Databases

Explain the terms in the table below.

|  |
| --- |
| Data |
|  |
| Information |
|  |
| Database |
|  |
| Column/Field |
|  |
| Row/Record |
|  |
| Table/File |
|  |
| Data types – Text, Number, Date/time, Currency, Autonumber, Yes/No, OLE Object, Memo |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| Flat-file Database |
|  |
| Relational Database |
|  |
| Why use a computerised database? |
|  |
| Primary key |
|  |
| Foreign key |
|  |
| Types of relationship – One-to-many, One-to-one, Many-to-many |
|  |
|  |
|  |
|  |
| Queries |
|  |
| Forms |
|  |

Relational databases

Five rules:

1. Data must be atomic – one piece of data per field. Eg. First name and last name must be in two separate fields
2. No duplication of data – one piece of data must be in one field only. Eg. Don’t enter date of birth and age, as they represent the same data item
3. Field independence – modifying the contents of one field must not force you to modify another data item
4. No calculation result in the fields – the fields would not be independent in such a case. Eg price inclusive of VAT is the result of a calculation on the price exclusive of VAT
5. A single index per table – this is called the ‘primary key’

# Student Mobile Phones

Create an Access Database that records the details of each pupil’s mobile phone. Initially that will be the student’s name, the make and model of their phone and the phone’s operating system.

There is a flat-file of mobile phones (see “accompanying database”) that you could use on: <http://agbonline.co.uk/SKE.html#Databases_SQL>

SQL

SQL is a standard language for accessing and manipulating databases

## What is SQL?

* SQL stands for Structured Query Language
* SQL lets you access and manipulate databases
* SQL is an ANSI (American National Standards Institute) standard

## What Can SQL do?

* SQL can execute queries against a database
* SQL can retrieve data from a database
* SQL can insert records in a database
* SQL can update records in a database
* SQL can delete records from a database
* SQL can create new databases
* SQL can create new tables in a database
* SQL can create stored procedures in a database
* SQL can create views in a database
* SQL can set permissions on tables, procedures, and views

# Work through the first few exercises on the w3schools website <http://www.w3schools.com/sql/default.asp> , down to ‘SQL Between’

## Here is the SQL Quick Reference from w3schools:

|  |  |
| --- | --- |
| **SQL Statement** | **Syntax** |
| AND / OR | SELECT column\_name(s)FROM table\_nameWHERE conditionAND|OR condition |
| ALTER TABLE | ALTER TABLE table\_name ADD column\_name datatypeorALTER TABLE table\_name DROP COLUMN column\_name |
| AS (alias) | SELECT column\_name AS column\_aliasFROM table\_nameorSELECT column\_nameFROM table\_name  AS table\_alias |
| BETWEEN | SELECT column\_name(s)FROM table\_nameWHERE column\_nameBETWEEN value1 AND value2 |
| CREATE DATABASE | CREATE DATABASE database\_name |
| CREATE TABLE | CREATE TABLE table\_name(column\_name1 data\_type,column\_name2 data\_type,column\_name2 data\_type,...) |
| CREATE INDEX | CREATE INDEX index\_nameON table\_name (column\_name)orCREATE UNIQUE INDEX index\_nameON table\_name (column\_name) |
| CREATE VIEW | CREATE VIEW view\_name ASSELECT column\_name(s)FROM table\_nameWHERE condition |
| DELETE | DELETE FROM table\_nameWHERE some\_column=some\_valueorDELETE FROM table\_name (**Note:**Deletes the entire table!!)DELETE \* FROM table\_name (**Note:**Deletes the entire table!!) |
| DROP DATABASE | DROP DATABASE database\_name |
| DROP INDEX | DROP INDEX table\_name.index\_name (SQL Server)DROP INDEX index\_name ON table\_name (MS Access)DROP INDEX index\_name (DB2/Oracle)ALTER TABLE table\_nameDROP INDEX index\_name (MySQL) |
| DROP TABLE | DROP TABLE table\_name |
| EXISTS | IF EXISTS (SELECT \* FROM table\_name WHERE id = ?)BEGIN--do what needs to be done if existsENDELSEBEGIN--do what needs to be done if notEND |
| GROUP BY | SELECT column\_name, aggregate\_function(column\_name)FROM table\_nameWHERE column\_name operator valueGROUP BY column\_name |
| HAVING | SELECT column\_name, aggregate\_function(column\_name)FROM table\_nameWHERE column\_name operator valueGROUP BY column\_nameHAVING aggregate\_function(column\_name) operator value |
| IN | SELECT column\_name(s)FROM table\_nameWHERE column\_nameIN (value1,value2,..) |
| INSERT INTO | INSERT INTO table\_nameVALUES (value1, value2, value3,....)*or*INSERT INTO table\_name(column1, column2, column3,...)VALUES (value1, value2, value3,....) |
| INNER JOIN | SELECT column\_name(s)FROM table\_name1INNER JOIN table\_name2 ON table\_name1.column\_name=table\_name2.column\_name |
| LEFT JOIN | SELECT column\_name(s)FROM table\_name1LEFT JOIN table\_name2 ON table\_name1.column\_name=table\_name2.column\_name |
| RIGHT JOIN | SELECT column\_name(s)FROM table\_name1RIGHT JOIN table\_name2 ON table\_name1.column\_name=table\_name2.column\_name |
| FULL JOIN | SELECT column\_name(s)FROM table\_name1FULL JOIN table\_name2 ON table\_name1.column\_name=table\_name2.column\_name |
| LIKE | SELECT column\_name(s)FROM table\_nameWHERE column\_name LIKE pattern |
| ORDER BY | SELECT column\_name(s)FROM table\_nameORDER BY column\_name [ASC|DESC] |
| SELECT | SELECT column\_name(s)FROM table\_name |
| SELECT \* | SELECT \*FROM table\_name |
| SELECT DISTINCT | SELECT DISTINCT column\_name(s)FROM table\_name |
| SELECT INTO | SELECT \*INTO new\_table\_name [IN externaldatabase]FROM old\_table\_name*or*SELECT column\_name(s)INTO new\_table\_name [IN externaldatabase]FROM old\_table\_name |
| SELECT TOP | SELECT TOP number|percent column\_name(s)FROM table\_name |
| TRUNCATE TABLE | TRUNCATE TABLE table\_name |
| UNION | SELECT column\_name(s) FROM table\_name1UNIONSELECT column\_name(s) FROM table\_name2 |
| UNION ALL | SELECT column\_name(s) FROM table\_name1UNION ALLSELECT column\_name(s) FROM table\_name2 |
| UPDATE | UPDATE table\_nameSET column1=value, column2=value,...WHERE some\_column=some\_value |
| WHERE | SELECT column\_name(s)FROM table\_nameWHERE column\_name operator value |