**Aggregate (Group) Function**

Aggregate functions perform an operation on a group of rows and return one result.

The operations are:

* SUM () gives the total of a group of rows, (satisfying any conditions) of the given column, where the given column is numeric.
* AVG () gives the average of the given column.
* MAX () gives the largest figure in the given column.
* MIN () gives the smallest figure in the given column.
* COUNT (\*) gives the number of rows satisfying the conditions.

This query below shows how many books are on the Book table.

A screenshot of a computer

AI-generated content may be incorrect.

We could also use an attribute name in the Count function, and it would return the same result.

A screenshot of a book

AI-generated content may be incorrect.



A screenshot of a book

AI-generated content may be incorrect.We could also use a different alias name, and it would return the same result.

This query returns the **total (SUM)** price of all the books in the Book table.

A screenshot of a computer

AI-generated content may be incorrect.



This query returns the **AVERAGE** price of a book in the Book table.

A screenshot of a computer

AI-generated content may be incorrect.



This query returns the **MINIMUM** price of a book in the Book table.

A screenshot of a computer

AI-generated content may be incorrect.



This query returns the **MAXIMUM** price of a book in the Book table.

A screenshot of a computer

AI-generated content may be incorrect.



You can return more than one aggregate function on the same attribute.

A screenshot of a graph

AI-generated content may be incorrect.



You can return more than one aggregate function, and not all the aggregate functions need be on the same attribute.

A screenshot of a computer

AI-generated content may be incorrect.

We can use **Where statements** to limit the rows selected.

A screenshot of a computer

AI-generated content may be incorrect.



If we want to use aggregate functions on ‘groups’ of rows in the table, we need to use the **GROUP BY clause.**

A white rectangular sign with black text

AI-generated content may be incorrect.

NB: This selects the maximum price of each ***type*** of book in the table.

Returning the most expensive book by **Type**

A screenshot of a computer screen

AI-generated content may be incorrect.

A screenshot of a table

AI-generated content may be incorrect.

**Having**

The HAVING clause is designed for use with the GROUP BY clause.

NB: ***WHERE ‘filters’ individual rows*** while ***HAVING ‘filters’ groups.***

That is; the HAVING clause allows you to restrict the groups you return. It is usual, and logical, to include the GROUP BY clause before the HAVING clause. Groups are formed and group functions calculated before the HAVING clause is applied to a SELECT group for output.

A screenshot of a computer

AI-generated content may be incorrect.Returning courses with more than 1 student on them:

A screenshot of a computer

AI-generated content may be incorrect.