```
-- Views in SQL

-- In relational algebra, we used assignment to give a name to intermediate
-- results.  In SQL, we can create something called a "view".  Here's an
-- example.

-- Let's see what's in the student table.
csc343h-dianeh=> select * from student ;
  sid  | firstname |  surname   | campus |   email   | cgpa
-------+-----------+------------+--------+-----------+------
 99132 | Avery     | Marchmount | StG    | avery@cs  | 3.13
 98000 | William   | Fairgrieve | StG    | will@cs   | 4.00
 99999 | Afsaneh   | Ali        | UTSC   | aali@cs   | 2.98
   157 | Leilani   | Lakemeyer  | UTM    | lani@cs   | 3.42
 11111 | Homer     | Simpson    | StG    | doh@gmail | 0.40
(5 rows)

-- Suppose students with a cgpa of at least 3.0 are winners of an award,
-- and we need their sid, first name, and email so we can invite them
-- to an award ceremony.  We can easily find that info:
csc343h-dianeh=> select sid, firstname, email
csc343h-dianeh-> from student
csc343h-dianeh-> where cgpa >= 3.0;
  sid  | firstname |  email
-------+-----------+----------
 99132 | Avery     | avery@cs
 98000 | William   | will@cs
   157 | Leilani   | lani@cs
(3 rows)

-- A view allows us to give a name to that query.  Here's the syntax,
-- with the new keywords highlighted in upper case:
csc343h-dianeh=> CREATE VIEW winners AS
csc343h-dianeh-> select sid, firstname, email
csc343h-dianeh-> from student
csc343h-dianeh-> where cgpa >= 3.0;
CREATE VIEW

-- Now I can use the winners view in any subsequent query, and SQL
-- knows what I mean.  For example:
csc343h-dianeh=> select * from winners;
  sid  | firstname |  email
-------+-----------+----------
 99132 | Avery     | avery@cs
 98000 | William   | will@cs
   157 | Leilani   | lani@cs
(3 rows)

-- Here we use it in a more interesting query:
csc343h-dianeh=> select *
csc343h-dianeh-> from winners natural join took
csc343h-dianeh-> where grade < 80;
  sid  |  name   | contact  | oid | grade
-------+---------+----------+-----+-------
 99132 | Avery   | avery@cs |   1 |    79
 99132 | Avery   | avery@cs |  14 |    39
 99132 | Avery   | avery@cs |  15 |    62
 99132 | Avery   | avery@cs |  34 |    75
 98000 | William | will@cs  |  11 |    79
 98000 | William | will@cs  |   6 |    72
 98000 | William | will@cs  |  16 |    79
 98000 | William | will@cs  |  17 |    79
 98000 | William | will@cs  |  22 |    54
 98000 | William | will@cs  |  31 |    78
```

```
 98000 | William | will@cs  |   9 |    78
   157 | Leilani | lani@cs  |  21 |    71
   157 | Leilani | lani@cs  |  11 |    39
   157 | Leilani | lani@cs  |  34 |    62
   157 | Leilani | lani@cs  |  35 |    75
   157 | Leilani | lani@cs  |   5 |    59
   157 | Leilani | lani@cs  |   6 |    72
   157 | Leilani | lani@cs  |  26 |    71
   157 | Leilani | lani@cs  |  17 |    59
(19 rows)

-- Now let's see another way to define a view, where we define the
-- attribute names as well (as we've been doing in relational algebra).

-- First let's get rid of the view we just defined.  The syntax for that
-- is DROP VIEW.
csc343h-dianeh=> DROP VIEW winners;
DROP VIEW

-- Now we'll define winners just like before, but in additional will
-- specify column names for the three attributes we are selecting:
csc343h-dianeh=> create view winners(sid, name, contact) as
csc343h-dianeh-> select sid, firstname, email
csc343h-dianeh-> from student
csc343h-dianeh-> where cgpa >= 3.0;
CREATE VIEW
csc343h-dianeh=> select * from winners;
  sid  |  name   | contact
-------+---------+----------
 99132 | Avery   | avery@cs
 98000 | William | will@cs
   157 | Leilani | lani@cs
(3 rows)

-- A view is simply a name for a query.  When we define it, SQL remembers
-- the name, and the query that it stands for. At that moment, SQL does *not*
-- evaluate the query.  Only later, when we mention the view in another query,
-- does SQL evaluate it.  This means that the value of the view is always up to
-- date.  It also means the view must be evaluated every single time it is
-- mentioned in a query.
-- The full name for this kind of view is "virtual view".  There is another
-- kind of view called a "materialized view", which we may talk about later.
```