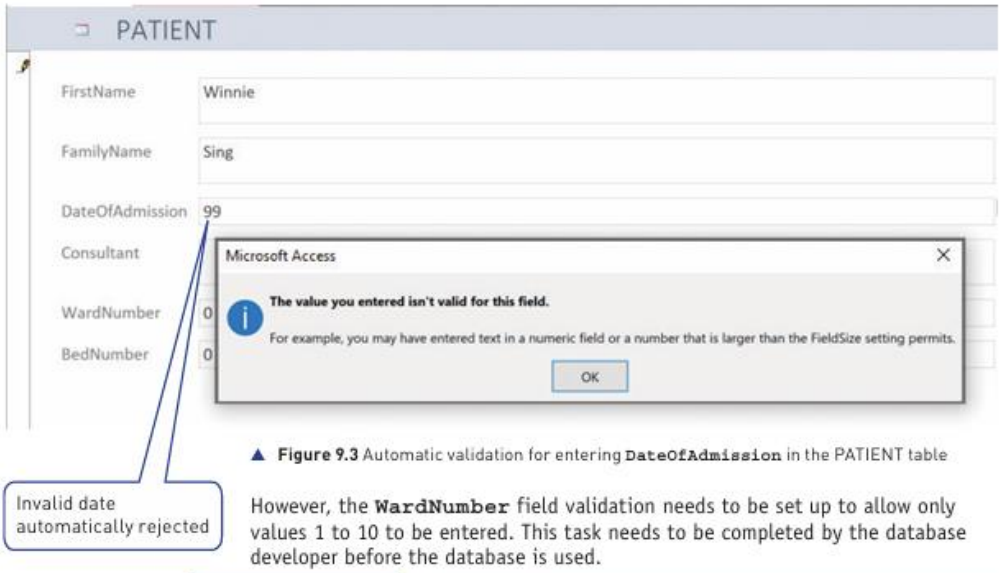
Record 1	FirstName	FamilyName	DateOfAdmission	Consultant	WardNumber	BedNumber
Record 2	FirstName	FamilyName	DateOfAdmission	Consultant	WardNumber	BedNumber
Record 3	FirstName	FamilyName	DateOfAdmission	Consultant	WardNumber	BedNumber
Record 4	FirstName	FamilyName	DateOfAdmission	Consultant	WardNumber	BedNumber
Record 5	FirstName	FamilyName	DateOfAdmission	Consultant	WardNumber	BedNumber
Record 6	FirstName	FamilyName	DateOfAdmission	Consultant	WardNumber	BedNumber

▲ **Figure 9.2** Structure of the PATIENT table

» For the table called BOOK the fields could include:
- Title of the book called `Title`
- Author of the book called `Author`
- `ISBN`, etc.

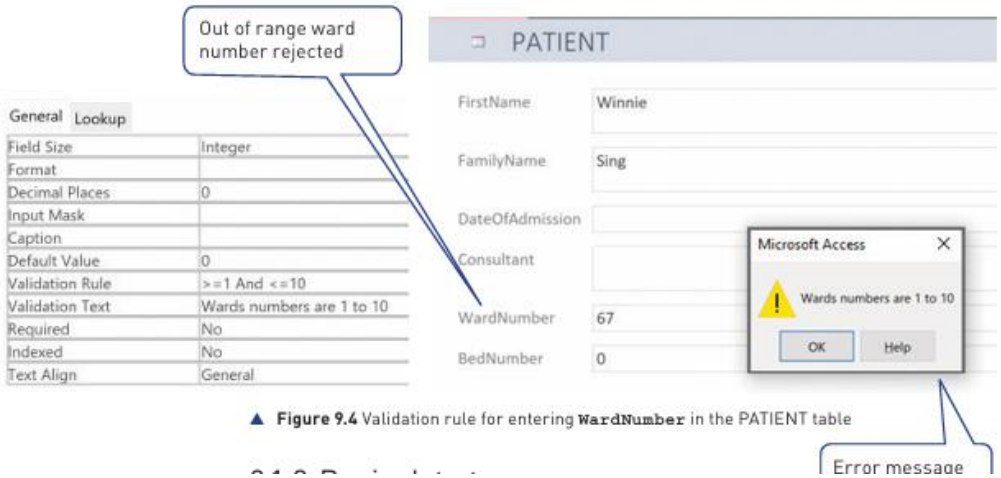
Other validation checks need to be set up by the database developer during the construction of the database. The practical use of a database management system is strongly recommended for all students. Practical examples will be used throughout this chapter.

The database management software used is Microsoft Access 365 as Microsoft Access is used by most schools with students studying IGCSE Computer Science.



▲ **Figure 9.3** Automatic validation for entering `DateOfAdmission` in the PATIENT table

However, the `WardNumber` field validation needs to be set up to allow only values 1 to 10 to be entered. This task needs to be completed by the database developer before the database is used.



▲ **Figure 9.4** Validation rule for entering `WardNumber` in the PATIENT table

9.1.2 Basic data types

There are six basic data types that you need to be able to use in a database:

» Text/alphanumeric » Character » Boolean » Integer
» Real » Date/time

What is a data type?

Each field will require a **data type** to be selected. A data type classifies how the data is stored, displayed and the operations that can be performed on the stored value. For example, a field with an integer data type is stored and displayed as a whole number and the value stored can be used in calculations.

These database data types are specified in the syllabus. They are available to use as *Access* data types, but the names *Access* uses may be different from the terms in the syllabus.

Syllabus data type	Description	Access data type
text/alphanumeric	A number of characters	short text/long text
character	A single character	short text with a field size of one
Boolean	One of two values: either True or False, 1 or 0, Yes or No	Yes/No
integer	Whole number	number formatted as fixed with zero decimal places
real	A decimal number	number formatted as decimal
date/time	Date and/or time	Date/Time

9.1.3 Primary keys

As each record within a table contains data about a single item, person, or event, it is important to be able to uniquely identify this item. In order to reliably identify an item from the data stored about it in a record there needs to be a field that uniquely identifies the item. This field is called the primary key.

9.1.4 SQL

Structured Query Language (SQL) is the standard query language for writing scripts to obtain useful information from a database. We will be using SQL to obtain information from single-table databases.

SQL scripts

An SQL script is a list of SQL commands that perform a given task, often stored in a file so the script can be reused.

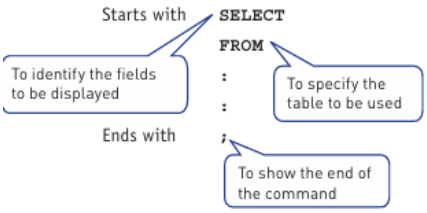
In order to be able to understand SQL and identify the output from an SQL script, you should have practical experience of writing SQL scripts. You can write scripts using SQL commands in Access.

There are many other applications that also allow you to do this – MySQL and SQLite are freely available ones. When using any SQL application, it is important that you check the commands available to use as these may differ slightly from those listed in the syllabus and shown below.

You will need to be able to understand and identify the output from the following SQL statements.

SQL Query Statement	Description
`SELECT`	Fetches specified fields (columns) from a table; queries always begin with `SELECT`.
`FROM`	Identifies the table to use.
`WHERE`	Includes only records (rows) in a query that match a given condition.
`ORDER BY`	Sorts the results from a query by a given column either alphabetically or numerically.
`SUM`	Returns the sum of all the values in a field (column). Used with `SELECT`.
`COUNT`	Counts the number of records (rows) where the field (column) matches a specified condition. Used with `SELECT`.

An SQL command:



Only the `SELECT` and `FROM` commands are mandatory in an SQL script. All other commands are optional.

A **SELECT** statement takes the form:

`SELECT Field1, Field2, Field3`, etc. – this specifies the individual fields (columns) to be shown.

`SELECT *` – this specifies that **all** fields (columns) are to be shown.

A **FROM** statement takes the form:

`FROM TableName` – this specifies the table to use.

A **WHERE** statement takes the form:

`WHERE Condition` – this specifies the condition to apply.

Conditions often include values from fields, these values need to be stated in a form that matches the data type for the field.

Field type	Example value	General notes	Access notes
text	`'Mr Smith'`	Text field values should be in enclosed in single quotation marks.	Double quotation marks can also be used.
character	`'M'`	Character field values should be in enclosed in single quotation marks.	Double quotation marks can also be used.
Boolean	`TRUE`	Boolean can be `TRUE` or `FALSE`	Data type is Yes/No
integer	`12`	Integer field values should be whole numbers.	Allows integer or decimal values.
real	`12.01`	Real field values should be decimal numbers.	Allows integer or decimal values.
Date/time	`'22/11/2022'`	Date/time field values should be in enclosed in single quotation marks.	Date/time field values **must** be in enclosed in hashes (**#**).

Conditions also require operators to compare values from fields.

Operator	Description
`=`	equal to
`>`	greater than
`<`	less than
`>=`	greater than or equal to
`<=`	less than equal to
`<>`	not equal to
`BETWEEN`	between a range of two values
`LIKE`	search for a pattern
`IN`	specify multiple values
`AND`	specify multiple conditions that must all be true
`OR`	specify multiple conditions where one or more conditions must be true
`NOT`	specify a condition that must be false

An **ORDER BY** statement takes the form:

`ORDER BY Field1, Field2`, etc. – this specifies a sort in ascending or alphabetical order starting with the first field.

`ORDER BY Field1, Field2 DESC` – this specifies a sort in descending or reverse alphabetical order starting with the first field.

A **SUM** statement takes the form:

SELECT SUM (Field) – this specifies the field (column) for the calculation. The field should be integer or real.

A **COUNT** statement takes the form:

SELECT COUNT (Field) – this specifies the field (column) to count if the given criterium is met.

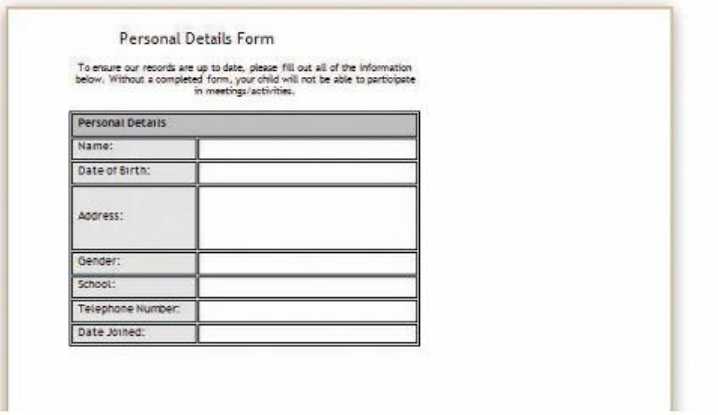
Practical use of a database

As an IGCSE Computer Science student you need to be able to do the following:

» Define a single-table database from given data storage requirements

» Choose a suitable primary key for a database table

» Read, complete and understand SQL scripts.



Personal Details Form

To ensure our records are up to date, please fill out all of the information below. Without a completed form, your child will not be able to participate in meetings/activities.

Personal Details	
Name:	
Date of Birth:	
Address:	
Gender:	
School:	
Telephone Number:	
Date Joined:	

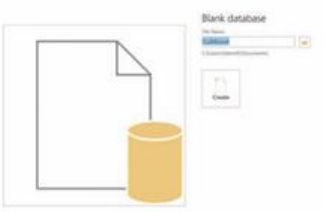
▲ **Figure 9.7** Enrolment form

Define a single-table database from given data storage requirements and choose a suitable primary key

To create the cub scout database, open *Access* and select the **Blank database** template.

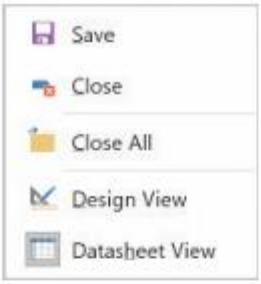
▲ **Figure 9.8** Blank database template

Then type the Filename **CubScout** and click the create button.



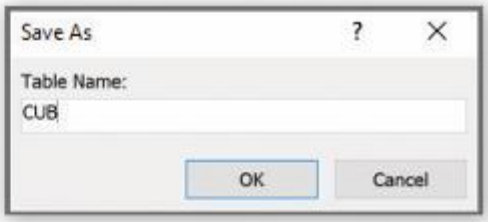
▲ **Figure 9.9** Creating the CubScout database

Select the table design view...



▲ **Figure 9.10** Design view

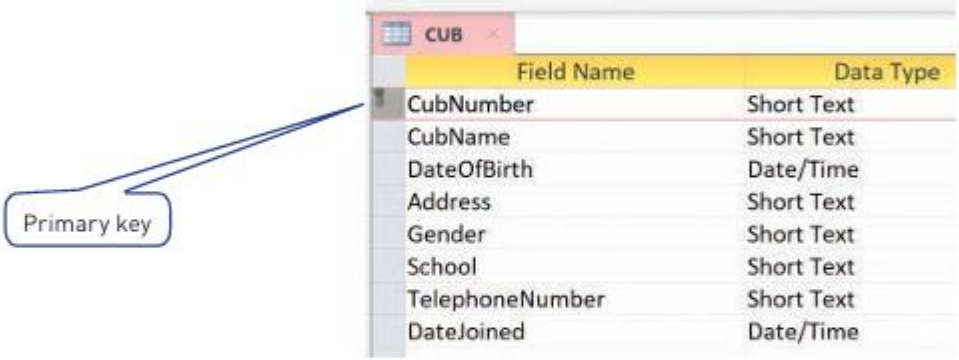
....and name the table **CUB**.



▲ **Figure 9.11** Naming the table

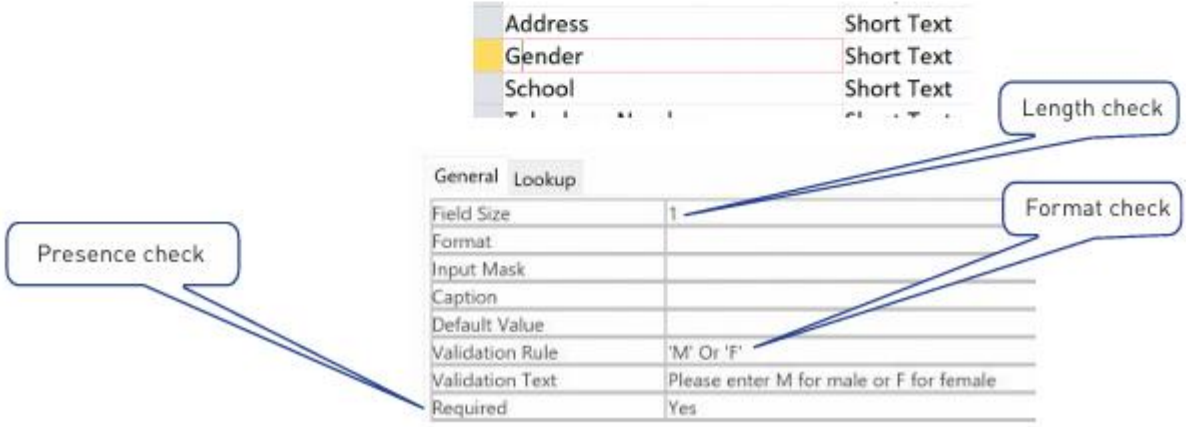
Set up the fields to match the data collection form in Figure 9.7 and include an extra field for a primary key.

Each field will require a meaningful name and a suitable data type must be selected.



▲ **Figure 9.12** Fields for the CUB table

Validation checks need to be built in for each field, for example the **Gender** field.



▲ **Figure 9.13** Validation rules for **Gender** field

Revision questions

1. A database, SOFASELECT, was set up to show the prices of suites, sofas and chairs for sale from an online furniture warehouse. Part of the database is shown below.

Description	Brochure Number	Number of Seats	Number of Pieces	Material	Colour	Price in $
Sofa	SF17	2	1	Leather	Red	950
Sofa	SF19	3	1	Vinyl	Black	1,000
Suite	SU10	4	3	Velvet	Green	1,500
Suite	SU23	5	3	Leather	Brown	950
Recliner chair	RC01	1	1	Leather	Cream	600
Chair	CH16	1	1	Vinyl	Red	250
Recliner sofa	RS23	4	1	Leather	Cream	1,200
Chair	CH10	1	1	Velvet	Red	175

(a) How many fields are in each record?

(b) State which field you would choose for the primary key

Give a reason for choosing this field.

(c) State the data type you would choose for each of the following fields

(d) The query-by-example grid below selects all the furniture in cream leather.

Field:	Description	Material	Colour	Price in $	Brochur
Table:	SOFASELECT	SOFASELECT	SOFASELECT	SOFASELECT	SOFASELEC
Sort:				Descending	
Show:	☑	☐	☐	☑	☑
Criteria:		= 'Leather'	= 'Cream'		
or:					

Show the output from the query-by-example.

(e) Complete the query-by-example grid below to select and show the brochure number, material, colour and price of all the furniture with 3 or more seats.

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

2. A database table, SHEEP, is used to keep a record of the sheep on a farm. Each sheep has a unique ear tag, EARnnnn; n is a single digit. The farmer keeps a record of the date of birth, the gender and the current weight of each sheep in kilograms.

(a) Identify the four fields required for the database. Give each field a suitable name and data type. Provide a sample of data that you could expect to see in the field.

Field 1 name .....

Data type .....

Data sample .....

Field 2 name .....

Data type .....

Data sample .....

Field 3 name .....

Data type .....

Data sample .....

Field 4 name .....

Data type .....

Data sample .....

(b) State the field that you would choose as the primary key.

(c) Using the query-by-example grid below, write a query to identify the ear tags of all male sheep weighing over 10 kilograms. Only display the ear tags.

Field:				
Table:				
Sort:				
Show:	☐	☐	☐	☐
Criteria:				
or:				

3. A garden centre sells garden tools and stores details of these in a database table named TOOLS. Code is the primary key in the TOOLS table.

Code	Description	Price ($)	Quantity_Stock	Quantity_Ord
GFLG	Garden Fork	50.00	1	50
GSLG	Garden Spade	50.00	11	0
GHLG	Garden Hoe	45.00	8	0
HFSM	Hand Fork	9.99	42	0
HSSM	Hand Spade	9.99	40	0
HWSM	Hand Weeder	9.99	11	0
HS20	Hose (20 metres)	45.00	10	0
HS35	Hose (35 metres)	60.00	2	0
HS50	Hose (50 metres)	75.00	20	60
YBLG	Yard Brush	24.99	100	0
LMHD	Lawn Mower	99.99	5	0
LMBT	Lawn Mower (Battery)	249.99	7	0
LMPT	Lawn Mower (Petrol)	349.99	10	25
TRBT	Edge Trimmer (Battery)	79.99	15	0
TRPT	Edge Trimmer (Petrol)	59.99	20	0
SHSM	Shears	40.00	40	0
HCSM	Hedge Clippers	40.00	45	0

a) State the purpose of the primary key in the TOOLS table.

(b) List the output from the data shown in the table TOOLS that would be given by this query-by example.

Field:	Code	Description	Price ($)	Quantity_Stock	Quantity_Or
Table:	TOOLS	TOOLS	TOOLS	TOOLS	TOOLS
Sort:					Descending
Show:	☑	☑	☐	☐	☑
Criteria:			>40	>0	>0
or:					

(c) Complete the query-by-example grid to output the tools where the quantity in stock is below 25. Only show the Code, Description and Quantity_Stock fields in ascending order of Code.

Field:				
Table:				
Sort:				
Show:	☐	☐	☐	☐
Criteria:				
or:				

4. A database table, JUICE, is used to keep a record of cartons of fresh fruit juice available for sale.

Juice code	Fruit 1	Fruit 2	Size	Volume (ml)	Sto
LMO10	Mango	Orange	Large	1000	18
MOO05	Orange	Orange	Medium	500	8
SAM02	Apple	Mango	Small	200	25
SAA02	Apple	Apple	Small	200	50
SPP02	Pineapple	Pineapple	Small	200	10
MMM05	Mango	Mango	Medium	500	12
LMM10	Mango	Mango	Large	1000	5
MGG05	Guava	Guava	Medium	500	5
SMO02	Mango	Orange	Small	200	7
MOP05	Orange	Pineapple	Medium	500	12
LAA10	Apple	Apple	Large	1000	32
SGO02	Guava	Orange	Small	200	10
LPP10	Pineapple	Pineapple	Large	1000	3
LOO10	Orange	Orange	Large	1000	25
SOO02	Orange	Orange	Small	200	40

(a) Identify a suitable field to use as the primary key. State a reason for your choice.

(b) Complete the query-by-example grid to display only the stock level and size of all cartons containing only apple juice.

Field:				
Table:				
Sort:				
Show:	☐	☐	☐	☐
Criteria:				
or:				

5. Draw the flowchart symbol for Decision and the flowchart symbol for Process

Decision	Process

6. A convenience store which sells general groceries wants to set up a database table called STOCK. The table will contain fields including a description of the item, the price of the item and the number in stock for each item. The STOCK table also has a fourth field to be used as a primary key

a. Complete the table to suggest a suitable field name for each of the four fields in the table STOCK. Give the purpose of the data to be stored in each field.

Field name	Purpose of field contents

b. Complete the query-by-example grid to output stock items where the quantity in stock has fallen below 20. Only show the primary key and description of the items.

Field:				
Table:				
Sort:				
Show:	☐	☐	☐	☐
Criteria:				
or:				

7. A wildlife park has a database table, called LIVESTOCK, to classify and record its animal species. Part of the database table is shown.

(a) Suggest another appropriate field that could be added to this database by stating its name and data type. State its purpose and give an example of the data it could contain.

Species	Classification	Diet	Legs
Giraffe	Mammal	Herbivore	4
Elephant	Mammal	Herbivore	4
Crocodile	Reptile	Carnivore	4
Ostrich	Bird	Omnivore	2
Gorilla	Mammal	Herbivore	2
Bear	Mammal	Omnivore	4
Rhinoceros	Mammal	Herbivore	4
Hippopotamus	Mammal	Herbivore	4
Flamingo	Bird	Omnivore	2
Lion	Mammal	Carnivore	4
Turtle	Reptile	Omnivore	4
Penguin	Bird	Carnivore	2

(b) Use the query-by-example grid below to provide a list of all four legged mammals that are herbivores, sorted alphabetically by species, with only the species displayed

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

8. A database table, TRAIN, is to be set up for a railway company to keep a record of the engines available for use. Each engine has a unique number made up of 5 digits, nnnnn. The engines are classified as freight (F) or passenger (P) together with a power classification that is a whole number between 0 and 9, for example F8. The railway company keeps a record of the date of the last service for each engine.

(a) Identify the three fields required for the database. Give each field a suitable name and data type. Provide a sample of data that you could expect to see in the field.

(b) State the field that you should choose as the primary key.

(c) Using the query-by-example grid below, write a query to identify all passenger engines that have not been serviced in the past 12 months. Only display the engine numbers.

Field:				
Table:				
Sort:				
Show:	☐	☐	☐	☐
Criteria:				
or:				