psql User Interface Feb 11, 2013

psql is the replacement for dbacess/sqlcmd

Usage:

```
psql   hd_ob6rhax                          -- opens psql on hd_ob6rhax db
                                           -- db must have been previously created

psql  -f   coop.sql    hd_ob6rhax          -- executes sql commands in file = coop.sql

psql  -l                                   -- lists all databases on postgresql server

psql   hd_ob6rhax  -U username             -- opens psql on database as username
                                           -- similar to "su  username"
```

Commands:

```
\c   dname                 -- changes connection to a new database
\d   tablename             -- lists columns, attributes, indexes for table = tablename
\dt                        -- lists tables and owners in current database
\dv                        -- list views in current database
\df+  function name        -- displays info including SQL code for a function
\i    filename             -- reads and executes queries from file (like psql  -f)
\x                         -- changes output format to vertical mode
\timing                    -- toggles timing info
\?                         -- help for "\" commands
\g                         -- repeat last command
\t                         -- suppress print of headers (recommended when
                                displaying text fields)
\q                         -- quit
```

END QUERIES WITH ; OR \g

To see previous queries use the “up-arrow” key

to show version number of postgresql:

```
select version();
```

to show the IP address of the server that accepted the connection:

```
select inet_server_addr();
```

to show all values of the runtime parameters:

```
show all;
```

to show the value of the "port" parameter:

```
        show port;
```

to signify a character string, use a single quote (‘) - dbaccess uses double quote (“)
to display a NULL value as ‘\N’ instead of a blank

```
        \pset  null  ‘\\N’
```

```
psql  hd_ob5rhax  -q  -t                -- does not print welcome messages, trailers or
                                                footers
```

```
psql  -A  -t  -d hd_ob90fwr  -c  “select * from Wfo”
```

- prints one record per line with no messages, trailers, headers or blank lines
- can redirect to a file

To print the number of records in the RadarLoc table:

```
nrad = $(echo "select count(*) from radarloc;" | psql hd_ob5rhax   -q  -t) ; echo $nrad
```

To generate a column label RADAR_ ID on output:

```
echo "select radid as \"RADAR_ID\" from radarloc;"  |  psql  dbname
```

(without the \", RADAR_ID would show up as radar_id)

```
psql   dbname

set timezone='UTC';

select current_timestamp;       -- prints current time in UTC
```

Example of using user’s environment variable within an SQL statement in psql

```
export  FXA_LOCAL_SITE=fwr

psql  hd_ob82fwr

\set  local_site  `echo “ ‘$FXA_LOCAL_SITE’ “ `      -- sets local_site variable to ‘fwr’
                                                     -- i.e.a string surrounded by single
```

```
                                        -- quotes

UPDATE  RadarLoc   SET   office_id  =  :local_site ;
```

Question:
In postgres 7.4.7, how do I change the field separator when outputing from psql?

The docs say that I can use something like "-F '\t' " but when I try this, the field sep remains '|'. My psql command is as follows

```
psql  -q  -t  hd_ob7ounx  -f file.sql   >  outfile
```

This results in all of the fields I want but with a field separator of '|'. Trying

```
psql  -q  -t  -F '' hd_ob7ounx ...
```

or

```
psql  -q  -t  -f '\t' hd_ob7ounx ...
```

still outputs results with field separator of '|'.

What am I missing?

Answer:
The documentation is lacking here....

You need to use unaligned mode to get it to use the new separator. To do this, add -A to the command line.

Question:
Can I get psql to output the fields separated by a <tab>? Using "psql ... -A -F '\t' ... gives me the literal \t as the separator.

Answer:
This is basically a bash problem

Use

```
'Ctrl v tab'
```

You hit ctrl-v then tab. It will expand. You also need to single quote it.

Question:

When I output floating point numbers from psql, sometimes I get the output with two decimal places (42.33) and sometimes I get output with many decimal places (42.3333333333333). In Informix, I could control this with the DBFLTMSK environment variable. How can I control the number of decimal places in PostgreSQL?

Answer:

Use the “to_char” function to modify the output. For example:

```
SELECT lid, to_char(lat,’99.99’), to_char(lon,’99.99’)
FROM Location;
```

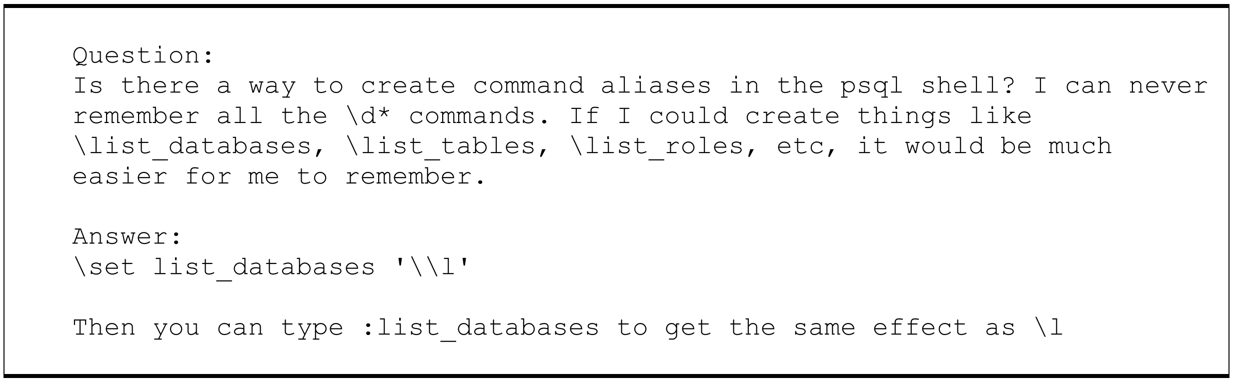
will output all latitude/longitude values to two decimal points. See Section 9.7 of the PostgreSQL documentation for a description of the “to_char” function.

Question:

Is there a way to load multiple .sql files in a single transaction?

Answer:

```
cat begin.sql file1.sql file2.sql ... commit.sql | psql …
```



Question:

Is there a way to create command aliases in the psql shell? I can never remember all the \d* commands. If I could create things like \list_databases, \list_tables, \list_roles, etc, it would be much easier for me to remember.

Answer:

```
\set list_databases '\\l'
```

Then you can type :list_databases to get the same effect as \l