Hannah Youssef

Dr. Alan Labouseur

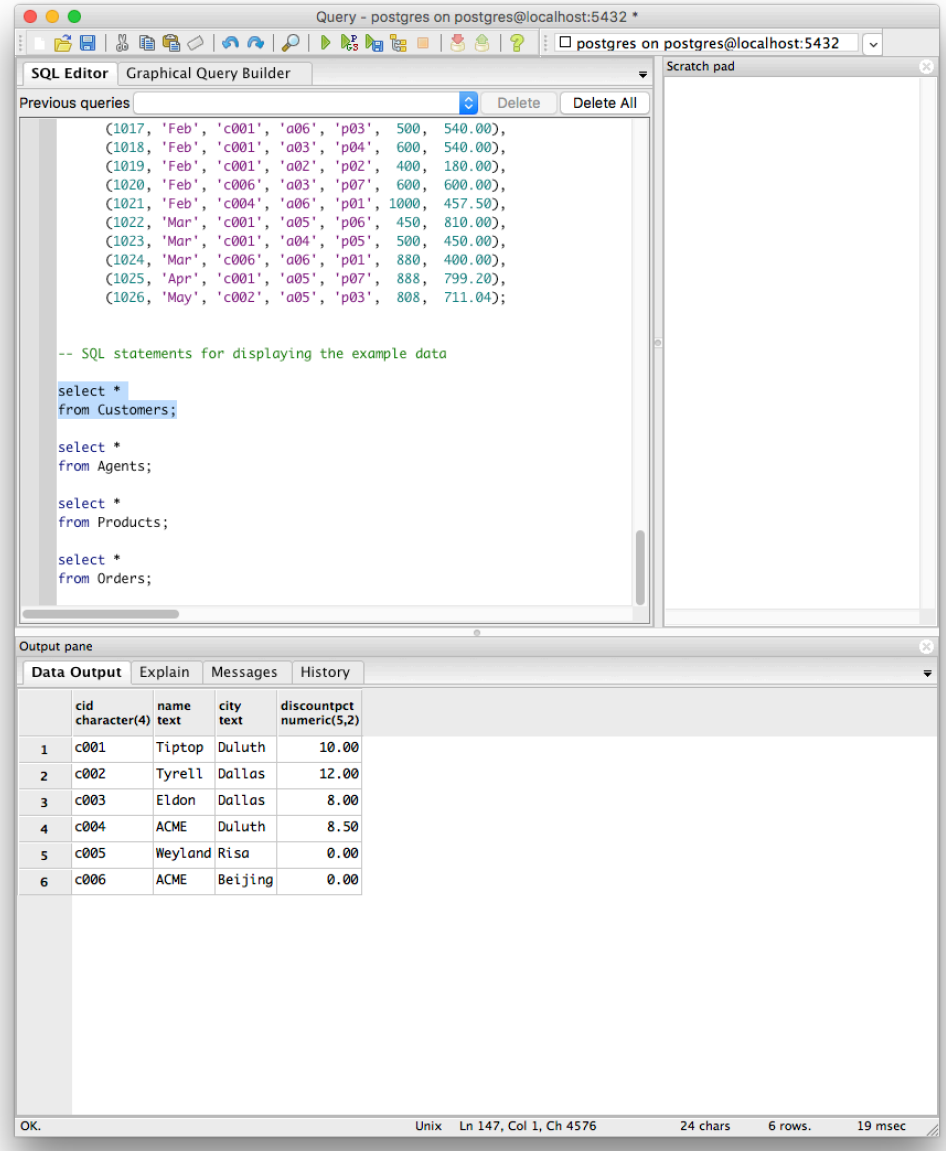
Database Management

11 September 2017

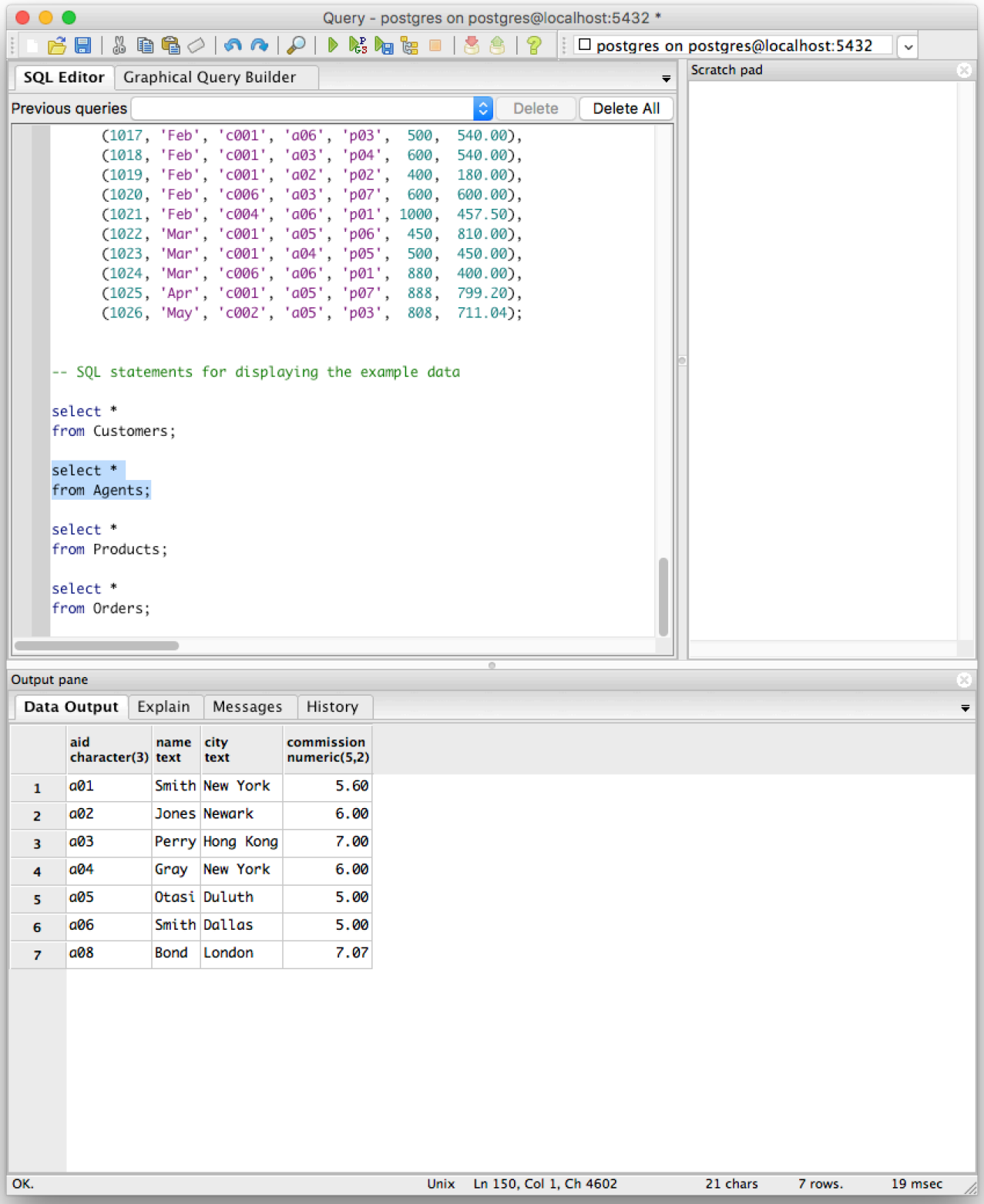
Lab 2 – CAP Database

1. pgAdmin3 Queries screenshots

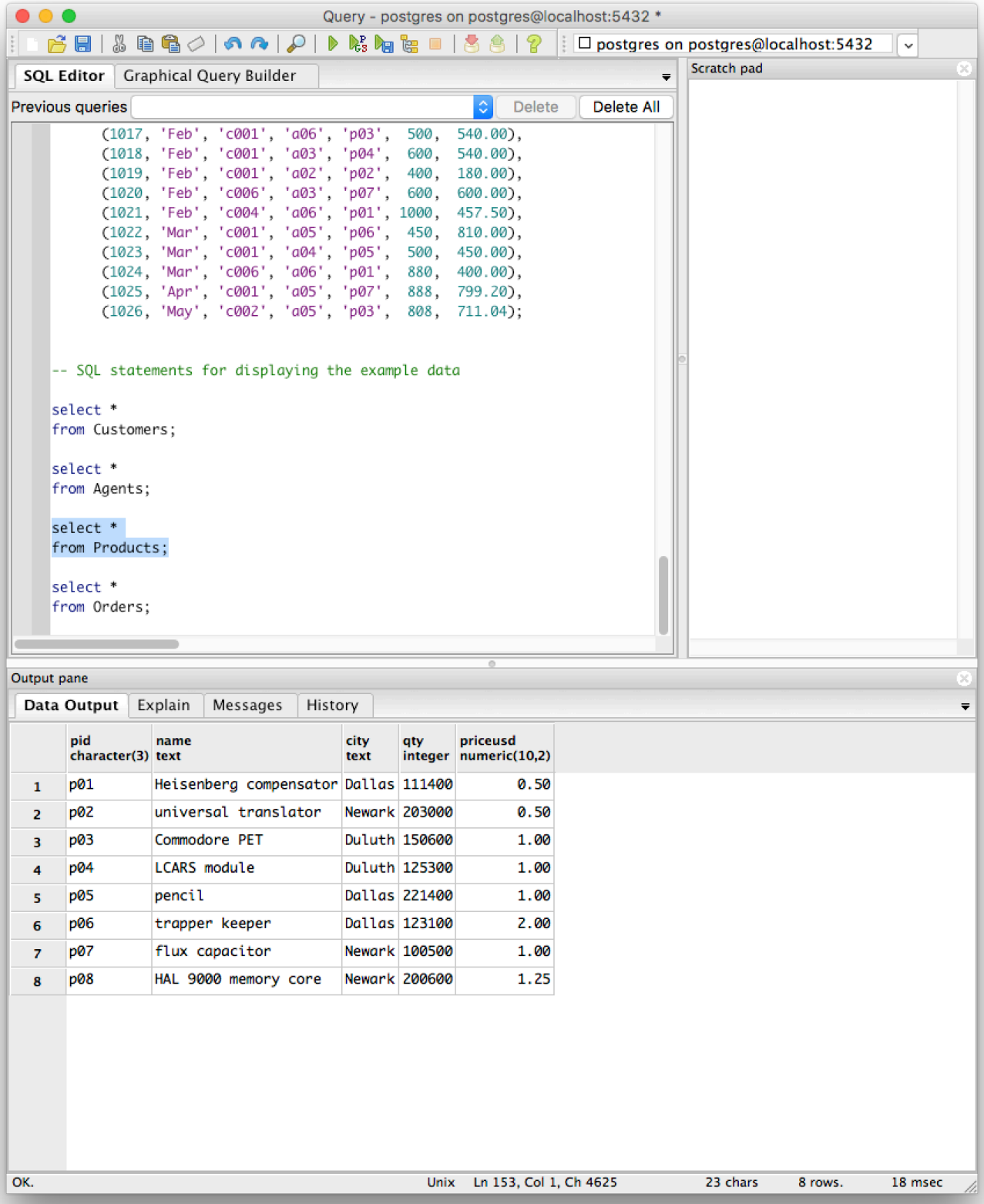
Customers



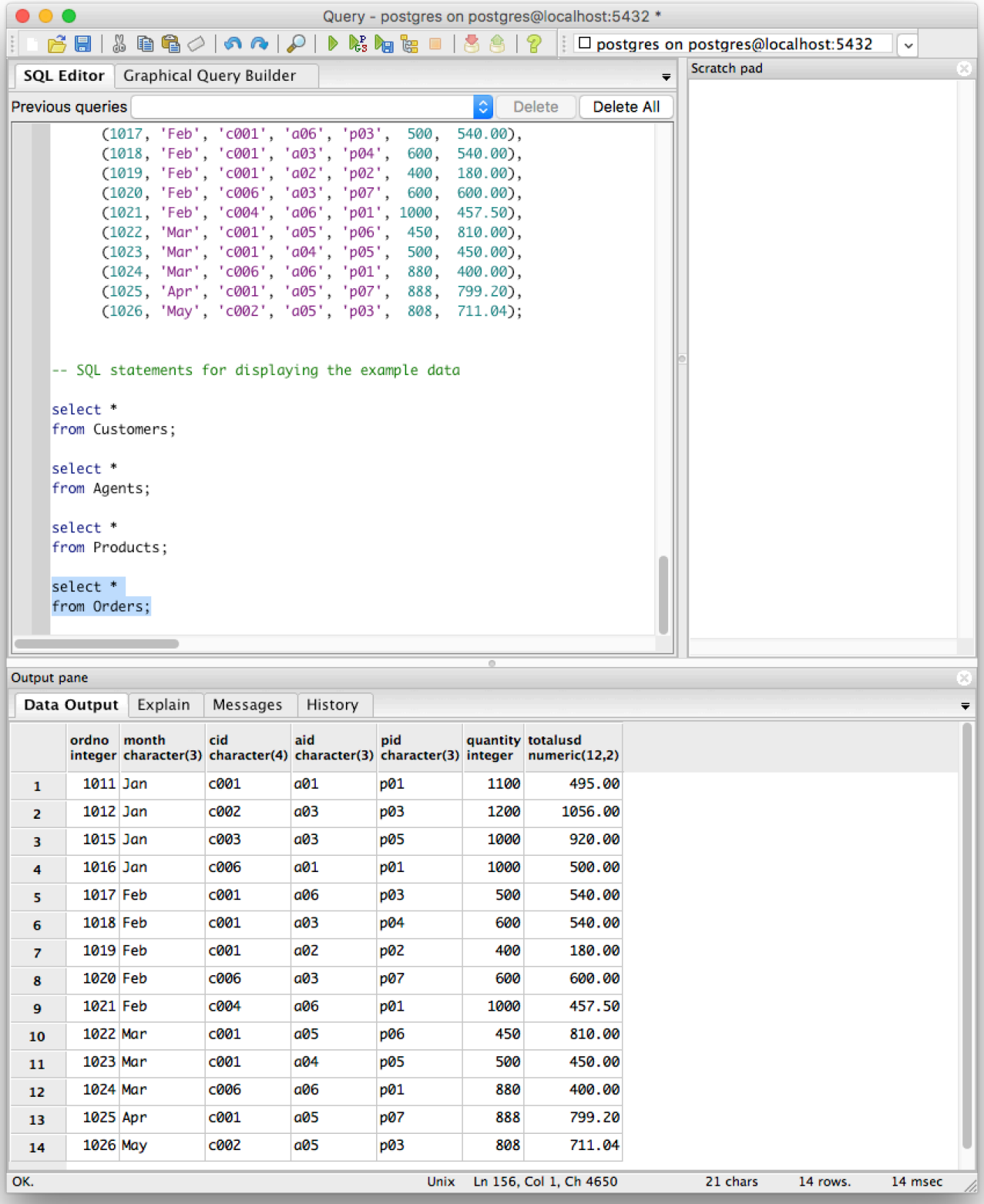
Agents


Query - postgres on postgres@localhost:5432 *
postgres on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
Scratch pad
(1017, 'Feb', 'c001', 'a06', 'p03', 500, 540.00),
(1018, 'Feb', 'c001', 'a03', 'p04', 600, 540.00),
(1019, 'Feb', 'c001', 'a02', 'p02', 400, 180.00),
(1020, 'Feb', 'c006', 'a03', 'p07', 600, 600.00),
(1021, 'Feb', 'c004', 'a06', 'p01', 1000, 457.50),
(1022, 'Mar', 'c001', 'a05', 'p06', 450, 810.00),
(1023, 'Mar', 'c001', 'a04', 'p05', 500, 450.00),
(1024, 'Mar', 'c006', 'a06', 'p01', 880, 400.00),
(1025, 'Apr', 'c001', 'a05', 'p07', 888, 799.20),
(1026, 'May', 'c002', 'a05', 'p03', 808, 711.04);
-- SQL statements for displaying the example data
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
commission numeric(5,2)
1 a01 Smith New York 5.60
2 a02 Jones Newark 6.00
3 a03 Perry Hong Kong 7.00
4 a04 Gray New York 6.00
5 a05 Otasi Duluth 5.00
6 a06 Smith Dallas 5.00
7 a08 Bond London 7.07
OK.
Unix Ln 150, Col 1, Ch 4602
21 chars
7 rows.
19 msec

Products


Query - postgres on postgres@localhost:5432 *
postgres on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
Scratch pad
(1017, 'Feb', 'c001', 'a06', 'p03', 500, 540.00),
(1018, 'Feb', 'c001', 'a03', 'p04', 600, 540.00),
(1019, 'Feb', 'c001', 'a02', 'p02', 400, 180.00),
(1020, 'Feb', 'c006', 'a03', 'p07', 600, 600.00),
(1021, 'Feb', 'c004', 'a06', 'p01', 1000, 457.50),
(1022, 'Mar', 'c001', 'a05', 'p06', 450, 810.00),
(1023, 'Mar', 'c001', 'a04', 'p05', 500, 450.00),
(1024, 'Mar', 'c006', 'a06', 'p01', 880, 400.00),
(1025, 'Apr', 'c001', 'a05', 'p07', 888, 799.20),
(1026, 'May', 'c002', 'a05', 'p03', 808, 711.04);
-- SQL statements for displaying the example data
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
qty integer
priceusd numeric(10,2)
1 p01 Heisenberg compensator Dallas 111400 0.50
2 p02 universal translator Newark 203000 0.50
3 p03 Commodore PET Duluth 150600 1.00
4 p04 LCARS module Duluth 125300 1.00
5 p05 pencil Dallas 221400 1.00
6 p06 trapper keeper Dallas 123100 2.00
7 p07 flux capacitor Newark 100500 1.00
8 p08 HAL 9000 memory core Newark 200600 1.25
OK.
Unix
Ln 153, Col 1, Ch 4625
23 chars
8 rows.
18 msec

Orders



	ordno integer	month character(3)	cid character(4)	aid character(3)	pid character(3)	quantity integer	totalusd numeric(12,2)
1	1011	Jan	c001	a01	p01	1100	495.00
2	1012	Jan	c002	a03	p03	1200	1056.00
3	1015	Jan	c003	a03	p05	1000	920.00
4	1016	Jan	c006	a01	p01	1000	500.00
5	1017	Feb	c001	a06	p03	500	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	457.50
10	1022	Mar	c001	a05	p06	450	810.00
11	1023	Mar	c001	a04	p05	500	450.00
12	1024	Mar	c006	a06	p01	880	400.00
13	1025	Apr	c001	a05	p07	888	799.20
14	1026	May	c002	a05	p03	808	711.04

2. **Primary key, Candidate key, and Super key**

A primary key is the attribute or column that uniquely identifies a row in a database. A table can only have one primary key. The candidate key is the attribute or column in a table that qualifies for uniqueness of each row. One table can have more than one candidate keys. A primary key is actually a candidate key that is the most suited to maintain uniqueness in a table at the row level. If you add any other column to a candidate key then it becomes a super key. A super key is a combination of attributes that can be uniquely used to identify a database record. A table can have many super keys. Candidate keys are a special subset of super keys that do not have an extraneous information in them.

3. **Data types**

This table example is called “Students in Database Management”.

The fields this table has are:

- firstName
    - data type: text
    - Nullable
- lastName
    - data type: text
    - Nullable
- Cwid
    - data type: int
    - Nullable

- classYear
    - data type: smallint
    - Nullable
- classStartDate
    - data type: datetime
    - Not nullable

4. Relational rules

a. The first normal form (1NF) rule is where data in each cell must be atomic and unstructured, meaning it cannot be broken down into more parts. An example of this demonstrating how this is important is if there is a field name called “fullName”. This field name can be broken down into “firstName” and “lastName” making the data atomic.

b. The access rows by content only rule means whenever addressing the contents of a table, you have to ask *what* content is there, rather than *where* it is. A table is a set, and elements of a set have no intrinsic or ordinal values. This is important because if someone says, “tell me the content of row 3”, it might be something one day, and then something completely different another day, because the content can move around a lot. Rather than asking where it is, someone should ask, “tell me the content of the row where xxx is”.

c. The all rows must be unique rule means that all the field names must be different and unique. This is important because it will make it hard to search for a specific piece of data in a field if there are more than one field with the same name.