SQL Quick Guide

SQL dates are stored in the format yyyy/mm/dd

SELECT Statements

SELECT statements are used to query a database and return some values.

- Specify the fields you want to display: **SELECT fieldname1, fieldname2**
- Name the tables that the fields are in: **FROM tablename**
- The criteria for your search: **WHERE fieldname1 = 'value'**
- Any sort order that you want to apply: **ORDER BY fieldname1 ASC/DESC**

```
SELECT name, age
FROM pupil
WHERE age <18
ORDER BY name ;
```

When searching for text values you must use 'speech marks'
You can use logical conditions such as AND, OR and NOT to create more complex queries.

DELETE Statements

DELETE statements delete records from a table

- Name the table where the records are to be deleted: **DELETE fieldname**
- Which tables that the fields are in: **FROM tablename**
- The criteria for your search: **WHERE fieldname1 = "value"**

```
DELETE FROM Course
WHERE courseID='BMX05';
```

If you don't specify criteria then every record in the table will be deleted!

INSERT Statements – Entering values for every field

INSERT statements can be used to add new records into the database.

- Specify the tablename to insert into: **INSERT INTO Instructor**
- Specify the values to be inserted: **(5, 'D Thomas', '1985/11/30', 5)**

```
INSERT INTO Instructor
VALUES  (5, 'D Thomas', '1985/11/30', 5);
```

INSERT Statements – Only entering values for some fields

Sometimes you may not be adding a value for every single field.

- Specify the tablename to insert into: **INSERT INTO Instructor**
- Specify the fields you are entering data for: **(InstructorID, Name)**
- List the values to be inserted: **(5, 'D Thomas')**

```
INSERT INTO Instructor (InstructorID, Name)
VALUES (5, 'D Thomas');
```

UPDATE Statements

UPDATE statements will edit existing data in the table.

- Name the table where the records are: **UPDATE tablename**
- Set the new values: **SET fieldname = value1, fieldname = value2**
- Criteria if there are any: **WHERE fieldname = fieldvalue**

```
UPDATE Course
SET Date = '2017/12/11'
```

EQUI-JOINS

An equi-join is when two matching columns from related tables are selected (joined) using the primary key and its matching foreign key from the tables.

```
SELECT Instructor.Name, Course.Title, Course.Date
FROM Instructor, Course
WHERE Instructor.InstructorID = Course.InstructorID AND Instructor.Name = "John Smith" ;
```