Exercises for The Accidental DBA Tutorial

Version 0.7

Exercises are meant to be performed in order. Some may not work out of order. See accompanying presentation in order to understand them.

Constants

EDITOR is your choice of command-line text editor:

- vi
- vim
- joe (a simple text editor)
- nano (best for Linux newbies)
- jmacs (lightweight version of emacs)

DOW is the current day of the week, as an abbreviation, used in log file names, such as "Mon", "Tue", or "Fri".

vagrant up and login

```
cd /dir/to/accidentalDBA/vagrant/
vagrant up
vagrant ssh
sudo su -
tmux
```

docker run and login

```
docker run -it -e TERM jberkus/accidentaldba:latest
```

packages

```
less /etc/apt/sources.list.d/pgdg.list
apt-cache search postgresql
```

initdb

```
su – postgres
mkdir test
initdb -D test
cd test
ls -l
exit
```

start/stop

```
service postgresql start
service postgresql restart
service postgresql reload
service postgresql stop
su - postgres
pg_ctl -D /etc/postgresql/9.4/main start
```

```
pg_ctl -D /etc/postgresql/9.4/main -m fast stop
exit
cd /etc/postgresql/9.4/main/
EDITOR postgresql.conf
# add the line "not_a_parameter = true" at the top
service postgresql start
less /var/log/postgresql/postgresql-DOW.log
EDITOR postgresql.conf
# delete the line "not_a_parameter = true"
service postgresql start
less /var/log/postgresql/postgresql-DOW.log
```

psql

```
su - postgres
psql libdata
\?
\h create table
\dt
\d+ copies
```

extensions

```
\dx
create extension hstore;
select * from pg_extension;
select 'fname => josh,
  lname => berkus'::hstore;
```

tmux new window

```
<ctrl>b,c
<ctrl>b,n
<ctrl>b,n
```

xlog directory

```
<ctrl>b,n
cd 9.4/main/pg_xlog
ls -lh
```

postgresql.conf

```
cd /etc/postgresql/9.4/main
EDITOR postgresql.conf
#no editing; just going over the file and what it contains
psql
show max_connections;
show all;
\x
select * from pg_settings;
```

create user

```
\x
create user bench password 'benchmark';
\du
\q
```

pg_hba.conf

```
<ctrl>b,n
$EDITOR /etc/postgresql/9.4/main/pg_hba.conf
#comment out the two "trust" lines which are the first uncommented lines
service postgresql reload
<ctrl>b,n
psql
\q
psql -U bench postgres
\q
<ctrl>b,n
cp /setup/postgres/.pgpass ~postgres/
chown postgres:postgres ~postgres/.pgpass
<ctrl>b,n
cd ~
chmod 700 .pgpass
less .pgpass
psql -U bench postgres
\q
```

pgbench

```
createdb bench
pgbench -U bench -i -s 10
<ctrl>b,n
```

pgbouncer

```
less /etc/pgbouncer/pgbouncer.ini
less /etc/pgbouncer/userlist.txt
service pgbouncer start
<ctrl>b,n
psql -U bench -p 6432
\q
cd ~/pgbench
less runbench_pool.sh
./runbench_pool.sh
<ctrl>b,n
su - postgres
psql
select count(*) from pg_stat_activity;
\q
<ctrl>b,n
<ctrl>c
```

pgdump

```
cd ~
pg_dump -Fc -v -f backup/libdata.dump libdata
ls -lh backup/libdata.dump
pg_restore -l backup/libdata.dump
createdb libdata2
pg_restore -v -d libdata2 backup/libdata.dump
psql libdata2
\dt
\q
```

PITR

```
<ctrl>b,n
exit
$EDITOR /etc/postgresql/9.4/main/postgresql.conf
#uncomment the parameters in the replication section
$EDITOR /etc/postgresql/9.4/main/pg_hba.conf
#uncomment the replication connection lines at the bottom
service postgresql restart
less /setup/postgres/archive_logs.sh
su - postgres
cd ~/pgbench
sh runbench_log.sh
<ctrl>b,n
date
# mark what time it is
psql libdata
drop table copy_history;
\q
cd ~
pg_basebackup -c fast -P -D 9.4/replica
cd wal_archive
ls -lh
cd ~/9.4/replica
cp -p -r ~/archive/* .
$EDITOR recovery.conf
# set recovery_target_time to a few minutes ago
pg_ctl -D . start
tail -f /var/log/postgresql/postgresql-replica
<ctrl>c
psql -p 5433 libdata
select pg_is_in_recovery();
\dt
select * from copy_history limit 10;
select pg_xlog_replay_resume();
select pg_is_in_recovery();
\dt
\q
```

replication

```
pg_ctl -D . -m fast stop
rm -rf *
pg_basebackup -X stream -c fast -P -D .
cp -p -r ~/archive/* .
cp recovery.conf.replica recovery.conf
$EDITOR recovery.conf
# again, just looking around
pg_ctl -D . start
psql -p 5433
select pg_is_in_recovery();
\q
psql -p 5432
select * from pg_stat_replication;
\q
pg_ctl -D . promote
psql -p 5433
select * from pg_is_in_recovery();
\q
pg_ctl -D . stop
```

database activity

```
<ctrl>b,n
$EDITOR /etc/postgresql/9.4/main/postgresql.conf
#comment out the logging parameters, and uncomment the logging
#parameters in the "query logging example" subsection
service postgresql reload
<ctrl>b,n
psql bench
create extension pg_stat_statements;
\q
cd ~/pgbench
runbench_log.sh
<ctrl>b,n
su - postgres
psql
select * from pg_stat_activity;
\x
select * from pg_stat_activity;
\c bench
select * from pg_stat_user_tables;
select pg_size_pretty(pg_total_relation_size('pgbench_history'));
select * from pg_stat_statements;
\x
select query, calls, total_time from pg_stat_statements order by calls desc limit
10;
select query, calls, total_time/calls as avg_time from pg_stat_statements order
by 3 desc limit 10;
select query, calls, shared_blks_read/calls as avg_reads from pg_stat_statements
order by 3 desc limit 10;
select * from pg_stat{TAB}
```

database activity log

```
exit
cd /var/log/postgresql
less activitylog-DOW.csv
pgbadger --format csv -o ~/badger.html activitylog-DOW.csv
```

vacuum

```
<ctrl>b,n
psql bench
\x
select * from pg_stat_user_tables;
select pg_total_relation_size('pgbench_tellers');
vacuum analyze pgbench_tellers;
select pg_total_relation_size('pgbench_tellers');
vacuum full pgbench_tellers;
select pg_total_relation_size('pgbench_tellers');
```

vacuum freeze

```
select relname, age(relfrozenxid) as xid_age
from pg_class JOIN pg_stat_user_tables USING (relname) order by xid_age desc;
VACUUM FREEZE;
select relname, age(relfrozenxid) as xid_age
from pg_class JOIN pg_stat_user_tables USING (relname) order by xid_age desc;
```

analyze and stats

```
\a
select * from pg_stats where schemaname = 'public';
\q
```

zombie killing

```
<ctrl>b,n
./runbench_locks.sh
<ctrl>b,n
psql bench
\x
select * from pg_stat_activity;
\x
select now()-xact_start as xact_age,
    now()-state_change as idle_time,
    pid, query
from pg_stat_activity
where state = 'idle in transaction';

select pg_terminate_backend(pid)
from pg_stat_activity
where state = 'idle in transaction'
and state_change < ( now() - interval '10 seconds');

select now()-xact_start as xact_age,
    now()-state_change as idle_time,
```

```
    pid, query
from pg_stat_activity
where state = 'idle in transaction';
```

explain analyze

```
\c libdata
explain select count(*) from loans where checkout_date between '2011-01-01' and
'2011-03-31';
explain analyze select count(*) from loans where checkout_date between '2011-01-
01' and '2011-03-31';
explain ( analyze on, buffers on ) select count(*) from loans where checkout_date
between '2011-01-01' and '2011-03-31';
explain ( analyze on, format yaml ) select count(*) from loans where
checkout_date between '2011-01-01' and '2011-03-31';

\i ~/archive/explain_quarterly_report.sql
```

end vagrant

```
<ctrl>b,d
exit
exit
vagrant destroy
```

exit docker

```
<ctrl>b,d
exit
```