

# SQL / Data Storage

**cs380**

# Data types

- in SQL standard:
  - char(n)
  - varchar(n)
  - int
  - smallint
  - numeric(p,d)
  - real
  - double precision
  - float(n)

bigint, bit, bit varying, boolean, char, character varying, character, varchar, date, double precision, integer, interval, numeric, decimal, real, smallint, time (with or without time zone), timestamp (with or without time zone), xml.

## In PostgreSQL (not in standard)

Name	Aliases	Description
<code>bigserial</code>	<code>serial8</code>	autoincrementing eight-byte integer
<code>box</code>		rectangular box on a plane
<code>bytea</code>		binary data ("byte array")
<code>cidr</code>		IPv4 or IPv6 network address
<code>circle</code>		circle on a plane
<code>inet</code>		IPv4 or IPv6 host address
<code>json</code>		textual JSON data
<code>jsonb</code>		binary JSON data, decomposed
<code>line</code>		infinite line on a plane
<code>lseg</code>		line segment on a plane
<code>macaddr</code>		MAC (Media Access Control) address
<code>money</code>		currency amount
<code>path</code>		geometric path on a plane

# Null

- Every data type can have a value or null
  - ie no data
  - Monday will include a lot of discussion of handling null
-

# Select — the primary SML in SQL

- General form:
  - SELECT ... ;
  - SQL is NOT case sensitive with respect to things in the language
    - SELECT, select, Select are all the same
  - ; is the ONLY command terminator
    - multiple lines are fine
    - whitespace is not significant
- Examples:
  - select 'abc';
  - select 5=5;
  - select true;
  - select 7='abc'; — error tries to compare integer and string, and cannot convert 'abc' to integer

# Almost Complete Select

```
SELECT selection_list           # What columns to select
                                   # the projection operator from last class
FROM table_list                 # Where to select rows from
                                   # when more than one, cartesian product operator
WHERE constraint                # What conditions rows must satisfy
                                   # the selection operator
GROUP BY grouping_columns       # How to group results
HAVING group constraints        # Like WHERE but on groups
ORDER BY sorting_columns       # How to sort results
LIMIT count;                   # Limit on results
```