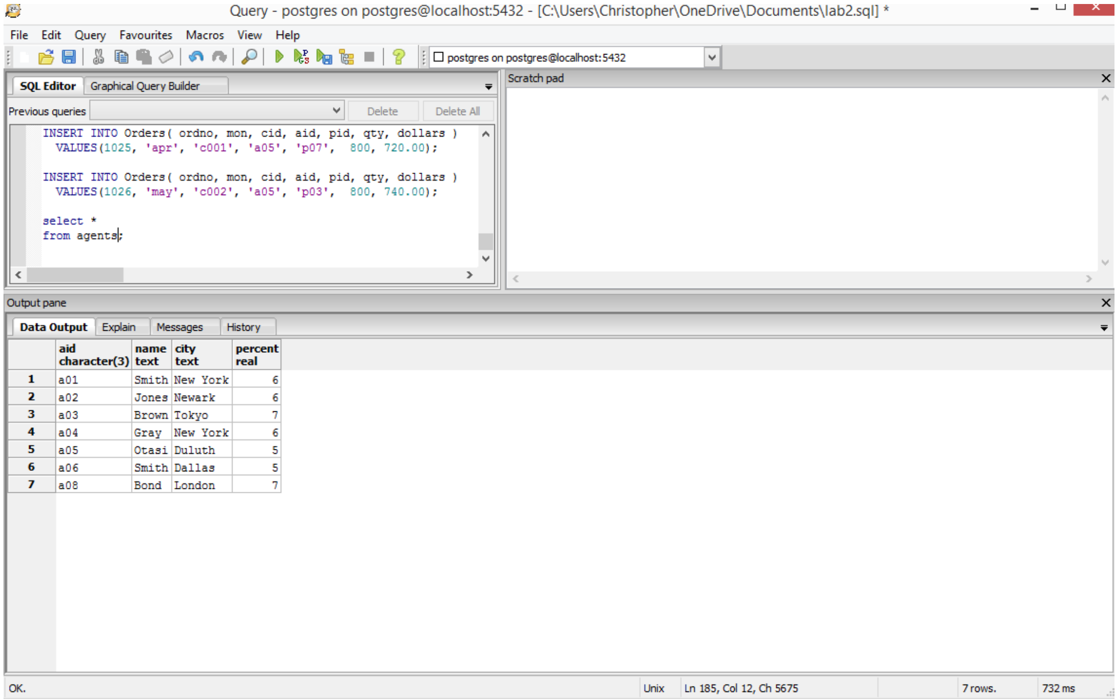
Chris Koriniskie

February 2, 2015

Database Systems Lab 2

1.

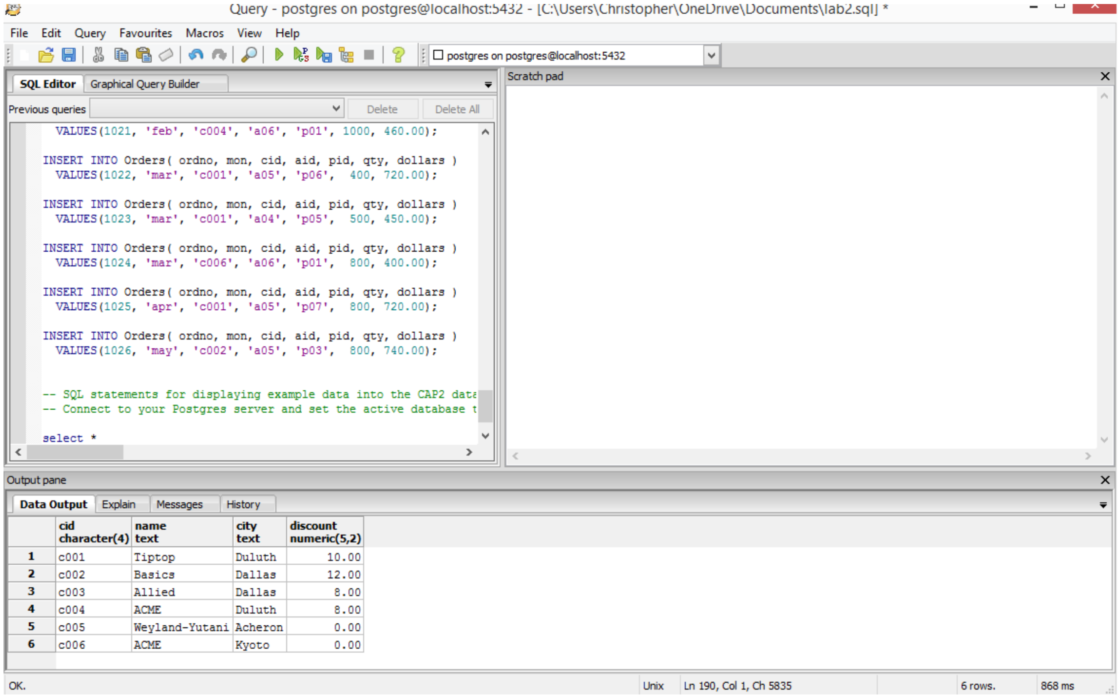


Query - postgres on postgres@localhost:5432 - [C:\Users\Christopher\OneDrive\Documents\lab2.sql] *

```
INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1025, 'apr', 'c001', 'a05', 'p07',  800, 720.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1026, 'may', 'c002', 'a05', 'p03',  800, 740.00);

select *
from agents;
```

	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. Unix Ln 185, Col 12, Ch 5675 7 rows. 732 ms

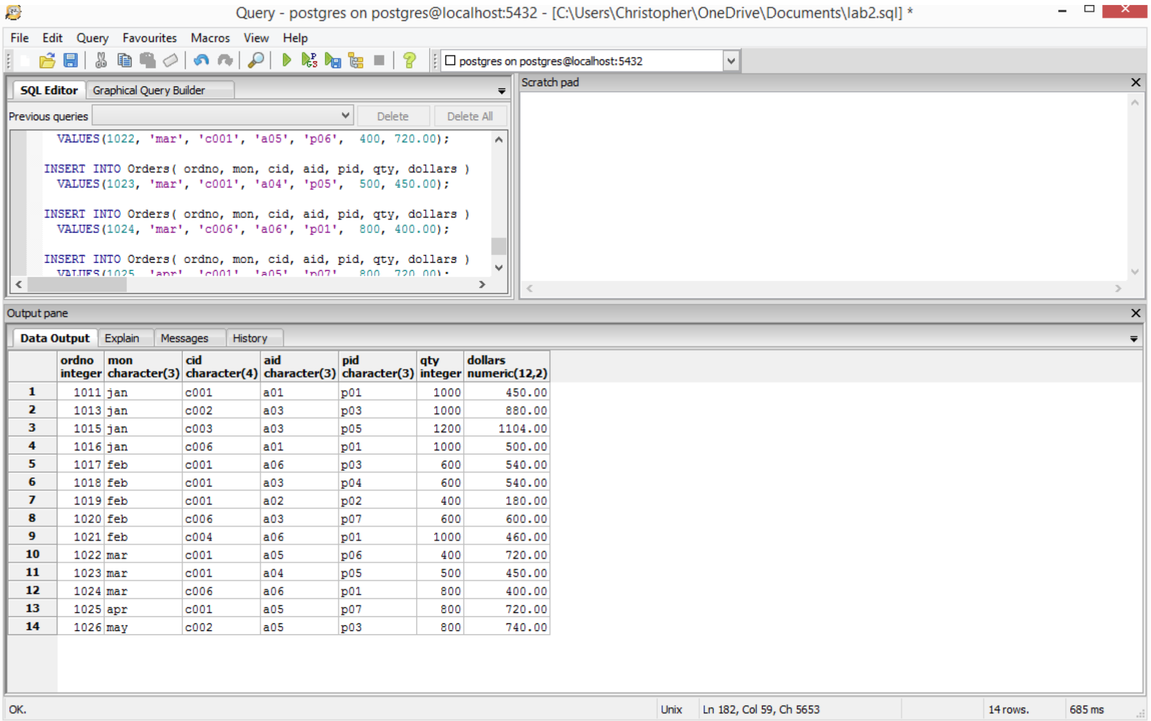


Query - postgres on postgres@localhost:5432 - [C:\Users\Christopher\OneDrive\Documents\lab2.sql] *

```
  VALUES(1021, 'feb', 'c004', 'a06', 'p01', 1000, 460.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1022, 'mar', 'c001', 'a05', 'p06',  400, 720.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1023, 'mar', 'c001', 'a04', 'p05',  500, 450.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1024, 'mar', 'c006', 'a06', 'p01',  800, 400.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1025, 'apr', 'c001', 'a05', 'p07',  800, 720.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1026, 'may', 'c002', 'a05', 'p03',  800, 740.00);


-- SQL statements for displaying example data into the CAP2 data
-- Connect to your Postgres server and set the active database t

select *
```

	cid character(4)	name text	city text	discount numeric(5,2)
1	c001	Tiptop	Duluth	10.00
2	c002	Basics	Dallas	12.00
3	c003	Allied	Dallas	8.00
4	c004	ACME	Duluth	8.00
5	c005	Weyland-Yutani	Acheron	0.00
6	c006	ACME	Kyoto	0.00

OK. Unix Ln 190, Col 1, Ch 5835 6 rows. 868 ms

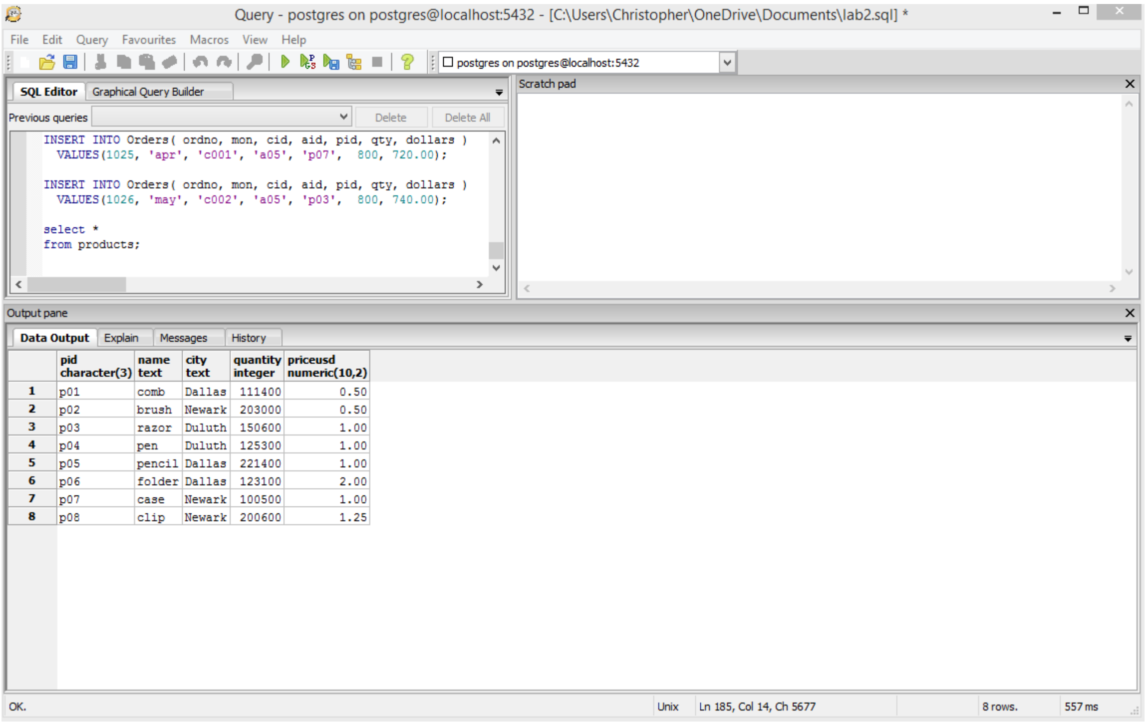


Query - postgres on postgres@localhost:5432 - [C:\Users\Christopher\OneDrive\Documents\lab2.sql] *
File Edit Query Favourites Macros View Help
postgres on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
Scratch pad
Output pane
Data Output
Explain
Messages
History

```
  VALUES(1022, 'mar', 'c001', 'a05', 'p06',  400, 720.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1023, 'mar', 'c001', 'a04', 'p05',  500, 450.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1024, 'mar', 'c006', 'a06', 'p01',  800, 400.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1025, 'apr', 'c001', 'a05', 'p07',  800, 720.00);
```

	ordno integer	mon character(3)	cid character(4)	aid character(3)	pid character(3)	qty integer	dollars 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
9	1021	feb	c004	a06	p01	1000	460.00
10	1022	mar	c001	a05	p06	400	720.00
11	1023	mar	c001	a04	p05	500	450.00
12	1024	mar	c006	a06	p01	800	400.00
13	1025	apr	c001	a05	p07	800	720.00
14	1026	may	c002	a05	p03	800	740.00

OK. Unix Ln 182, Col 59, Ch 5653 14 rows. 685 ms



Query - postgres on postgres@localhost:5432 - [C:\Users\Christopher\OneDrive\Documents\lab2.sql] *
File Edit Query Favourites Macros View Help
postgres on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
Scratch pad
Output pane
Data Output
Explain
Messages
History

```
INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1025, 'apr', 'c001', 'a05', 'p07',  800, 720.00);

INSERT INTO Orders( ordno, mon, cid, aid, pid, qty, dollars )
  VALUES(1026, 'may', 'c002', 'a05', 'p03',  800, 740.00);

select *
from products;
```

	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 185, Col 14, Ch 5677 8 rows. 557 ms

2. The primary key uniquely identifies each record in the table. It can be a single attribute or multiple attributes. It can be a unique id number used that cannot be duplicated. The primary key will guarantee only one result. Candidate key is a sort of sub-category of primary key. They may combine more than one piece of information to make up the id of that key. Superkey is a set of columns that uniquely identify any row in the RDBMS. It is a set of attributes within a set of values that identifies it.

3. Data types will identify what value and classification a piece of information may be. Examples of data types would be integers, booleans, text, characters, and numbers. A situation where you may create a table could be any inventory system. Lets says a grocery store wants to keep track of their inventory. They may have a table called inventory or products. The fields they may want to include would be an id number for the product (char), the name of the product (text) quantiy of the item (how much is in stock…integer), quantity sold within a certain period (maybe monthly….integer), they may also want to put wholesale cost and resale cost of product (numeric as well as what currency it is). Usually the primary key is going to be not null because you cannot leave that field empty. In this case product id is going to be the primary key. This field has to have a number in it to uniquely identify the product. No other product will have the same id which will keep them all unique.

4. First normal form is the condition that every element of a tuple has an atomic value. This means that it cannot be divided into smaller pieces. The benefits are that there is no repeating groups. Every table has one value. An example is if there was a field that had a customer's phone number you could not put 2 numbers into one field. Accesing rows by content only I am assuming means you can be very specific in what you are looking for. If you need certain content like number of items sold of a certain product you would be able to get that information easy. All rows must be unique will ensure that there is a distinguishment between the content being entered. You would not want two rows with customer id numbers because then you wouldn't know which customer id number would be the correct one or the primary key.