The Psql program

Outline

1. The Psql command-line format
2. The psql meta-commands
3. The psqlrc.conf file
4. Importing data with Psql

1. The Psql command-line format

Open SQL shell

- C1:
  - Start/All Programs/PostgreSQLx.x/SQL shell

- C2:
  - cmd
  - cd « C:\Program Files\PostgreSQL\x.x\bin »
  - psql –h localhost postgres postgres

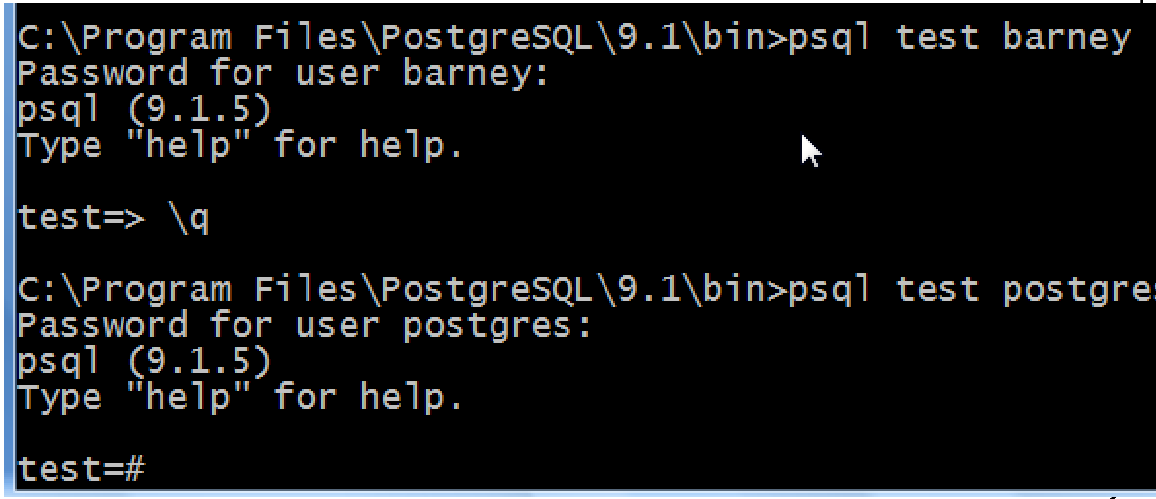
Connection Options

- **Format:** psql [*options*] [*databasename* [*username*]]
  - *options* can be one or more options that define additional controling information
  - The *databasename* and *username* parameters directly specify these values on the command line WITHOUT having *options* format
  - Example:

    psql postgres postgres

    psql test barney

- Connect as
  - superuser: #
  - normal users: >



```
C:\Program Files\PostgreSQL\9.1\bin>psql test barney
Password for user barney:
psql (9.1.5)
Type "help" for help.

test=> \q

C:\Program Files\PostgreSQL\9.1\bin>psql test postgre
Password for user postgres:
psql (9.1.5)
Type "help" for help.

test=#
```

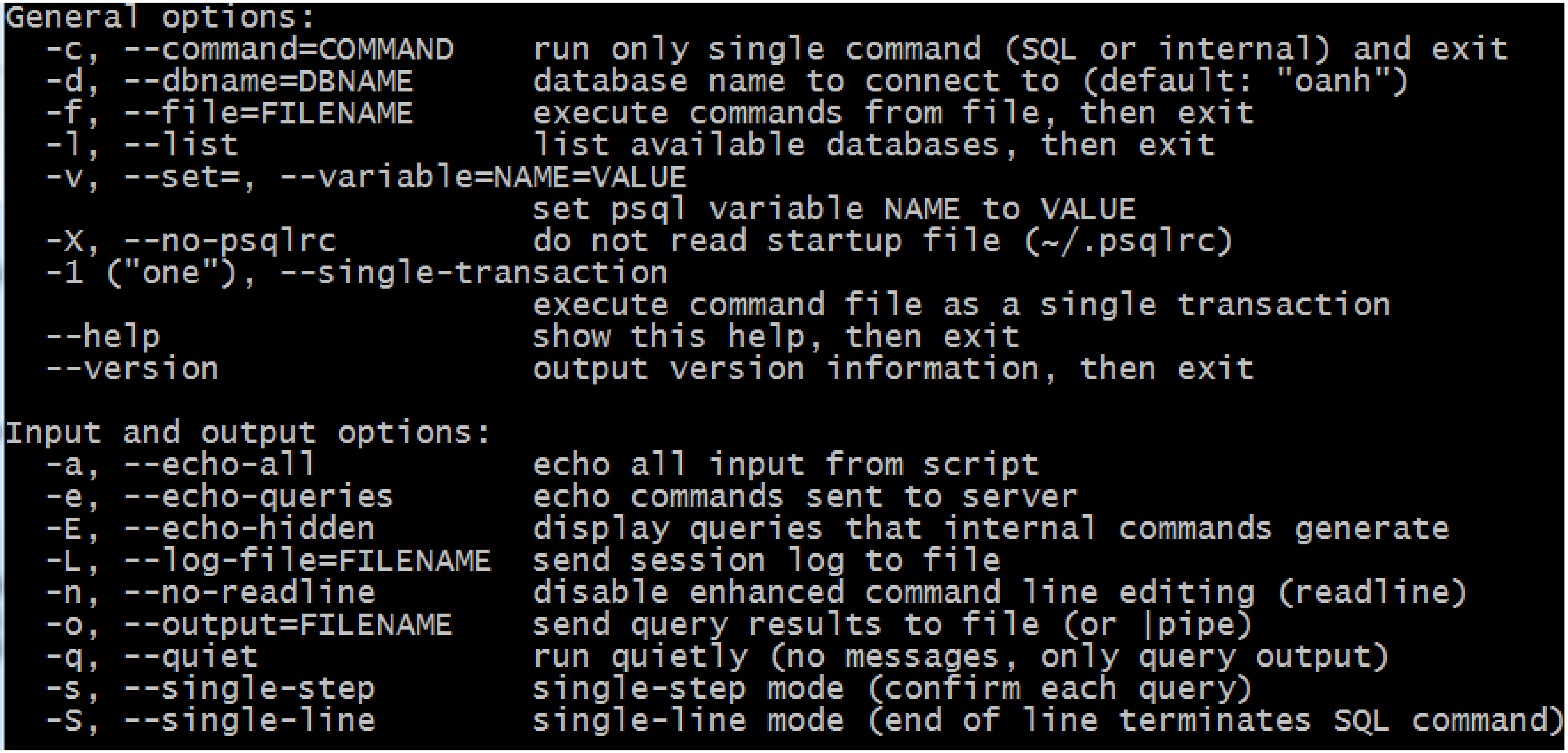
Feature Options

- file:///C:/Program%20Files/PostgreSQL/9.4/doc/postgresql/html/app-psql.html
- Format: options
- *optionname* parameter
  - **Long-name format** Uses a common name to represent the option, preceded by a double dash, such as --port

    psql --host=localhost --port=5432
  - **Short-name format** Uses a single character to represent the option, preceded by just a single dash, such as -e.

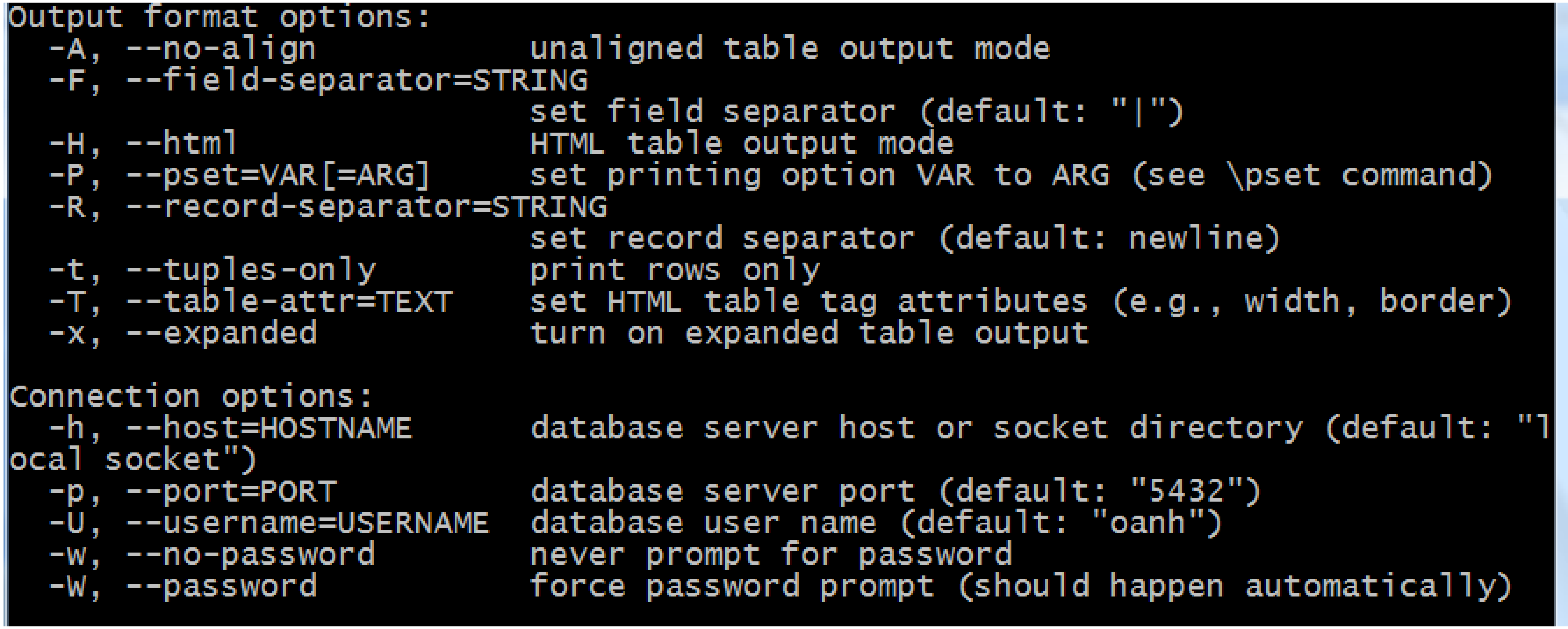
    psql -h localhost -p 5432

Be careful when using the optionnames, as they are case sensitive.



```
General options:
  -c, --command=COMMAND    run only single command (SQL or internal) and exit
  -d, --dbname=DBNAME      database name to connect to (default: "oanh")
  -f, --file=FILENAME      execute commands from file, then exit
  -l, --list               list available databases, then exit
  -v, --set=, --variable=NAME=VALUE
                           set psql variable NAME to VALUE
  -X, --no-psqlrc          do not read startup file (~/.psqlrc)
  -1 ("one"), --single-transaction
                           execute command file as a single transaction
  --help                   show this help, then exit
  --version                output version information, then exit

Input and output options:
  -a, --echo-all           echo all input from script
  -e, --echo-queries       echo commands sent to server
  -E, --echo-hidden        display queries that internal commands generate
  -L, --log-file=FILENAME  send session log to file
  -n, --no-readline        disable enhanced command line editing (readline)
  -o, --output=FILENAME    send query results to file (or |pipe)
  -q, --quiet              run quietly (no messages, only query output)
  -s, --single-step        single-step mode (confirm each query)
  -S, --single-line        single-line mode (end of line terminates SQL command)
```



```
Output format options:
  -A, --no-align           unaligned table output mode
  -F, --field-separator=STRING
                           set field separator (default: "|")
  -H, --html               HTML table output mode
  -P, --pset=VAR[=ARG]     set printing option VAR to ARG (see \pset command)
  -R, --record-separator=STRING
                           set record separator (default: newline)
  -t, --tuples-only        print rows only
  -T, --table-attr=TEXT    set HTML table tag attributes (e.g., width, border)
  -x, --expanded           turn on expanded table output

Connection options:
  -h, --host=HOSTNAME      database server host or socket directory (default: "l
ocal socket")
  -p, --port=PORT          database server port (default: "5432")
  -U, --username=USERNAME  database user name (default: "oanh")
  -w, --no-password        never prompt for password
  -W, --password           force password prompt (should happen automatically)
```

Examples

- Help

```
psql -?
psql --help
```

- Database and username

```
psql  test  postgres
psql -d test --username=postgres
psql --database=test --username=postgres
```

Examples

- Run commands from file ***filename***

```
psql -f filename  test  postgres
psql --file= filename  test  postgres
```

- Put all query output into file *filename*

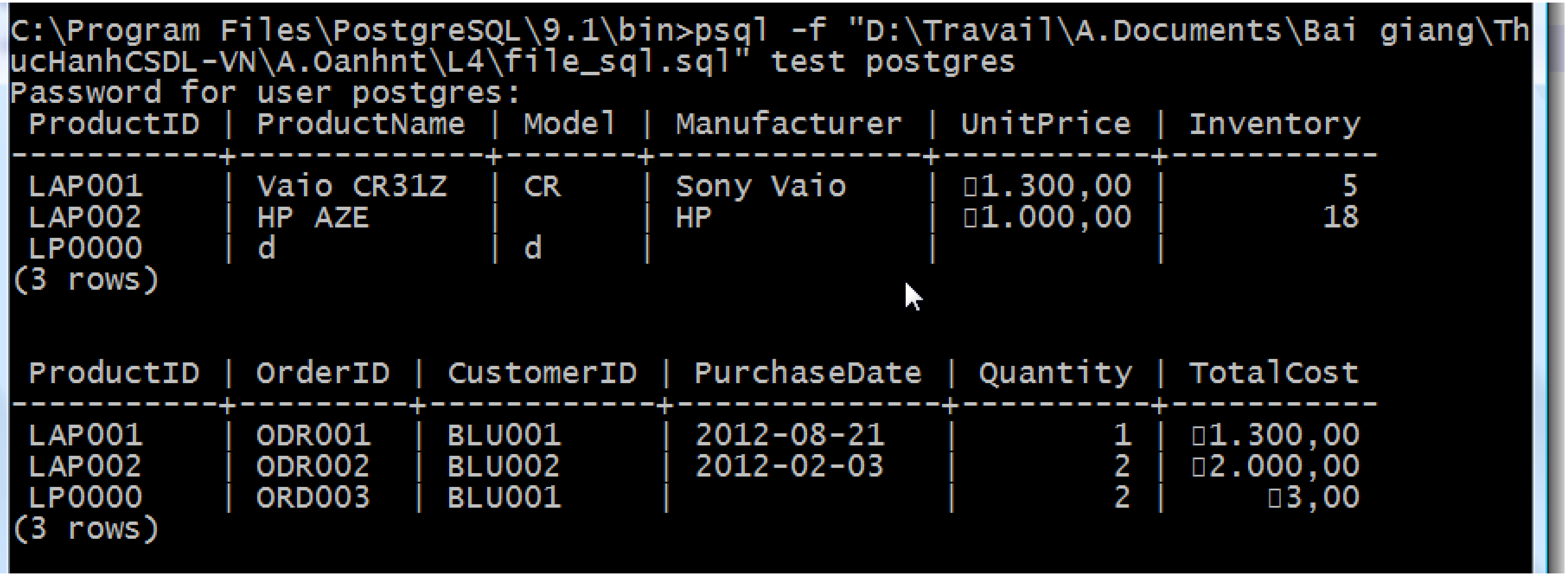
```
psql -o filename  test  postgres
psql --output= filename test  postgres
```

```
psql -f filename -o filename_out test  postgres
```

- E.g.: The content of file *file_sql.sql* :

  select * from store."Product";

  select * from store."Order";

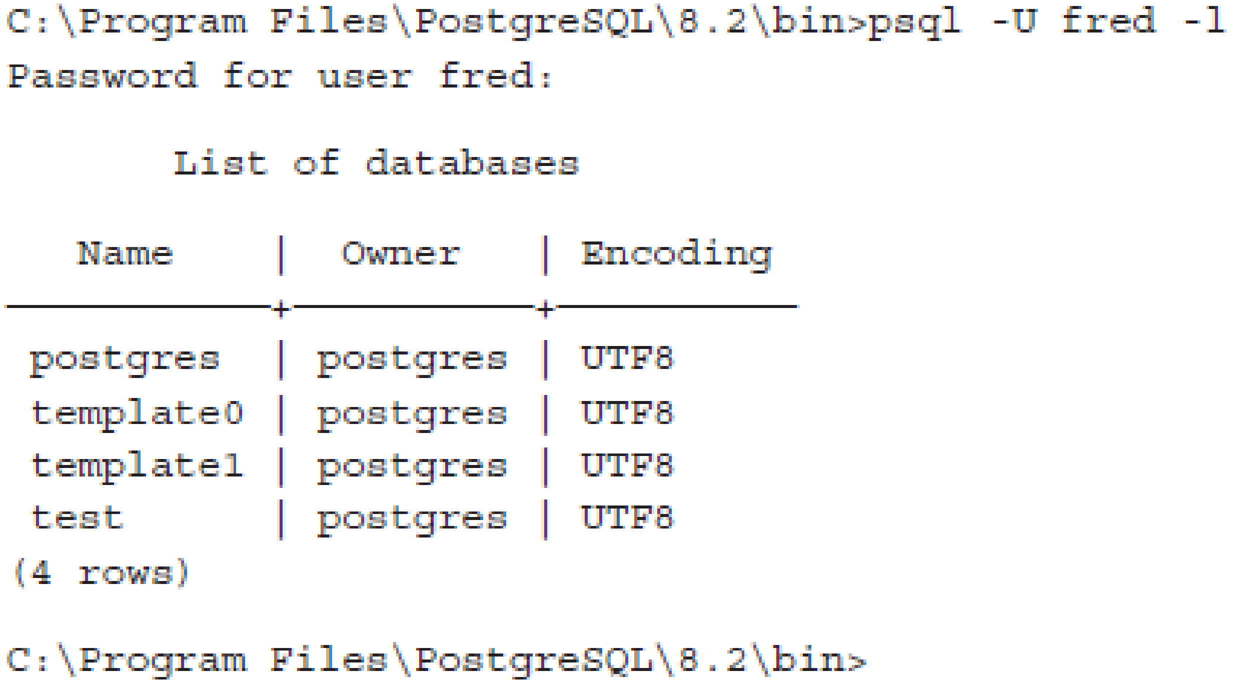


```
C:\Program Files\PostgreSQL\9.1\bin>psql -f "D:\Travail\A.Documents\Bai giang\Th
ucHanhCSDL-VN\A.Oanhnt\L4\file_sql.sql" test postgres
Password for user postgres:
 ProductID | ProductName | Model | Manufacturer | UnitPrice | Inventory
-----------+-------------+-------+--------------+-----------+-----------
 LAP001    | Vaio CR31Z  | CR    | Sony Vaio    | □1.300,00 |         5
 LAP002    | HP AZE      |       | HP           | □1.000,00 |        18
 LP0000    | d           | d     |              |           |
(3 rows)


 ProductID | OrderID | CustomerID | PurchaseDate | Quantity | TotalCost
-----------+---------+------------+--------------+----------+-----------
 LAP001    | ODR001  | BLU001     | 2012-08-21   |        1 | □1.300,00
 LAP002    | ODR002  | BLU002     | 2012-02-03   |        2 | □2.000,00
 LP0000    | ORD003  | BLU001     |              |        2 |     □3,00
(3 rows)
```

Using the Command-Line Options

- Can use more than one option within the command line, but any values associated with the command-line option must be included after the specific command-line option



```
C:\Program Files\PostgreSQL\8.2\bin>psql -U fred -l
Password for user fred:

       List of databases

   Name      |   Owner    | Encoding
-------------+------------+-----------
 postgres    | postgres   | UTF8
 template0   | postgres   | UTF8
 template1   | postgres   | UTF8
 test        | postgres   | UTF8
(4 rows)

C:\Program Files\PostgreSQL\8.2\bin>
```

2. The psql meta-commands

The psql *meta-commands*

- are predefined shortcuts in psql that save you from typing more complex SQL commands
- Each meta-command is preceded by a backslash
- Divided into
  - General meta-commands
  - Query buffer meta-commands
  - Input/output meta-commands
  - Informational meta-commands
  - Formatting meta-commands
  - Copy and large object meta-commands
- Use **\?** to see

The psql *meta-commands (…)*

- General meta-commands
  - \copyright
  - \h *[name] : ex. \h* or *h SELECT*
  - \q : quit
  - \cd *directory*
  - *! cd*
- Input/output commands
  - \i *filename:* execute commands from file *filename*
  - \o *filename:* send all query results to file
  - \copy …

The psql *meta-commands (…)*

- Informational meta-commands
  - \conninfo : connection information
  - \c : current database
  - \c mydb: change current database
  - \list or \l: list all databases

  - \dn: list of schemas
  - \du : list of roles
  - \df : lists functions
  - \dy : lists event trigger
  - ..

The psql *meta-commands (…)*

- Informational meta-commands
  - \d: show tables, views in the current contexte (database/schema)

    \d *.*

    \d *. «Customer »
  - \dp: list tables, views and sequences with their associated access privileges
  - \dt: show tables in the current contexte (database/schema)

    \dt table: table description. Ex. \dt «Customer»

    \dt *.*
  - ..

  (SHOW search_path; SET search_path to store;)

Other examples

- select 1+3;
- select current_user;
- select current_date;
- select current_timestamp;
- select current_database();
- select current_schema();
- select version();
- select * from pg_user;
- select * from pg_tables;

3. The psqlrc.conf file

The psqlrc.conf file

- The psqlrc startup file allows you to **place commonly used metacommands** and **SQL statements in a file** that is processed every time you start psql
- Is not created this file automatically
- >echo %APPDATA%
- Locate: %APPDATA%\postgresql\psqlrc.conf
- Example: create psqlrc.conf with
  - \set cust "Customer"'
  - \set prod "Product”

  C:\Program Files\PostgreSQL\9.4\bin>psql -q test postgres
  - test=> select * from :cust;
  - test=> select * from store.:cust;

4. Importing data with Psql

Importing data with Psql

- The format of the \copy commands
  - \copy *tablename* from|to *filename* [delimiter '*delim*'] […]
- Convert data: Excel →text or CSV files
  - BLU002,Blum,Barbara,879 Oak,Gary,IN,46100,555-4321
  - BLU003,Blum,Katie,342 Pine,Hammond,IN,46200,555-9242
  - BLU004,Blum,Jessica,229 State,Whiting,IN,46300,555-0921

  - test=> \copy store."Customer" from data.txt delimiter ‘,'
  - test=> select * from :cust;

http://wiki.postgresql.org/wiki/COPY