Lizeth Sanchez

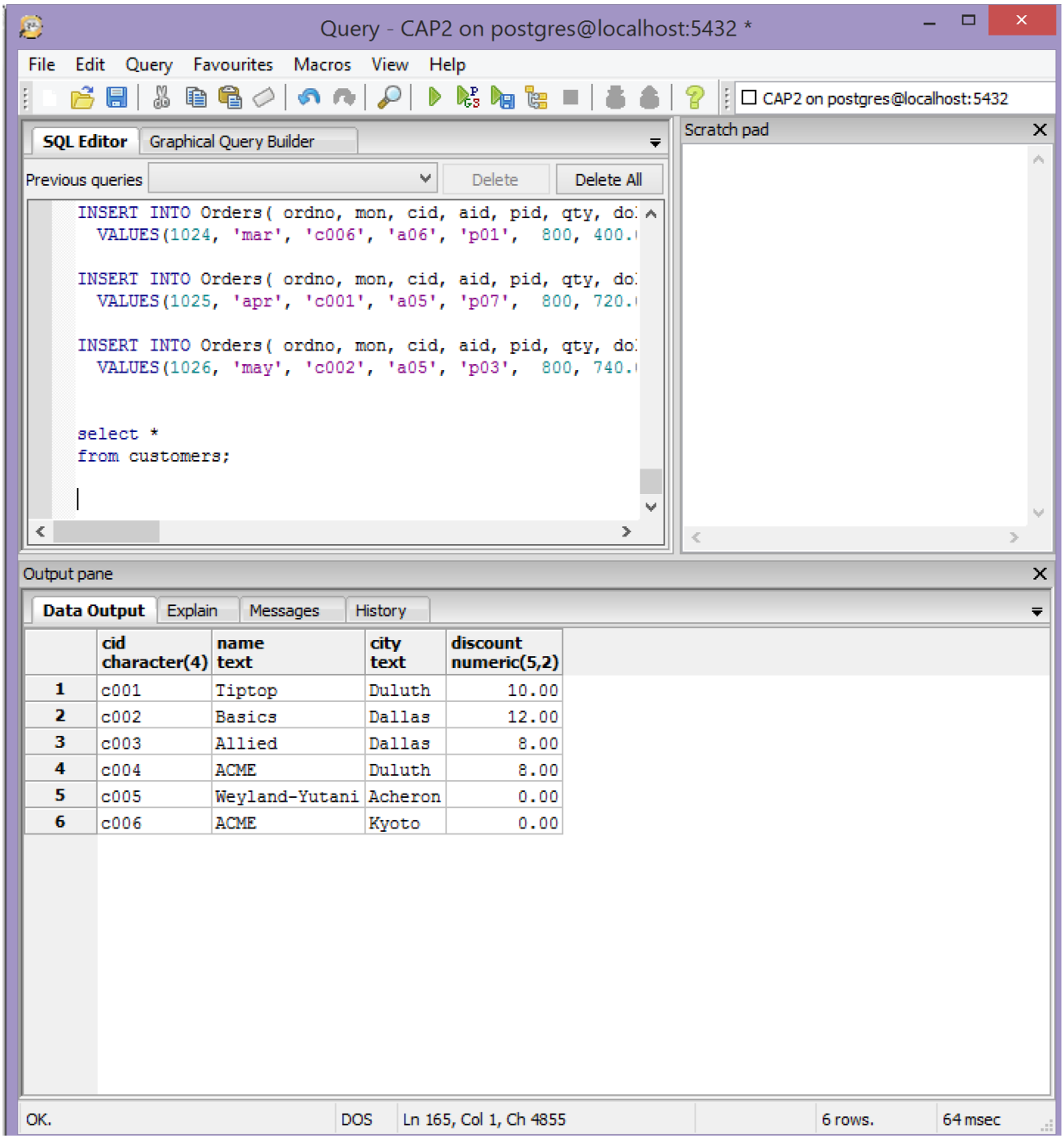
CMPT_308L_200_16S

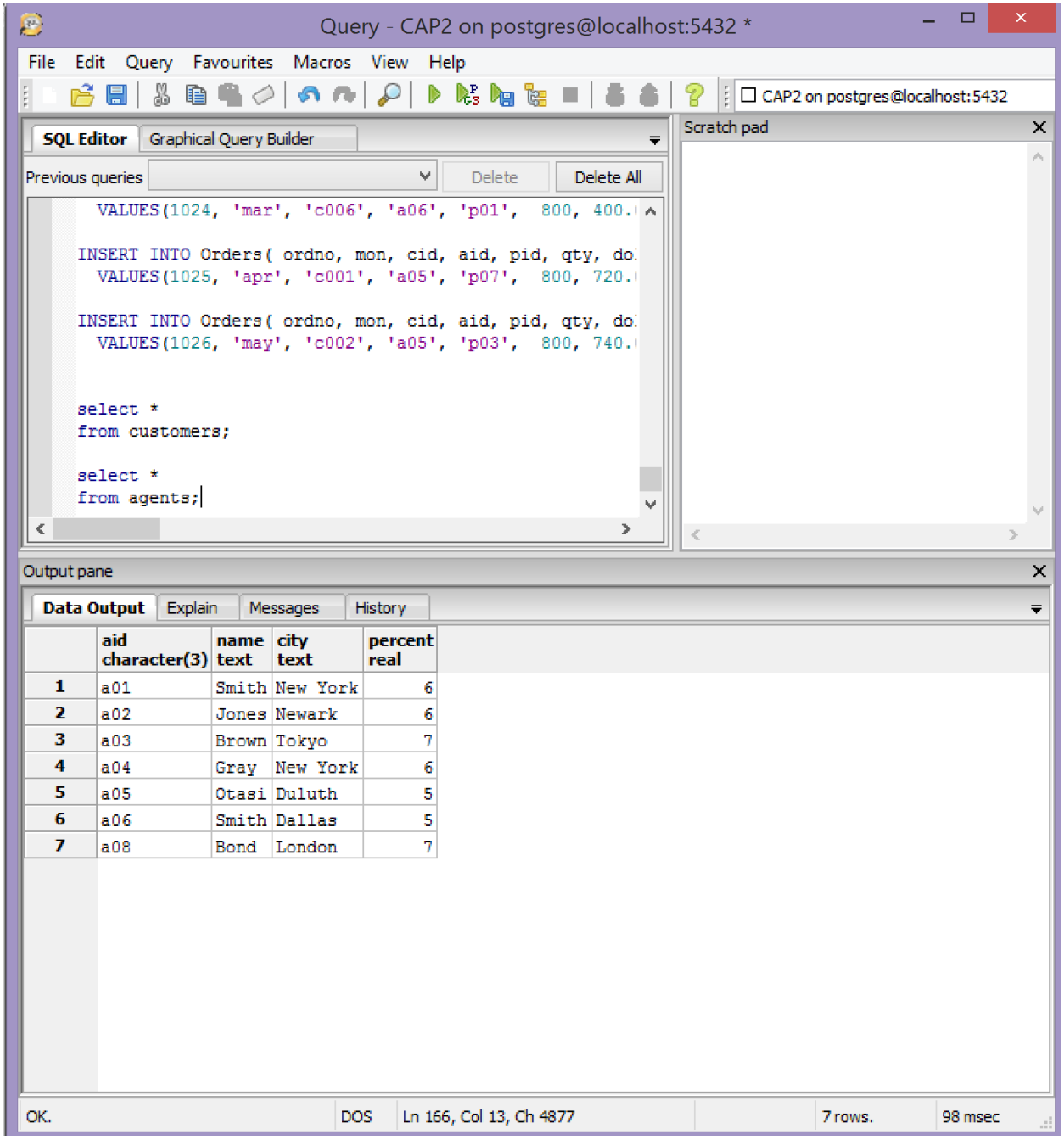
Professor Rivas

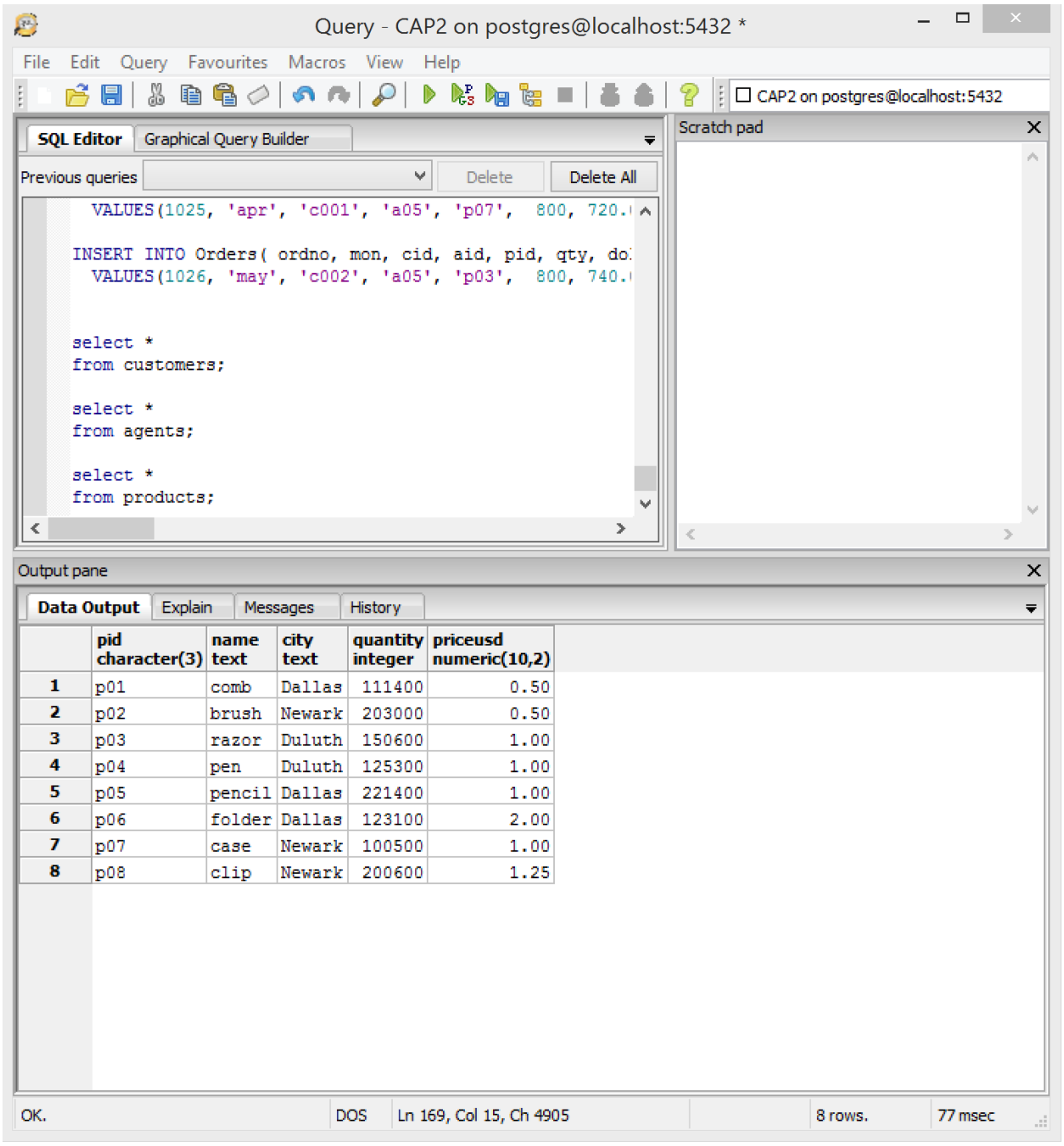
4 February 2016

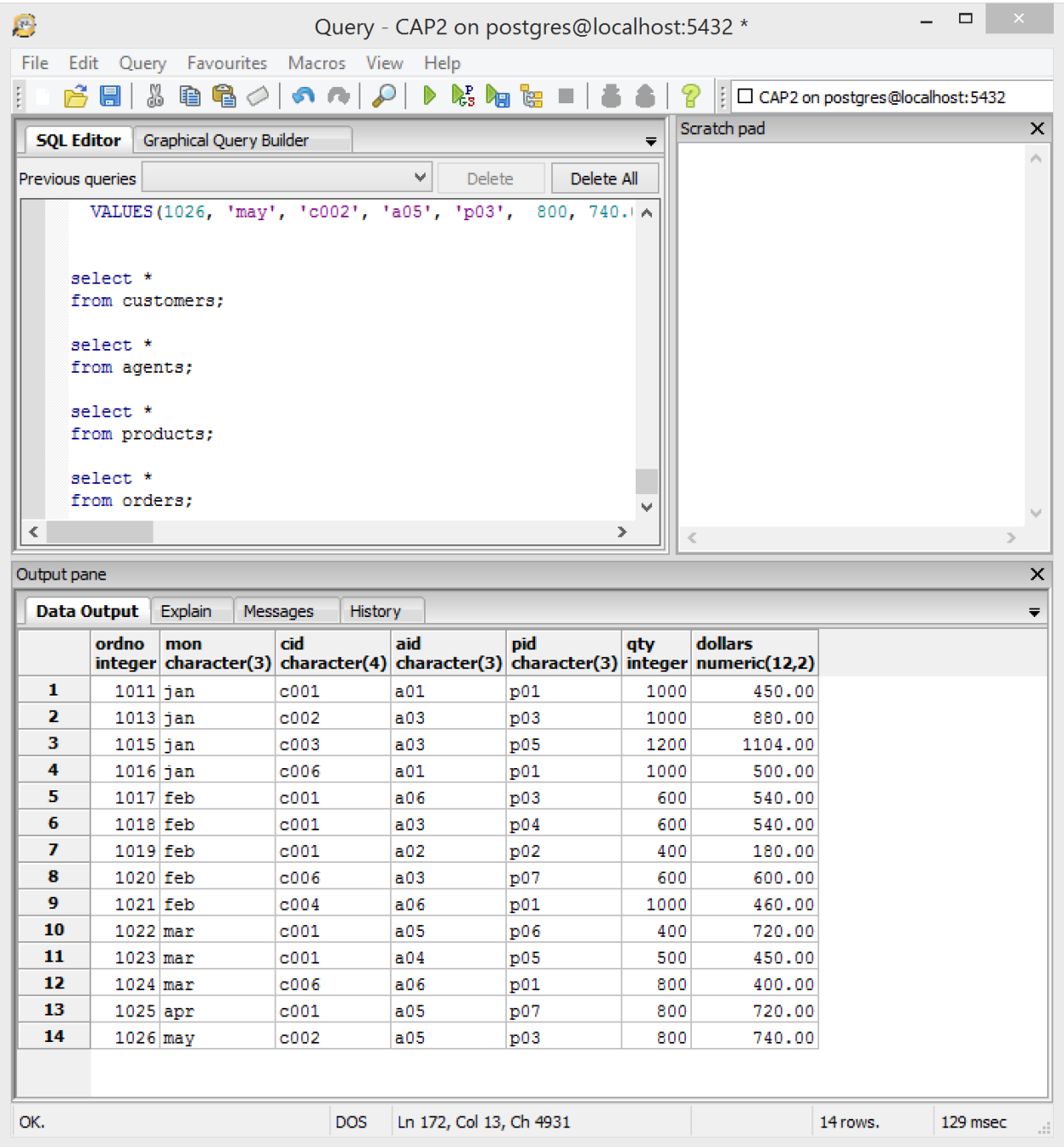
Lab 2

1.




Query - CAP2 on postgres@localhost:5432 *
File Edit Query Favourites Macros View Help
CAP2 on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
VALUES(1024, 'mar', 'c006', 'a06', 'p01', 800, 400.
INSERT INTO Orders(ordno, mon, cid, aid, pid, qty, do
VALUES(1025, 'apr', 'c001', 'a05', 'p07', 800, 720.
INSERT INTO Orders(ordno, mon, cid, aid, pid, qty, do
VALUES(1026, 'may', 'c002', 'a05', 'p03', 800, 740.
select *
from customers;
select *
from agents;
Scratch pad
Output pane
Data Output
Explain
Messages
History
aid character(3)
name text
city text
percent real
1 a01 Smith New York 6
2 a02 Jones Newark 6
3 a03 Brown Tokyo 7
4 a04 Gray New York 6
5 a05 Otasi Duluth 5
6 a06 Smith Dallas 5
7 a08 Bond London 7
OK.
DOS
Ln 166, Col 13, Ch 4877
7 rows.
98 msec


Query - CAP2 on postgres@localhost:5432 *
File Edit Query Favourites Macros View Help
CAP2 on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
VALUES(1025, 'apr', 'c001', 'a05', 'p07', 800, 720.
INSERT INTO Orders(ordno, mon, cid, aid, pid, qty, do
VALUES(1026, 'may', 'c002', 'a05', 'p03', 800, 740.
select *
from customers;
select *
from agents;
select *
from products;
Scratch pad
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.
DOS
Ln 169, Col 15, Ch 4905
8 rows.
77 msec



2. A primary key is a candidate key that is used as a reference and it makes all the rows in a table unique. A candidate key is the minimum set of superkeys that makes the rows of a table unique. A superkey is a set of columns in a table that makes every row unique and is used to identify mistakes in Normal Form (Rule I).

3. There are different kinds of data types in a table. One type is FLOAT, or floating point, which is a number with a fixed decimal. Another type is DATE and TIME which denote date and time. A third data type is character string CHAR which is a fixed-length string and character string VARCHAR where the

string-length is used. A fourth data type is bit strings which use strings of bits instead of characters. A fifth type is BOOLEAN which implies a logical value like TRUE and FALSE. A sixth type is INTEGER or INT which is used for integer values.

I would create a table for the Agents' recorded working time. The table name would be Salary and its fields would be Agent ID (aid), Agent Name (name), Time In (timein), Time Out (timeout), Total Time (totalhours), and Money Earned (earnedUSD). The type for Agent ID (aid) would be INTEGER. The type for Agent Name (name) would be a VARCHAR character string. The type for Time In (timein) would be TIME. The type for Time Out (timeout) would also be TIME. The type for Total Time (totalhours) would be INTEGER. The type for Money Earned (earnedUSD) would be FLOAT.

4. The three relational rules for databases are important because they help us better organize our data in a way that we can actually use. The first rule is the "first normal form" rule. This rule follows the "atomicity" part of the ACID test. Through this rule we make sure that data is placed in such a way that it is isolated. By doing this, we can separate data into different tables and call it individually. The second rule is the "access rows by content only" rule. This rule says that we can ask what a piece of data is but not where it is. By this we mean that the placement or order of the data should not matter when trying to find the data. If it matters, then our design is probably flawed. The third rule is that all rows must be unique. This is important because having multiple rows with the same name will cause problems when trying to find and call the data. This can lead to errors and incorrect information that once again says our design is flawed.