Table of Contents

PGSQL-RUNNER

1. WHAT IS THIS ?!

A bash script tool to run your postgres sql script files in batches, exit on any sql script error and log it's actions nicely …
More docs with more info could be found in the doc/md or doc/pdf dirs …
The tool has been developed and tested on Ubuntu, any Linux distro with gnu bash should work out of the box, no joy for MACs, neither Windows... Athough I am ready to buy a beer to the one who ports its first ... as there are only 2 issues to resolve ...

2. INSTALLATION AND CONFIGURATION

This installation instructions are for Ubuntu. Feel free to apply the package manager commands of your OS.

2.1. Prerequisites

The must have binaries are:
bash, perl, zip,postgresql
Check the DevOps guide on how-to install the postgres RDBMS.
The nice to have are:
tmux, vim ,ctags
The examples are for Ubuntu - use you OS package manager …

```
apt-get autoclean
apt-get install --only-upgrade bash

sudo apt-get install -y perl

# optionally
sudo apt-get install -y excuberant-ctags
sudo apt-get install -y 7z

apt-get upgrade
```

2.2. Fetch the source

Fetch the source from git hub as follows:

```
# generate your ssh keys to authenticate yourself against github
ssh-keygen -t ecdsa -b 521
# paste the output as new key into the https://github.com/settings/keys section
cat ~/.ssh/id_ecdsa.pub
# of course you could skip those 2 lines above if you have configured keys



# create your product dir:
mkdir -p /opt/csitea/pgsql-runner
cd /opt/csitea/pgsql-runner/

# fetch the source
```

```
git clone git@github.com:YordanGeorgiev/pgsql-runner.git
```

2.3. Build the first pgsql-runner instance

When doing for first time do exactly as shown bellow, otherwise no joy ...
Each pgsql-runner instance has it's own version, environmnt type and owner. For now just follow the instruction - after half an hour you will be hacking this …

```
# build your product version dir - a kind of "this instance of the thingy dir"
mv -v /opt/csitea/pgsql-runner/pgsql-runner /opt/csitea/pgsql-runner/pgsql-runner.0.1.3.dev.$USER

# the dir name basically means :
# the 0.1.3 ver dev env deployment instance for the $USER ...
# thus you could have multiple versions with multiple enviornments
# for different usrs on the same host ...
```

2.4. Create you local conf file

The default conf file provides only limited functionality (this is by design) , thus copy and configure the configuration file for your host

```
# go to the product version dir
cd /opt/csitea/pgsql-runner/pgsql-runner.0.1.3.dev.$USER

mv -v sfw/bash/pgsql-runner/pgsql-runner.set-your-host.conf \
sfw/bash/pgsql-runner/pgsql-runner.`hostname -s`.conf
```

2.5. Start hacking

Start hacking … or wait check at least the test call running all the functions of the tool …

```
# opionally if you are in the vim camp open the "project relative files list file"
vim meta/.dev.pgsql-runner
# each uncommented "action" will be run
vim src/bash/pgsql-runner/tests/run-pgsql-runner-tests.lst

# Ctrl + Z ,
bash sfw/bash/pgsql-runner/test-pgsql-runner.sh

# now clone your own instance
bash sfw/bash/pgsql-runner/pgsql-runner.sh -a to-app=my-tool
```