Databases and simple sql commands

Database Course

What we'll cover

- What is sql?
- Create, read, update and delete operations in sql.
- Advanced single table queries using the select statement.

ARCTICTIGER

SQL

- Structured Query Language
- A standardised language for retrieving and updating data in a relational database.

Creating a row

- The insert statement

```
INSERT INTO  tablename
(column-name1, column- name2, …,  column-nameX)
VALUES
('value1', 'value2', …, valueX);
```

ARCTICTIGER

```
INSERT INTO staff
(firstname, surname, dob, street_address, town, postcode, mobile, email, salary)
VALUES
('Tom', 'Blackmore', '1976-04-21', '2 Studland Road', 'Kingston-Upon-Thames',
'KT2 5HJ', '0208 546 2786', 'tom.blackmore@arctictiger.se', 37000);
```

- Strings should be enclosed in single quotation marks.

  ‘value'

- Numeric values should be without quotes
- null can be used to represent no data and is better than ''

Read

- The SELECT statement is where the power in SQL lies.

SELECT column1, column2, …, columnX
FROM table-name;

WHERE

SELECT column1, column2, …, columnX

FROM table-name

WHERE coulmn1 = ‘hello’ and column2 like ‘%b%’ and columnX > 1000;

- WHERE can be really complex too, but beware complex SELECT statements eat your processing power.

UPDATE

```
UPDATE Person
SET namn = 'Fredrik Reinfeldt'
WHERE yrke = 'Statsminister';
```

DELETE

```
DELETE FROM Person
WHERE stad = 'Atlantis';
```

SQL exercises

- http://www.databasteknik.se/webbkursen/sql/index.html

Part 2 What we'll cover

- Creating new tables
- Deleting and truncating tables
- Changing tables and adding indexes
- Creating views

Creating tables

```
CREATE TABLE people
(
        id int,
        firstname varchar(255),
        surname varchar(255),
        address varchar(255),
        city varchar(255)
);
```

Creating a table

In the following example, firstname, surname and city are required for every row.

City must be unique. I.e. you can't have 2 people from the same city.

```
CREATE TABLE people
(
        id mediumint(8) unsigned NOT NULL auto_increment,
        firstname varchar(255) NOT NULL,
        surname varchar(255) NOT NULL,
        address varchar(255) default NULL,
        city varchar(255) default NULL,
        UNIQUE(city),
        PRIMARY KEY (id)
) AUTO_INCREMENT=1;
```

Altering a table

To add a column to an existing table

```
ALTER TABLE table_name
ADD column_name datatype;
```

To delete a column from an existing table

```
ALTER TABLE table_name
DROP COLUMN column_name
```

To change the data type of an existing column

```
ALTER TABLE table_name
MODIFY COLUMN column_name datatype
```

To add an index on a certain column

```
ALTER TABLE table_name ADD INDEX product_id (product_id)
```

Deleting tables

Delete contents of a table, but not it's structure

```
TRUNCATE table_name;
```

Deleting tables

Delete a table entirely

```
DROP TABLE table_name;
# or
DROP TABLE IF EXISTS table_name;
```

Views

- We can output the results of our select statements to a view.
- After creating a view it can be queried as if it was a table.
- Note views DO NOT contain any data, the data is read from the underlying tables.

View syntax

```
CREATE VIEW birthdays
AS
        SELECT *
        FROM staff
        WHERE month(dob) = month(now())
                and day(dob) > day(now());
```

ARCTICTIGER

Deleting a view

```
DROP VIEW birthdays;
```