

SQL Cheat Sheet

```
In [1]: # !pip install ipython-sql
# !pip install psycopg2 # PostgreSQL
# !pip install mysql # MySQL
```

```
In [2]: SQL_CONN = 'postgresql://wing@localhost/wing'

%config SqlMagic.displaycon=False
%load_ext sql
%sql {SQL_CONN}
```

1. Tables: CREATE, ALTER, DROP

```
In [3]: %%sql
DROP TYPE IF EXISTS t_grade;
CREATE TYPE t_grade AS ENUM('A', 'B', 'C', 'D');
```

Done.

Done.

```
Out[3]: []
```

```
In [4]: %%sql
CREATE TABLE advisors (
    id SERIAL PRIMARY KEY,
    name VARCHAR(255) NOT NULL
);

CREATE TABLE students (
    id SERIAL PRIMARY KEY,
    advisorId INT NOT NULL,
    name VARCHAR(255) UNIQUE NOT NULL,
    age INT NOT NULL,
    height DECIMAL(8,2),
    gpa REAL,
    grade t_grade DEFAULT 'D',
    created TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY(advisorId) REFERENCES advisors(id)
);
```

Done.

Done.

```
Out[4]: []
```

```
In [5]: %%sql
ALTER TABLE advisors
ADD telephone VARCHAR(255);

ALTER TABLE advisors
DROP COLUMN telephone;
```

Done.

Done.

Out[5]: []

```
In [6]: %%sql
DROP TABLE students;
DROP TABLE IF EXISTS advisors;
```

Done.
Done.

Out[6]: []

2. Views: CREATE VIEW ... AS ...

```
In [7]: %%sql
CREATE VIEW cust_names AS
SELECT firstName, lastName FROM customers WHERE state='OR';

SELECT * FROM cust_names;
```

Done.
2 rows affected.

Out[7]: **firstname** **lastname**

jane	doe
john	doe

```
In [8]: %%sql
DROP VIEW cust_names;
```

Done.

Out[8]: []

3. Records: INSERT, UPDATE, DELETE

3.1 INSERT INTO ...

```
In [9]: %%sql
INSERT INTO customers (firstName, lastName, age) VALUES ('jack', 'daniels', 99)
```

1 rows affected.

Out[9]: []

```
In [10]: %%sql
INSERT INTO customers (firstName, lastName, age) SELECT firstName, lastName, age
```

1 rows affected.

Out[10]: []

3.2 UPDATE ... SET

```
In [11]: %%sql
UPDATE customers SET
```

```
age=age-1,
email='jack@daniels.net',
state='CA'
WHERE lastname='daniels' AND age=99;
```

1 rows affected.

Out[11]: []

3.3 DELETE FROM ...

```
In [12]: %%sql
DELETE FROM customers
WHERE firstName='jack' AND age=98;
```

2 rows affected.

Out[12]: []

SELECT ... (AS ...) FROM ... (AS ...) WHERE ... GROUP BY ... HAVING ... ORDER BY ... (DESC)
LIMIT ... OFFSET ...

4. SELECT ... AS ... FROM

```
In [13]: %%sql
SELECT CONCAT(firstName, ' ', lastName) AS name, age + 1 AS "age + 1" FROM cust
```

4 rows affected.

Out[13]:

name	age + 1
jane doe	19
john doe	20
jack daniels	23
jane austen	23

4.1 DISTINCT

```
In [14]: %%sql
SELECT DISTINCT state FROM customers;
```

3 rows affected.

Out[14]: state

None

OR

WA

```
In [15]: %%sql
SELECT DISTINCT lastName, state FROM customers;
```

3 rows affected.

Out[15]: **lastname state**

austen	None
daniels	WA
doe	OR

4.2 UNION

```
In [16]: %%sql
SELECT name FROM states
UNION
SELECT name FROM products;
```

5 rows affected.

Out[16]: **name**

xbox
ps5
ps6
Oregon
California

4.3 SELECT ... INTO / INSERT INTO

```
In [17]: %%sql
SELECT * INTO new_orders
FROM orders WHERE qty>50;

INSERT INTO new_orders
SELECT *
FROM orders WHERE qty<=50;

DROP TABLE new_orders;
```

1 rows affected.

3 rows affected.

Done.

Out[17]: []

5. WHERE ... IS / NOT / AND / OR / IN / LIKE / BETWEEN

```
In [18]: %%sql
SELECT * FROM customers WHERE firstName='john' AND lastName='doe' OR lastName='
```

2 rows affected.

Out[18]: **id firstname lastname age email state**

2	john	doe	19	john@doe.com	OR
4	jack	daniels	22	None	WA

In [19]:

```
%%sql
SELECT * FROM customers where age=(SELECT MIN(age) AS max_age FROM customers);
```

1 rows affected.

Out[19]:

id	firstname	lastname	age	email	state
1	jane	doe	18	jane@doe.com	OR

In [20]:

```
%%sql
SELECT * FROM customers WHERE age>=20;
```

2 rows affected.

Out[20]:

id	firstname	lastname	age	email	state
4	jack	daniels	22	None	WA
5	jane	austen	22	None	None

In [21]:

```
%%sql
SELECT * FROM customers WHERE age BETWEEN 17 AND 19
```

2 rows affected.

Out[21]:

id	firstname	lastname	age	email	state
1	jane	doe	18	jane@doe.com	OR
2	john	doe	19	john@doe.com	OR

In [22]:

```
%%sql
SELECT * FROM customers WHERE firstName LIKE 'jan%';
```

2 rows affected.

Out[22]:

id	firstname	lastname	age	email	state
1	jane	doe	18	jane@doe.com	OR
5	jane	austen	22	None	None

In [23]:

```
%%sql
SELECT * FROM customers WHERE firstName NOT LIKE 'ja__';
```

1 rows affected.

Out[23]:

id	firstname	lastname	age	email	state
2	john	doe	19	john@doe.com	OR

In [24]:

```
%%sql
SELECT * FROM customers WHERE lastName IN ('austen', 'daniels');
```

2 rows affected.

Out[24]:

id	firstname	lastname	age	email	state
4	jack	daniels	22	None	WA
5	jane	austen	22	None	None

In [25]:

```
%%sql
SELECT * FROM customers WHERE email IS NULL /* "= NULL" wouldn't work! */;
```

2 rows affected.

Out[25]:

id	firstname	lastname	age	email	state
4	jack	daniels	22	None	WA
5	jane	austen	22	None	None

6. GROUP BY ... HAVING

Aggregate Functions

In [26]:

```
%%sql
SELECT MIN /*MAX,AVG,SUM*/ (age) FROM customers;
```

1 rows affected.

Out[26]:

min
18

In [27]:

```
%%sql
SELECT COUNT(*) FROM customers WHERE age>20;
```

1 rows affected.

Out[27]:

count
2

In [28]:

```
%%sql
SELECT COUNT(state) AS "#States", COUNT(*) AS "#Custs" FROM customers;
```

1 rows affected.

Out[28]:

#States	#Custs
3	4

In [29]:

```
%%sql
SELECT COUNT(lastName), COUNT(DISTINCT lastName) FROM customers;
```

1 rows affected.

Out[29]:

count	count_1
4	3

SELECT ... AggFn() GROUP BY ... HAVING AggFn() ...

In [30]:

```
%%sql
SELECT lastName, COUNT(*), MAX(age) FROM customers GROUP BY lastName;
```

3 rows affected.

Out[30]: **lastname count max**

austen	1	22
daniels	1	22
doe	2	19

In [31]: `%%sql
SELECT lastName, MAX(age) FROM customers WHERE age<20 GROUP BY lastName;`

1 rows affected.

Out[31]: **lastname max**

doe	19
-----	----

In [32]: `%%sql
SELECT lastName, COUNT(*) FROM customers GROUP BY lastName HAVING COUNT(*)>1;`

1 rows affected.

Out[32]: **lastname count**

doe	2
-----	---

7. ORDER BY ... DESC

In [33]: `%%sql
SELECT * FROM customers ORDER BY lastname DESC, firstname;`

4 rows affected.

Out[33]: **id firstname lastname age email state**

1	jane	doe	18	jane@doe.com	OR
2	john	doe	19	john@doe.com	OR
4	jack	daniels	22	None	WA
5	jane	austen	22	None	None

8. LIMIT ... OFFSET

In [34]: `%%sql
SELECT * FROM customers;`

4 rows affected.

Out[34]: **id firstname lastname age email state**

1	jane	doe	18	jane@doe.com	OR
2	john	doe	19	john@doe.com	OR
4	jack	daniels	22	None	WA
5	jane	austen	22	None	None

In [35]: `%%sql`

```
SELECT * FROM customers LIMIT 2 OFFSET 2;
```

2 rows affected.

Out[35]:

	id	firstname	lastname	age	email	state
	4	jack	daniels	22	None	WA
	5	jane	austen	22	None	None

9. JOIN ... ON

In [36]:

```
%%sql
SELECT * FROM customers
```

4 rows affected.

Out[36]:

	id	firstname	lastname	age	email	state
	1	jane	doe	18	jane@doe.com	OR
	2	john	doe	19	john@doe.com	OR
	4	jack	daniels	22	None	WA
	5	jane	austen	22	None	None

In [37]:

```
%%sql
SELECT * FROM products
```

3 rows affected.

Out[37]:

	id	name	price
	1	xbox	299.95
	2	ps5	399.00
	3	ps6	None

In [38]:

```
%%sql
SELECT * FROM orders
```

4 rows affected.

Out[38]:

	id	custid	prodid	qty	date
	1	1	2	1	2020-01-01 00:00:00
	2	2	1	1	2020-01-02 00:00:00
	3	4	1	2	2020-02-03 00:00:00
	18	1	1	99	2022-06-26 19:56:21.233966

9.1 JOIN (Inner/Default)

In [39]:

```
%%sql
SELECT customers.id, customers.state, states.name
FROM customers
JOIN states ON customers.state=states.state;
```

2 rows affected.

Out[39]:

id	state	name
2	OR	Oregon
1	OR	Oregon

In [40]:

```
%%sql
SELECT customers.firstName, customers.lastName, products.name AS "Product Name"
FROM customers
JOIN orders ON customers.id=orders.custId
JOIN products ON orders.prodId=products.id;
```

4 rows affected.

Out[40]:

firstname	lastname	Product Name	qty
jane	doe	ps5	1
john	doe	xbox	1
jack	daniels	xbox	2
jane	doe	xbox	99

In [41]:

```
%%sql
SELECT c.firstName, c.lastName, p.name, p.price, o.qty
FROM orders AS o
JOIN customers AS c ON o.custid=c.id
JOIN products AS p ON o.prodId=p.id
```

4 rows affected.

Out[41]:

firstname	lastname	name	price	qty
jane	doe	ps5	399.00	1
john	doe	xbox	299.95	1
jack	daniels	xbox	299.95	2
jane	doe	xbox	299.95	99

In [42]:

```
%%sql
SELECT c.firstName, c.lastName, o.id
FROM customers AS c
JOIN orders AS o ON c.id=o.custId;
```

4 rows affected.

Out[42]:

firstname	lastname	id
jane	doe	1
john	doe	2
jack	daniels	3
jane	doe	18

9.2 LEFT JOIN (Outer)

In [43]:

```
%%sql
SELECT customers.id, customers.state, states.name
```

```
FROM customers
LEFT JOIN states ON customers.state=states.state;
```

4 rows affected.

```
Out[43]:
```

	id	state	name
	2	OR	Oregon
	1	OR	Oregon
	5	None	None
	4	WA	None

9.3 RIGHT JOIN (Outer)

```
In [44]: %%sql
SELECT customers.id, customers.state, states.name
FROM customers
RIGHT JOIN states ON customers.state=states.state;
```

3 rows affected.

```
Out[44]:
```

	id	state	name
	2	OR	Oregon
	1	OR	Oregon
	None	None	California

10. Strings

```
In [45]: %%sql
SELECT 'a' || 'bc', CONCAT('a', 'bc');
```

1 rows affected.

```
Out[45]:
```

	?column?	concat
	abc	abc

```
In [46]: %%sql
SELECT name, LENGTH(name) AS len FROM products ORDER BY len, name;
```

3 rows affected.

```
Out[46]:
```

	name	len
	ps5	3
	ps6	3
	xbox	4

```
In [47]: %%sql
SELECT LOWER /*UPPER*/ ('aBc');
```

1 rows affected.

Out[47]: **lower**

abc

```
In [48]: %%sql
SELECT SUBSTR('2022-02-14', 9, 2) AS day, SUBSTR('abc', 2);
```

1 rows affected.

Out[48]: **day substr**

14	bc
----	----

```
In [49]: %%sql
SELECT TRIM /*LTRIM,RTRIM*/ ('-abc-', '-');
```

1 rows affected.

Out[49]: **btrim**

abc

```
In [50]: %%sql
SELECT 'ab'='aB'
```

1 rows affected.

Out[50]: **?column?**

False

11. Numbers

```
In [51]: %%sql
SELECT 1 / 2 AS "1/2", 1 % 2 AS "1%2";
```

1 rows affected.

Out[51]: **1/2 1%2**

0	1
---	---

```
In [52]: %%sql
SELECT ROUND(1.51);
```

1 rows affected.

Out[52]: **round**

2

12. Transactions

```
In [53]: %%sql /* SQLite */
BEGIN TRANSACTION;
INSERT INTO orders (custId, prodId, qty) VALUES (1, 1, 99);
DELETE FROM orders WHERE qty=99;
END TRANSACTION;
SELECT * from orders;
```

Done.
1 rows affected.
2 rows affected.
Done.
3 rows affected.

Out[53]:

id	custid	prodid	qty	date
1	1	2	1	2020-01-01 00:00:00
2	2	1	1	2020-01-02 00:00:00
3	4	1	2	2020-02-03 00:00:00

In [54]:

```
%%sql /* SQLite */
BEGIN TRANSACTION;
INSERT INTO orders (custId, prodId, qty) VALUES (1, 1, 99);
ROLLBACK;
SELECT * from orders;
```

Done.
1 rows affected.
Done.
4 rows affected.

Out[54]:

id	custid	prodid	qty	date
1	1	2	1	2020-01-01 00:00:00
2	2	1	1	2020-01-02 00:00:00
3	4	1	2	2020-02-03 00:00:00
20	1	1	99	2022-06-26 19:56:32.214131

13. Triggers

In []:

```
%%sql /* SQLite */
CREATE TRIGGER my_trigger AFTER INSERT ON products
BEGIN
    UPDATE products SET price=0.95*price WHERE id=NEW.id;
END;
```

14. Procedures

In [55]:

```
%%sql
CREATE PROCEDURE insert_delete(first_name VARCHAR, last_name VARCHAR, age INT)
BEGIN ATOMIC
INSERT INTO customers (firstName, lastName, age) VALUES
(first_name, last_name, age);
DELETE FROM customers
WHERE firstName=first_name AND lastName=last_name;
END;
```

Done.

Out[55]: []

In [56]:

```
%%sql
```

```
CALL insert_delete('joe', 'shmoe', 2);
DROP PROCEDURE insert_delete;
```

Done.
Done.

Out[56]: []

15. Subselects

```
In [57]: %%sql
SELECT * FROM orders WHERE prodId IN (SELECT id FROM products WHERE price>300);
```

1 rows affected.

Out[57]:

id	custid	prodid	qty	date
1	1	2	1	2020-01-01 00:00:00

```
In [58]: %%sql
SELECT firstName, lastName, ss.qty, products.name
FROM (SELECT custId, prodId, qty FROM orders WHERE qty>1) AS ss
JOIN customers ON ss.custId=customers.id
JOIN products ON ss.prodId=products.id;
```

2 rows affected.

Out[58]:

firstname	lastname	qty	name
jack	daniels	2	xbox
jane	doe	99	xbox

16. ANY

```
In [59]: %%sql
SELECT name FROM products
WHERE id = ANY
(SELECT prodId FROM orders WHERE qty=1);
```

2 rows affected.

Out[59]:

name
xbox
ps5

17. CASE ... WHEN ... THEN ... ELSE ... END

```
In [60]: %%sql
SELECT *,
CASE
  WHEN qty>50 THEN 'whoa!'
  ELSE 'normal'
END AS comments
FROM orders
```

4 rows affected.

Out[60]:

id	custid	prodid	qty	date	comments
1	1	2	1	2020-01-01 00:00:00	normal
2	2	1	1	2020-01-02 00:00:00	normal
3	4	1	2	2020-02-03 00:00:00	normal
20	1	1	99	2022-06-26 19:56:32.214131	whoa!

In []: