More SQL, continued

Remy Wang, 2025

2 kinds of values in SQL

data values
256
10.33
04-08-2025
"UCLA"
NULL

logical values
T
F
UKNOWN

a.k.a. NULL in SQLite 💀

3 kinds of operators in SQL

op. kind	**example**	**input→output**
data	+ - * /	data→data
predicate	> < =	data→logic
logical	AND OR NOT	logic→logic

rules for NULL

op. kind	example	output w/null
data	+ - * /	NULL
predicate	> < =	UNKNOWN
logical	AND OR NOT	**3 valued logic**

To check if x is NULL: x IS NULL

3 valued logic

AND	T	F	U
T	T	F	**U**
F	F	F	**F**
U	**U**	**F**	U

OR	T	F	U
T	T	T	**T**
F	T	F	**U**
U	**T**	**U**	U

not U = U

```
SELECT *
  FROM R
 WHERE x < y
```

Only return rows where x < y is **TRUE**

outer joins

name	addr	phone	job
remy	...	123	UCLA
zifan	...	234	UCLA
vincent	...	345	UCLA
remy	...	123	UW
dan	...	456	UW
remy	...	123	**USC**

name	addr
UCLA	LA
UW	seattle

outer joins

name	addr	phone	job	name	addr
remy	...	123	UCLA	UCLA	LA
zifan	...	234	UCLA	UCLA	LA
vincent	...	345	UCLA	UCLA	LA
remy	...	123	UW	UW	seattle
dan	...	456	UW	UW	seattle
remy	...	123	**USC**	**NULL**	**NULL**

name	addr
UCLA	LA
UW	seattle

```
SELECT * FROM p LEFT OUTER JOIN e
                ON p.job = e.name
```

Aggregates ignore NULLs

What should these return?

R = { NULL, 1 }

```
SELECT COUNT(x) FROM R
```

```
SELECT SUM(x) FROM R
```

```
SELECT AVG(x) FROM R
```

```
SELECT MIN(x) FROM R
```

What should these return?

R = { **NULL** }

```
SELECT COUNT(x) FROM R
```

```
SELECT SUM(x) FROM R
```

```
SELECT AVG(x) FROM R
```

```
SELECT MIN(x) FROM R
```

What should these return?

R = { NULL }

```
SELECT COUNT(x) FROM R
SELECT SUM(x) FROM R
SELECT AVG(x) FROM R
SELECT MIN(x) FROM R
SELECT TOTAL(x) FROM R
```

Q: how many offices?

name	addr	phone	job
remy	...	123	UCLA
zifan	...	234	UCLA
vincent	...	345	UCLA
remy	...	123	UW
dan	...	456	UW
remy	...	123	**USC**
seymour	...	367	**USC**

name	addr
UCLA	LA
UW	seattle

Q: how many offices?

name	addr	phone	job
remy	…	123	UCLA
zifan	…	234	UCLA
vincent	…	345	UCLA
remy	…	123	UW
dan	…	456	UW
remy	…	123	USC
seymour	…	367	USC

name	addr
UCLA	LA
UW	seattle

```
SELECT COUNT(addr)
  FROM p, e
 WHERE p.job=e.name
 GROUP BY p.name
```

```
SELECT p.name, COUNT(addr)
  FROM p OUTER JOIN e
            ON p.job = e.name
GROUP BY p.name
```

name	addr	phone	job
remy	...	123	UCLA
zifan	...	234	UCLA
vincent	...	345	UCLA
remy	...	123	UW
dan	...	456	UW
remy	...	123	**USC**
seymour	...	367	**USC**

name	addr
UCLA	LA
UCLA	LA
UCLA	LA
UW	seattle
UW	seattle
NULL	**NULL**
NULL	**NULL**

name	addr	phone	job		name	addr
remy	...	123	UCLA		UCLA	LA
remy	...	123	UW		UW	seattle
remy	...	123	**USC**		**NULL**	**NULL**
zifan	...	234	UCLA		UCLA	LA
vincent	...	345	UCLA		UCLA	LA
dan	...	456	UW		UW	seattle
seymour	...	367	**USC**		**NULL**	**NULL**

name	addr	phone	job		name	addr
remy	...	123	UCLA		UCLA	LA
remy	...	123	UW		UW	seattle
remy	...	123	**USC**		**NULL**	**NULL**
zifan	...	234	UCLA		UCLA	LA
vincent	...	345	UCLA		UCLA	LA
dan	...	456	UW		UW	seattle
seymour	...	367	**USC**		**NULL**	**NULL**

COUNT
(addr)

name	addr	phone	job		name	addr
remy	...	123	UCLA		UCLA	LA
remy	...	123	UW		UW	seattle
remy	...	123	**USC**		**NULL**	**NULL**
zifan	...	234	UCLA		UCLA	LA
vincent	...	345	UCLA		UCLA	LA
dan	...	456	UW		UW	seattle
seymour	...	367	**USC**		**NULL**	**NULL**

COUNT
(addr)

2

name	addr	phone	job		name	addr
remy	...	123	UCLA		UCLA	LA
remy	...	123	UW		UW	seattle
remy	...	123	**USC**		**NULL**	**NULL**
zifan	...	234	UCLA		UCLA	LA
vincent	...	345	UCLA		UCLA	LA
dan	...	456	UW		UW	seattle
seymour	...	367	**USC**		**NULL**	**NULL**

COUNT
(addr)

2

1

name	addr	phone	job		name	addr	**COUNT** (addr)
remy	...	123	UCLA		UCLA	LA	
remy	...	123	UW		UW	seattle	2
remy	...	123	**USC**		**NULL**	**NULL**	
zifan	...	234	UCLA		UCLA	LA	1
vincent	...	345	UCLA		UCLA	LA	1
dan	...	456	UW		UW	seattle	
seymour	...	367	**USC**		**NULL**	**NULL**	

name	addr	phone	job		name	addr	**COUNT** (addr)
remy	...	123	UCLA		UCLA	LA	2
remy	...	123	UW		UW	seattle	
remy	...	123	**USC**		**NULL**	**NULL**	
zifan	...	234	UCLA		UCLA	LA	1
vincent	...	345	UCLA		UCLA	LA	1
dan	...	456	UW		UW	seattle	1
seymour	...	367	**USC**		**NULL**	**NULL**	

name	addr	phone	job		name	addr	**COUNT** (addr)
remy	…	123	UCLA		UCLA	LA	
remy	…	123	UW		UW	seattle	2
remy	…	123	**USC**		**NULL**	**NULL**	
zifan	…	234	UCLA		UCLA	LA	1
vincent	…	345	UCLA		UCLA	LA	1
dan	…	456	UW		UW	seattle	1
seymour	…	367	**USC**		**NULL**	**NULL**	**0**

The witness problem (*argmax*):

Q: who's the oldest cat?

name	breed	age	origin	kind	person
casa	tabby	8	seattle	cat	remy
kira	tuxedo	6	hawaii	cat	remy
toby	border collie	17	seattle	dog	remy
maya	husky	10	LA	dog	sam

The ORDER BY trick

```
SELECT * FROM R ORDER BY x
```

```
SELECT name, age
  FROM pets
 WHERE kind = "cat"
ORDER BY age
 LIMIT 1
```

Multiple oldest cats?

First find the oldest age...

Then find the cats

First find the oldest age...

```
SELECT max(age)
  FROM pets
 WHERE kind = "cat"
```

Then find the cats

First find the oldest age...

```
SELECT max(age)
  FROM pets
 WHERE kind = "cat"
```

Then find the cats

```
SELECT name, age
  FROM pets
 WHERE kind = "cat"
   AND age = ()
```

```
SELECT name, age
  FROM pets
 WHERE kind = "cat"
   AND age = (SELECT max(age)
                FROM pets
               WHERE kind = "cat")
```

Q: oldest pets per kind?

```
SELECT name, age
  FROM pets
 WHERE kind = "cat"
   AND age = (SELECT max(age)
                FROM pets
               WHERE kind = "cat")
```

```
SELECT name, age
  FROM pets
 WHERE kind = "cat"
   AND age = (SELECT max(age)
                FROM pets
               WHERE kind = "cat")
```

```
SELECT name, age
  FROM pets
 WHERE age = (SELECT max(age)
                FROM pets
               WHERE kind =      )
```

```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE kind =      )
```

```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE p2.kind = p1.kind)
```

name	breed	age	origin	kind	person
casa	tabby	8	seattle	cat	remy
kira	tuxedo	6	hawaii	cat	remy
toby	border collie	17	seattle	dog	remy
maya	husky	10	LA	dog	sam

name	breed	age	origin	kind	person
casa	tabby	8	seattle	cat	remy
kira	tuxedo	6	hawaii	cat	remy
toby	border collie	17	seattle	dog	remy
maya	husky	10	LA	dog	sam


```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE p2.kind = p1.kind)
```

Replace nesting with join

```
SELECT name, age
  FROM pets as p1

WHERE age = (SELECT max(age)
               FROM pets as p2
              WHERE p2.kind = p1.kind)
```

merge

merge

```
SELECT name, age
  FROM pets as p1, pets as p2
 WHERE age = max(age)
   AND p2.kind = p1.kind
```

```
SELECT name, age
  FROM pets as p1, pets as p2
 WHERE p2.kind = p1.kind
HAVING p1.age = max(p2.age)
```

```
SELECT name, age
  FROM pets as p1, pets as p2
 WHERE p2.kind = p1.kind
 GROUP BY p2.kind
HAVING p1.age = max(p2.age)
```

```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE p2.kind = p1.kind)
```

```
SELECT name, age
  FROM pets as p1, pets as p2
 WHERE p2.kind = p1.kind
 GROUP BY p2.kind
HAVING p1.age = max(p2.age)
```

```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE p2.kind = p1.kind)

                SELECT max(p2.age)
                  FROM pets as p2
                 GROUP BY p2.kind
```

```
SELECT name, age
  FROM pets as p1, pets as p2
 WHERE p2.kind = p1.kind
 GROUP BY p2.kind
HAVING p1.age = max(p2.age)
```

```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE p2.kind = p1.kind)

              SELECT max(p2.age)
                FROM pets as p2
               GROUP BY p2.kind
```

```
SELECT name, age
  FROM pets as p1, pets as p2
 WHERE p2.kind = p1.kind
 GROUP BY p2.kind
HAVING p1.age = max(p2.age)
```

```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE p2.kind = p1.kind)

              SELECT max(p2.age)
                FROM pets as p2
               GROUP BY p2.kind
```

```
SELECT name, age
  FROM pets as p1, pets as p2
 WHERE p2.kind = p1.kind
 GROUP BY p2.kind
HAVING p1.age = max(p2.age)
```

```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE p2.kind = p1.kind)

              SELECT max(p2.age)
                FROM pets as p2
               GROUP BY p2.kind
```

```
SELECT name, age
    FROM pets as p1, pets as p2
   WHERE p2.kind = p1.kind
   GROUP BY p2.kind
  HAVING p1.age = max(p2.age)
```

```
SELECT name, age
  FROM pets as p1
 WHERE age = (SELECT max(age)
                FROM pets as p2
               WHERE p2.kind = p1.kind)

              SELECT max(p2.age)
                FROM pets as p2
               GROUP BY p2.kind
```

```
SELECT name, age
  FROM pets as p1, pets as p2
 WHERE p2.kind = p1.kind
 GROUP BY p2.kind
HAVING p1.age = max(p2.age)
```

```
EXISTS (SELECT ...)

x IN (SELECT ...)    x NOT IN (SELECT ...)

x > ANY (SELECT ...)

x >= ALL (SELECT ...)
```

`EXISTS (SELECT ...)` Find people with pets

`x` **`IN (SELECT ...)`**

`x` **`NOT IN (SELECT ...)`** Find strays

`x >` **`ANY (SELECT ...)`**

`x >=` **`ALL (SELECT ...)`** Find oldest pets

Find people with pets

```
SELECT people.name FROM people
 WHERE EXISTS (
   SELECT * FROM pets
    WHERE pets.person = people.name
)
```

Find strays

```
SELECT pets.name FROM pets
 pets.person NOT IN (
   SELECT name FROM people
)
```

Find oldest pets

```
SELECT pets.name FROM pets
 WHERE pets.age >= (
    SELECT max(age) FROM pets
)
```

Q: unnest these?

Find people with pets

```
SELECT people.name FROM people
 WHERE EXISTS (
  SELECT * FROM pets
   WHERE pets.person = people.name
)
```

```
SELECT pets.name FROM pets
 pets.person NOT IN (
  SELECT name FROM people
 )
```

Find people with pets

```
SELECT people.name FROM people
 WHERE EXISTS (
  SELECT * FROM pets
   WHERE pets.person = people.name
)
```

```
SELECT people.name FROM people, pets
 WHERE pets.person = people.name
```

```
SELECT pets.name FROM pets
 pets.person NOT IN (
  SELECT name FROM people
 )
```