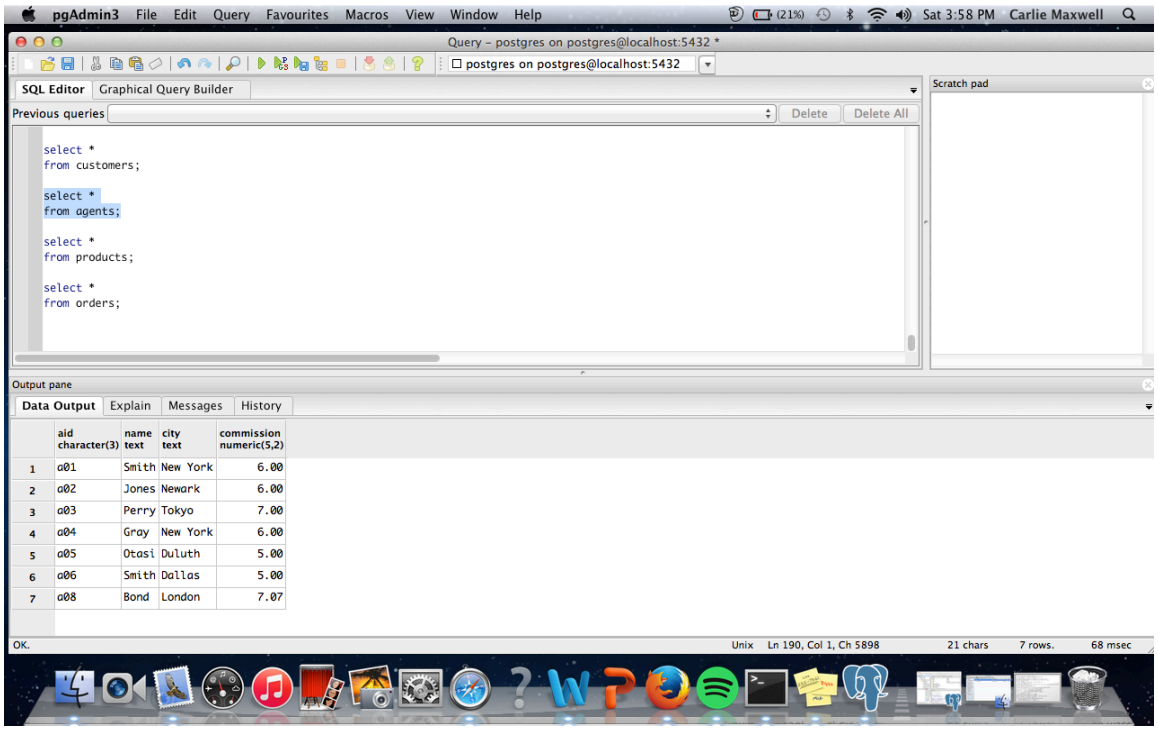
Carlie Maxwell
Lab 2
1/30/16

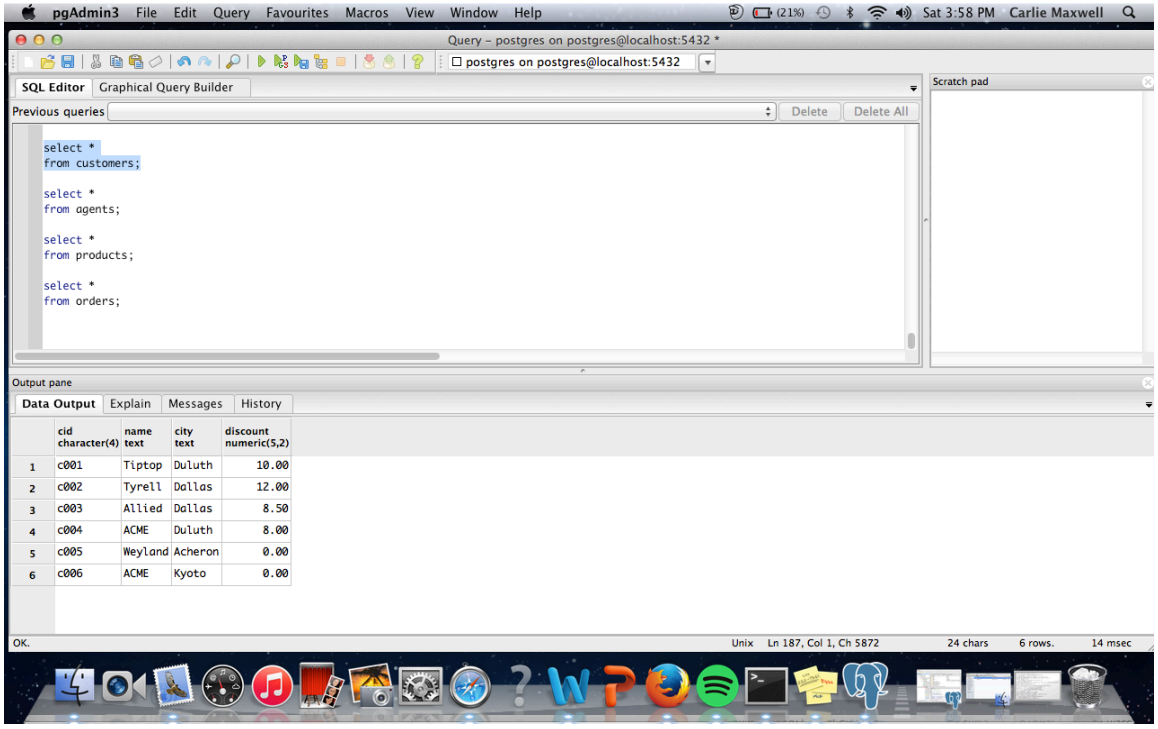
1. A superkey is any column, or set of columns, that uniquely identify every row in a table. For example in the SuperPower table that we created in class, both the SID and description are superkeys because if you say A, it correlates with one row only, and likewise if you say "wit," it only correlates with one row. Both A and wit are only listed once in the table, which allows them to uniquely identify which row you are talking about in that table. A candidate key is essentially a minimal superkey. This means that it is the fewest number of columns inside a table that can uniquely identify every row. In the SuperPower table, the SID itself can accurately identify any row, which means that you can just use that column, instead of both SID and description, to be the candidate key. A primary key is a superkey that you choose to make primary. It's also the minimum number of columns used to uniquely identify every row. You essentially pick one superkey to be the primary key.

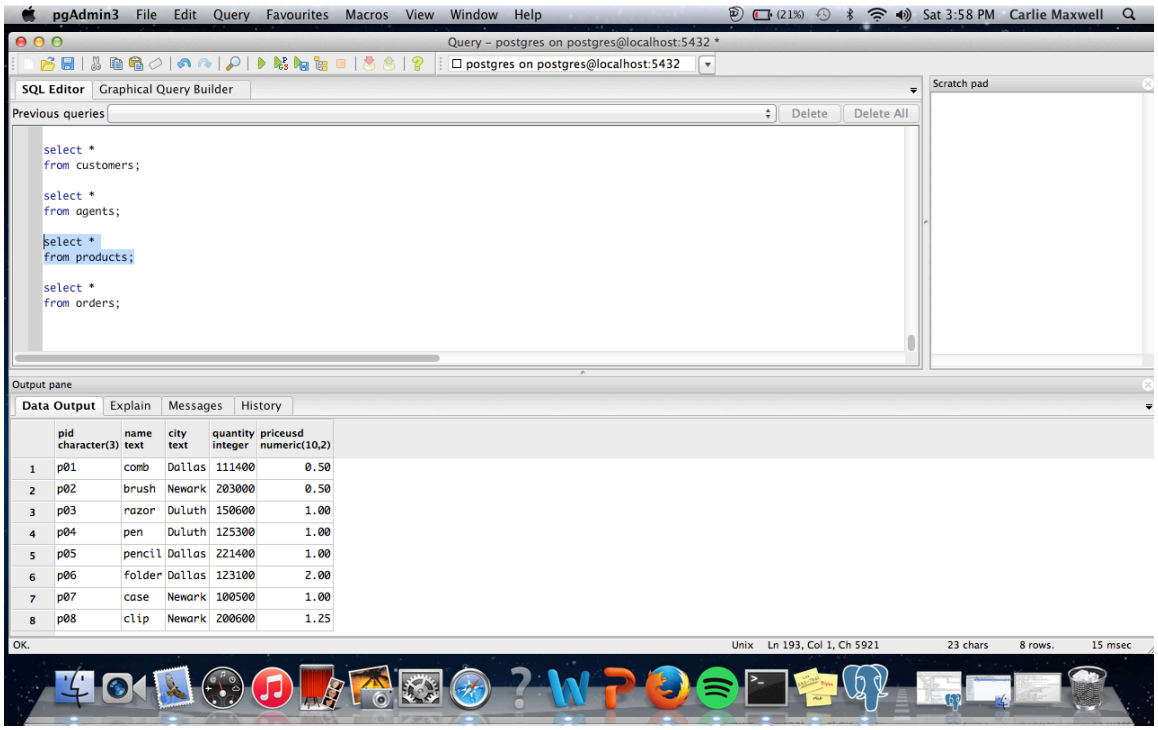
2. The two data types are generic data types and enumerated data types. An enumerated data type means that you have to "spell it out ahead of time." This means that the enumerated type can only contain the values you list in it. If you have the enumerated type, gender, the only 2 values can be male or female, nothing else can be put in "gender." Another example is the enumerated type, Boolean, which only contains true or false. No other values can be used under the Boolean type. On the contrary, the generic data type allows for many possibilities. You can use many different types under a generic class. This gives you the option to use whatever data type is necessary, rather than sticking to the assigned values in an enumerated type. A topic that I could create a table with could be US states. The columns would be the state ID number (SID), the state's name, and their capital. The SID would be a generic data type, as you can list any number. The states' name would be enumerated, as the existing 50 states are all distinguished (you can't make up a name or add any to the 50), and the capital would be enumerated for the reason that every state only has one capital that can't be changed. None of the columns are nullable. Every state has a name, every state has a capital, and every state has to be assigned an ID number.

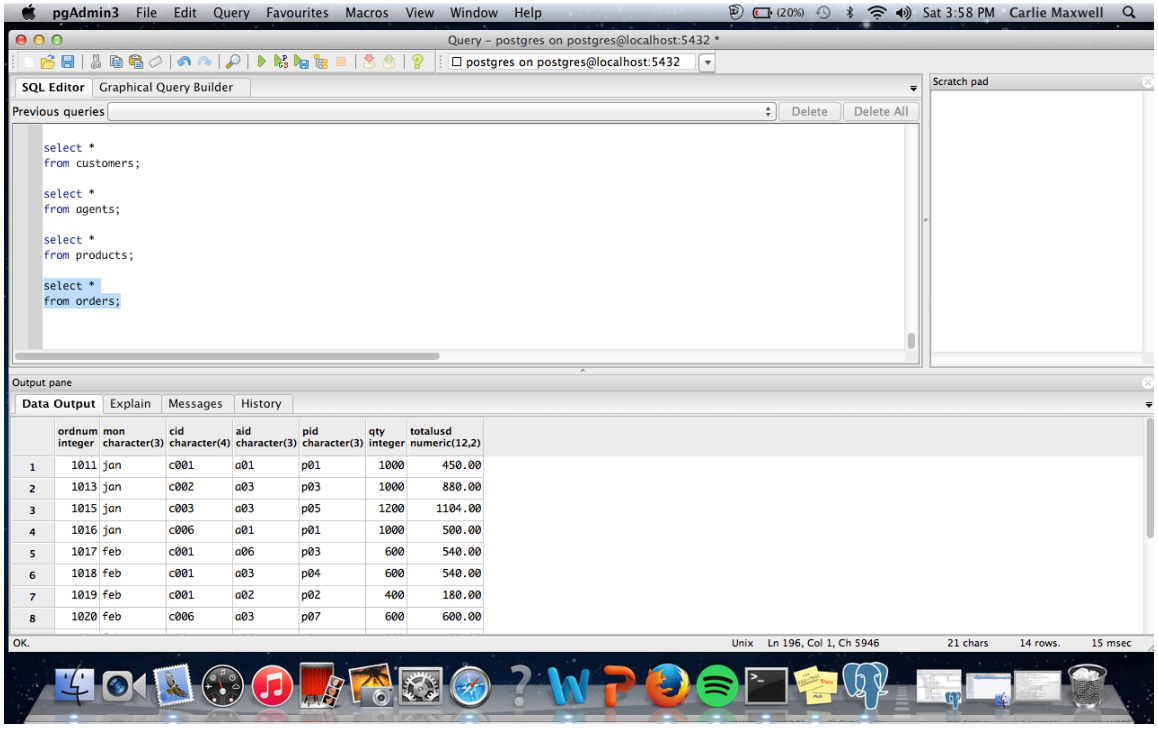
3. a. The "Wirst normal form" rule – In a relational model, columns with multiple attributes are not allowed; the intersect of all rows and columns are atomic, which means the lowest unit. This means that one column can't have more than one thing listed in it. In the SuperPower column, we listed that Sean has the superpower of s = sh and no bullets. This breaks the rule because 2 things are listed under superpowers instead of one. This is an important rule to keep it to 1 attribute because then the computer can find exactly what the user wants when they request a superpower of Sean, instead of deciding between 2.
b. The "access rows by content only" rule – This is also known as the What? Not where? rule. This rule states that you access rows by content only, not where they are. For example, you would say give me the row for Sean, not give me the 1st row.

This is important because it makes it easier on the user, as they may only know what they want but not know exactly where it is located in the database. It leads to more accurate results because the user doesn't have to guess where what they're looking for may be located.
c. The "all rows must be unique" – This means that you can't have duplicate rows. For example, you can't have two rows that contain 007, Sean, s=sh. This is important because if there are 2 identical rows, the computer doesn't know which one to retrieve for the user. By making the rows unique, the user is able to easily take out information from the database without question of which row you're actually talking about.




pgAdmin3 File Edit Query Favourites Macros View Window Help
Sat 3:58 PM Carlie Maxwell
Query - postgres on postgres@localhost:5432 *
postgres on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
Scratch pad
select *
from customers;
select *
from agents;
select *
from products;
select *
from orders;
Output pane
Data Output
Explain
Messages
History
cid character(4)
name text
city text
discount numeric(5,2)
1 c001 Tiptop Duluth 10.00
2 c002 Tyrell Dallas 12.00
3 c003 Allied Dallas 8.50
4 c004 ACME Duluth 8.00
5 c005 Weyland Acheron 0.00
6 c006 ACME Kyoto 0.00
OK.
Unix Ln 187, Col 1, Ch 5872
24 chars
6 rows.
14 msec


pgAdmin3 File Edit Query Favourites Macros View Window Help
Sat 3:58 PM Carlie Maxwell
Query - postgres on postgres@localhost:5432 *
postgres on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
Scratch pad
select *
from customers;
select *
from agents;
select *
from products;
select *
from orders;
Output pane
Data Output
Explain
Messages
History
pid character(3)
name text
city text
quantity integer
priceusd numeric(10,2)
1 p01 comb Dallas 111400 0.50
2 p02 brush Newark 203000 0.50
3 p03 razor Duluth 150600 1.00
4 p04 pen Duluth 125300 1.00
5 p05 pencil Dallas 221400 1.00
6 p06 folder Dallas 123100 2.00
7 p07 case Newark 100500 1.00
8 p08 clip Newark 200600 1.25
OK.
Unix Ln 193, Col 1, Ch 5921
23 chars
8 rows.
15 msec


pgAdmin3 File Edit Query Favourites Macros View Window Help
Sat 3:58 PM Carlie Maxwell
Query - postgres on postgres@localhost:5432 *
postgres on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
Scratch pad
select *
from customers;
select *
from agents;
select *
from products;
select *
from orders;
Output pane
Data Output
Explain
Messages
History
ordnum integer
mon character(3)
cid character(4)
aid character(3)
pid character(3)
qty integer
totalusd numeric(12,2)
1 1011 jan c001 a01 p01 1000 450.00
2 1013 jan c002 a03 p03 1000 880.00
3 1015 jan c003 a03 p05 1200 1104.00
4 1016 jan c006 a01 p01 1000 500.00
5 1017 feb c001 a06 p03 600 540.00
6 1018 feb c001 a03 p04 600 540.00
7 1019 feb c001 a02 p02 400 180.00
8 1020 feb c006 a03 p07 600 600.00
OK.
Unix Ln 196, Col 1, Ch 5946
21 chars
14 rows.
15 msec