



Adding new data

SQL INSERT



When new data arrives

Data is produced very quickly. It is therefore not uncommon to store new data in a database table. It could be a new Customer registering, or a new Order being placed. Or as with our example databases, a new Book is added to the books table or a new Car is added to the Cars table.

With SQL this is performed using the INSERT INTO statement:

```
INSERT INTO <table> (<col>[,<col>]*) VALUES (<val>[,<val>]*);
```

Example:

```
INSERT INTO cars (make, color, licensenumber)
VALUES ('Berlingo', 'Red', 'HES 000');
```

The order matters

Let's look at the previous INSERT INTO statement again:

```
INSERT INTO cars (make, color, licensenumber)
  VALUES ('Berlingo', 'Red', 'HES 000');
```

We listed the column names and then the values in the corresponding order.

If you have very large tables (with many columns), this is of course a source for errors if you get mixed up concerning the order of the columns/values.

Read

http://www.w3schools.com/sql/sql_insert.asp

What's next?

Now that we know how to retrieve (`SELECT`), change (`UPDATE`), remove (`DELETE`), and add (`INSERT`) data, we are going to look at a more realistic data model involving more than one table.

This is called normalisation in a fancy language but we'll simply call it “decomposing large tables” and “combining (joining) data from linked tables”.