SQL: Nulls

The ugly side of SQL

(major headache for query optimization)

Can be a value of any attribute

e.g: branch =

bname	bcity	assets
Downtown	Boston	9M
Perry	Horseneck	1.7M
Mianus	Horseneck	.4M
Waltham	Boston	NULL

What does this mean?

(unknown) We don't know Waltham's assets?

(inapplicable) Waltham has a special kind of account without assets

(withheld) We are not allowed to know

SQL: Nulls

Arithmetic Operations with `Null`

n + NULL = NULL   (similarly for all *arithmetic ops*: +, -, *, /, mod, …)

e.g: branch =

bname	bcity	assets
Downtown	Boston	9M
Perry	Horseneck	1.7M
Mianus	Horseneck	.4M
Waltham	Boston	NULL

```
SELECT bname, assets * 2 as a2
FROM branch
```
=

bname	a2
Downtown	18M
Perry	3.4M
Mianus	.8M
Waltham	NULL

SQL: Nulls

Boolean Operations with `Null`

n < NULL = UNKNOWN   (similarly for all *boolean ops*: >, <=, >=, <>, =, ...)

e.g: branch =

bname	bcity	assets
Downtown	Boston	9M
Perry	Horseneck	1.7M
Mianus	Horseneck	.4M
Waltham	Boston	NULL

```
SELECT *
FROM branch
WHERE assets = NULL
```

=

bname	bcity	assets

Counter-intuitive: NULL * 0 = NULL

Counter-intuitive: select * from movies
                   where length >= 120 or length <= 120

SQL: Nulls

Boolean Operations with `Null`

n < NULL = UNKNOWN   (similarly for all _boolean ops_: >, <=, >=, <>, =, …)

e.g: branch =

bname	bcity	assets
Downtown	Boston	9M
Perry	Horseneck	1.7M
Mianus	Horseneck	.4M
Waltham	Boston	NULL

```
SELECT *
FROM branch
WHERE assets IS NULL
```

=

bname	bcity	assets
Waltham	Boston	NULL

SQL: Unknown

Boolean Operations with `Unknown`

`n < NULL = UNKNOWN`   (similarly for all *boolean ops*: `>, <=, >=, <>, =`, …)

`FALSE OR UNKNOWN = UNKNOWN`

`TRUE AND UNKNOWN = UNKNOWN`

Intuition: substitute each of TRUE, FALSE for unknown. If different answer results, results is unknown

`UNKNOWN OR UNKNOWN = UNKNOWN`

`UNKNOWN AND UNKNOWN = UNKNOWN`

`NOT (UNKNOWN) = UNKNOWN`

Can write:
```
SELECT …
FROM …
WHERE booleanexp IS UNKNOWN
```

UNKNOWN tuples are not included in final result