SQL Syntax Cheatsheet

Thomas Boxall

up2108121@myport.ac.uk

April 2023

1 PostegreSQL 'Slash' Commands

`\l` list all databases in the server
`\c databaseName` connect to database
`\d` list everything in the database
`\dt` list just the tables

2 CREATE

a database: `CREATE DATABASE databaseName;`
a table:

```
CREATE TABLE tableName(
    attribute dataType constraints,
    ...
);
```

3 DROP & DELETE

a whole database: `DROP DATABASE databaseName;`
a single table: `DROP TABLE tableName;`
a single record: `DELETE FROM tableName WHERE attribute = value;`

4 ALTER

a table to add new column: `ALTER TABLE tableName ADD COLUMN colName dataType;`

5 INSERT

data to a table: `INSERT INTO tableName (attr1, attr2) VALUES(val1, val2);`

6 UPDATE

a record in a table: `UPDATE tableName SET attributeName = val WHERE idAttribute = val;`

7 SELECT

everything from a table: `SELECT * FROM tableName;`
only certain records from a table: `SELECT * FROM tableName WHERE attributeName = desiredValue;`
only certain attributes from a table: `SELECT attr1, attr2 FROM tableName;`

8 Functions

number of values: `COUNT(toCount)`
maximum value: `MAX()`
minimum value: `MIN()`
average value: `AVG()`
generate a random number: `RANDOM()`
current date and time: `NOW()`

9 `JOIN`

two tables together using an inner join

```
SELECT tableName.attributeName, ... FROM tableName
JOIN tableToJoinName ON tableToJoinName.attribute = currentTableName.attribute;
```

two tables together not using inner join but achieve the same results:

```
SELECT attributeName, attributeName FROM tableOneName, tableTwoName
WHERE tableOneName.attributeName = tableTwoName.attributeName;
```

two tables together using a left join:

```
SELECT currentTableName.attributeName, ... FROM currentTableName
LEFT JOIN tableTwo ON tableToJoinName.attribute = currentTableName.attribute;
```

two tables together using an outer join:

```
SELECT currentTableName.attributeName, ... FROM currentTableName
RIGHT JOIN tableTwo ON tableToJoinName.attribute = currentTableName.attribute;
```

two tables together using a full outer join:

```
SELECT currentTableName.attributeName, ... FROM currentTableName
FULL OUTER JOIN tableTwo ON tableToJoinName.attribute = currentTableName.attribute;
```

10 `GROUP BY`

an attribute in the table:

```
SELECT attributeOne, attributeTwo FROM tableName
GROUP BY attributeOne;
```

an attribute in the table and only show certain records:

```
SELECT attributeOne, attributeTwo FROM tableName
GROUP BY attributeOne
HAVING attributeName = value;
```

11 `ORDER BY`

an attribute in ascending order:

```
SELECT attributeName FROM tableName ORDER BY attributeName;
```

an attribute in ascending order:

```
SELECT attributeName FROM tableName ORDER BY attributeName ASC;
```

an attribute in descending order:

```
SELECT attributeName FROM tableName ORDER BY attributeName DESC;
```

12   Wildcards

any number of characters: `%`

13   Views

create a view: `CREATE viewName AS queryString;`
executeView `SELECT * FROM viewName;`