LAB 2

1.

Customers Table

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

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	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

Agent Table

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

select *
from customers;

select *
from agents;

select *
from products;

select *
from orders;
```

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

Product Table

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

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

Orders Table

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

select *
from customers;

select *
from agents;

select *
from products;

select *
from orders;
```

Output pane

Data Output | Explain | Messages | History

	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

2. Explain the distinctions among the terms primary key, candidate key, and super key

A super key is any set of columns that uniquely identify every row in the table. When choosing a super key, no two sets can be identical. A super key can be all the columns of table. The candidate key is a minimal super key with the smallest number of columns that keeps a row unique. It is like the "lowest common denominator" of super keys. The candidate key can be one or more columns. A primary key is the candidate key that is chosen by the designer in order to enforce relational rules 2 and 3; that all rows must be unique and in turn a set where order does not matter.

3. Write a short essay on data types. Select a topic for which you might create a table. Name the table and list its fields (columns). For each field, give its data type and whether or not it is nullable.

There are six main types of data. The first and most common form of data are character strings. Character strings are just streams of characters that can either be fixed or vary in length. Simply put, this is just text. An example of a character string can be "moose", "M", or "yes". Although though numbers can technically be a character string, that is not recommended; the reason being that characters cannot have be in mathematical expressions. The simplest form of numbers type are INTs or Integers. An integer is any whole number that is positive, negative or zero. With integers we can query the data using such mathematical expressions as <, >, <=, !=… Sometimes we want to save numbers that contain decimals, especially when precision matters. This is when we can save float data. Floats follow the same rules as integers but can also be manipulated using their decimals (e.g. "mod"). The fourth kind of common data is the Boolean. Boolean denotes an attribute whose value is logical. The possible values are true, false, and unknown. The final two types of data are date and time. These are enumerated domains.

I could create a table called Clash of Clans building guide. This would be useful to those who want to plan how to utilize their builders in Clash of Cans. The table would be called "Buildings_Guide'. There would be six columns: (1) Town Hall Level [**text**], (2) Building [**text**], (3) Building Level [**INT**], (4) Cost (GoldOrElixer) [**INT**], (5) TimeToBuild (DDHH) [**TIME**], (6)GemPrice (gems) [**INT**]. The candidate key Building and Building Level. This is because there can never be duplicates of those two combine. None the fields are able to be null. Each one always has a value.

3. Explain the following relational "rules" with examples and reasons why they are important.

a. The "first normal" rule

This rule states that in an intersection of a row and column there can only be one value. This ensures that each entity in itself is anatomic and is not a set in itself. This is ensures the relational rules between the data and the set that it is in

b. The "access rows by content only" rule

One of the properties of a set is that it has no ordinal value. In order to keep the data in a set relationship, the data cannot be accessed by its position in a table. For example, you cannot query the third row on a table.

c. The "all rows must be unique" rule

This is vital because if we cannot access data by its position in a table must be unique. If each row was not unique there would be no way of querying for a specific row of data.