

# SQL Basics (cont.)

Question: "If you put in an imaginary date, say a 13 in the month column, would SQL give an error?"

Answer: "It depends. Mostly it should, but different SQL engines will respond differently. Typically this is a point you would look up, rather than remember.

## Conditions Involving Text

Example:

```
SELECT *  
  
FROM students  
  
WHERE first_name = 'Kieran';
```

Note:

- 'Kieran', 'kieran' and 'KiErAn' treated as equal
- 'Kieran', 'Kieran ' and ' Kieran' treated as unequal (spaces)
- There is a way of making SQL case-sensitive, but we'll leave that alone for the moment.

## Case-sensitivity

Linux filenames are case sensitive, while Windows filenames aren't (examples)

SQL technically isn't case-sensitive, but not all implementations respect this, especially with table name and database names.

It is safest to work as if SQL is case-sensitive and to be consistent.

## Comparing Strings

In everyday life we use dictionary ordering to order words:

e.g. aardvark < anaconda < armadillo < baboon < cat < zebra

Note: computer < computers (space < a)

SQL expands on this by giving symbols their own letter positions.

## Comparisons Involving Different Types

Comparing integers to strings (e.g.) satisfies SQL's rules, but makes no sense.

Most of the time you'll get an empty results table, but the results can be unpredictable.

Avoid these kinds of comparisons.

## Escape Characters

If you have a name with an apostrophe in it (e.g. O'Reilly), you need to take precautions to avoid syntax errors.

Precede the apostrophe in the word with a "\", e.g.:

```
SELECT... = 'O\'Reilly'
```

For a backslash, type "\\".

## More SQL Basics

### Conditions

The SQL conditions "hometown = 'Cork'" and "points >= 500" are either true or false for each row in the table students.

They are expressions with true/false value.

What if we want both to be true?

Use the keyword AND:

```
SELECT ...  
FROM ...  
WHERE hometown = 'Cork' AND points >= 500;
```

PHP also includes and/or operators (subtle differences)

There is also the OR keyword, which operates as a normal logical or.

Note we can't simplify "hometown = 'Cork' OR hometown = 'Limerick'" to "hometown = 'Cork' OR 'Limerick'".

Have to put in the "hometown =" for each one.

Also, can use brackets for grouping.

## **AND/OR Precedence**

AND is evaluated before OR, except where brackets are used appropriately.

## **IN Keyword**

...WHERE points >= 500 AND

hometown IN ('Cork', 'Limerick', 'Tralee');

If the value in the hometown entry is a member of the set specified after IN, the IN operation returns true.