Aggregation

Aggregate Functions

So far, all SQL expressions have referred to the values in a single row at a time

```
[expression] as [name], [expression] as [name], ...

select [columns] from [table] where [expression] order by [expression];
```

An aggregate function in the [columns] clause computes a value from a group of rows

```
create table animals as
  select "dog" as kind, 4 as legs, 20 as weight union
  select "cat"        , 4        , 10           union
  select "ferret"     , 4        , 10           union
  select "parrot"     , 2        , 6            union
  select "penguin"    , 2        , 10           union
  select "t-rex"      , 2        , 12000;
```

animals:

kind	legs	weight
dog	4	20
cat	4	10
ferret	4	10
parrot	2	6
penguin	2	10
t-rex	2	12000

```
select max(legs) from animals;
```

max(legs)
4

(Demo)

Mixing Aggregate Functions and Single Values

An aggregate function also selects some row in the table to supply the values of columns that are not aggregated. In the case of max or min, this row is that of the max or min value. Otherwise, it is arbitrary.

```
select max(weight), kind from animals;       select max(legs), kind from animals;

select min(kind), kind from animals;         select avg(weight), kind from animals;
```

(Demo)

```
create table animals as
  select "dog" as kind, 4 as legs, 20 as weight union
  select "cat"        , 4        , 10           union
  select "ferret"     , 4        , 10           union
  select "parrot"     , 2        , 6            union
  select "penguin"    , 2        , 10           union
  select "t-rex"      , 2        , 12000;
```

animals:

kind	legs	weight
dog	4	20
cat	4	10
ferret	4	10
parrot	2	6
penguin	2	10
t-rex	2	12000

Discussion Question

What are all the kinds of animals that have the maximal number of legs?

Groups

Grouping Rows

Rows in a table can be grouped, and aggregation is performed on each group

```
[expression] as [name], [expression] as [name], ...

select [columns] from [table] group by [expression] having [expression];
```

The number of groups is the number of unique values of an expression

```
select legs, max(weight) from animals group by legs;
```

legs	max(weight)
4	20
2	12000

legs=4

legs=2

animals:

kind	legs	weight
dog	4	20
cat	4	10
ferret	4	10
parrot	2	6
penguin	2	10
t-rex	2	12000

(Demo)

Selecting Groups

Rows in a table can be grouped, and aggregation is performed on each group

```
[expression] as [name], [expression] as [name], ...

select [columns] from [table] group by [expression] having [expression];
```

A having clause filters the set of groups that are aggregated

```
select weight/legs, count(*) from animals group by weight/legs having count(*)>1;
```

weight/legs	count(*)
5	2
2	2

weight/legs=5
weight/legs=2
weight/legs=2
weight/legs=3
weight/legs=5
weight/legs=6000

animals:

kind	legs	weight
dog	4	20
cat	4	10
ferret	4	10
parrot	2	6
penguin	2	10
t-rex	2	12000

Discussion Question

What's the maximum difference between leg count for two animals with the same weight?

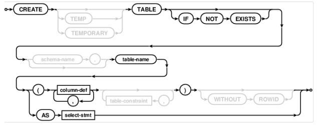
Example: Big Game

(Demo)

Create Table and Drop Table

Create Table

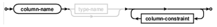
CREATE TABLE expression syntax:



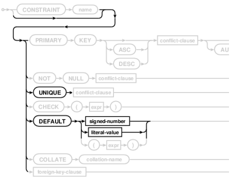
Examples:

```
CREATE TABLE numbers (n, note);
CREATE TABLE numbers (n UNIQUE, note);
CREATE TABLE numbers (n, note DEFAULT "No comment");
```

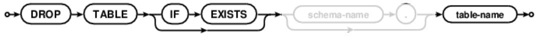
column-def:



column-constraint:

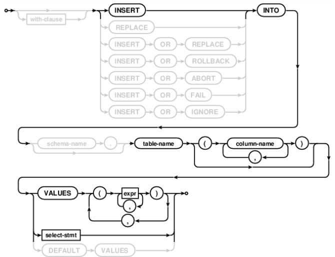


Drop Table



Modifying Tables

Insert



For a table t with two columns...

To insert into one column:

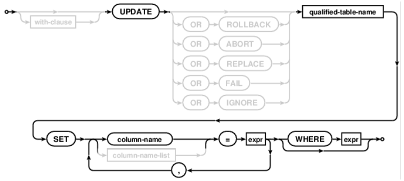
```
INSERT INTO t(column) VALUES (value);
```

To insert into both columns:

```
INSERT INTO t VALUES (value0, value1);
```

(Demo)

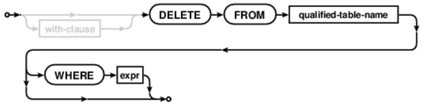
Update



Update sets all entries in certain columns to new values, just for some subset of rows.

(Demo)

Delete



Delete removes some or all rows from a table.

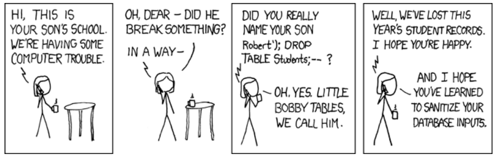
(Demo)

Python and SQL

(Demo)

SQL Injection Attack

A Program Vulnerable to a SQL Injection Attack


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

```
name = "Robert'); DROP TABLE Students; --"
~~cmd = "INSERT INTO Students VALUES ('" + name + "');"~~
~~db.executescript(cmd)~~   db.execute("INSERT INTO Students VALUES (?)", [name])
            INSERT INTO Students VALUES ('Robert''); DROP TABLE Students; --');
            ~~INSERT INTO Students VALUES ('Robert'); DROP TABLE Students; --');~~
```

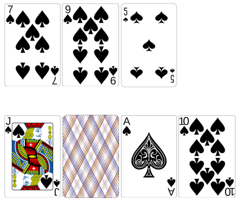
https://xkcd.com/327/

Database Connections

Casino Blackjack

Player:

Dealer:



(Demo)