

SQLITE - HAVING CLAUSE

http://www.tutorialspoint.com/sqlite/sqlite_having_clause.htm

Copyright © tutorialspoint.com

The HAVING clause enables you to specify conditions that filter which group results appear in the final results.

The WHERE clause places conditions on the selected columns, whereas the HAVING clause places conditions on groups created by the GROUP BY clause.

Syntax:

The following is the position of the HAVING clause in a SELECT query:

```
SELECT
FROM
WHERE
GROUP BY
HAVING
ORDER BY
```

The HAVING clause must follow the GROUP BY clause in a query and must also precede the ORDER BY clause if used. The following is the syntax of the SELECT statement, including the HAVING clause:

```
SELECT column1, column2
FROM table1, table2
WHERE [ conditions ]
GROUP BY column1, column2
HAVING [ conditions ]
ORDER BY column1, column2
```

Example:

Consider COMPANY table is having the following records:

ID	NAME	AGE	ADDRESS	SALARY
1	Paul	32	California	20000.0
2	Allen	25	Texas	15000.0
3	Teddy	23	Norway	20000.0
4	Mark	25	Rich-Mond	65000.0
5	David	27	Texas	85000.0
6	Kim	22	South-Hall	45000.0
7	James	24	Houston	10000.0
8	Paul	24	Houston	20000.0
9	James	44	Norway	5000.0
10	James	45	Texas	5000.0

Following is the example, which would display record for which name count is less than 2:

```
sqlite > SELECT * FROM COMPANY GROUP BY name HAVING count(name) < 2;
```

This would produce the following result:

ID	NAME	AGE	ADDRESS	SALARY
2	Allen	25	Texas	15000
5	David	27	Texas	85000
6	Kim	22	South-Hall	45000
4	Mark	25	Rich-Mond	65000
3	Teddy	23	Norway	20000

Following is the example, which would display record for which name count is greater than 2:

```
sqlite > SELECT * FROM COMPANY GROUP BY name HAVING count(name) > 2;
```

This would produce the following result:

ID	NAME	AGE	ADDRESS	SALARY
10	James	45	Texas	5000