nix- (nix original command)

$nix-env –list-generations $nix-env -G 42 or $nix-env –switch-generation 42
$nix-env –rollback
$nix-env -e firefox $nix-env -e '.*'
$nix-store –delete $nix-store –gc
$nix-collect-garbage -d

nix (nix command improved)

$nix-env   nix profile

$nix-env -iA firefox $nix profile install firefox $nix profile install nixpkgs#hello
$nix-env -e firefox $nix profile remove firefox
$nix-env -q $nix profile list
$nix-env –list-generations $nix profile history
$nix-env –rollback $nix profile rollback
$nix-env -qa
$nix-env -qaPs

$nix profile install $nix profile remove $nix profile rollback
$nix profile list $nix profile history

nix-env App profile manifest.nix , nix profile   App manifest.json . nix profile App profile
nix-env   .

$nix-store -q –tree 'realpath ./result/bin/S4
$nix-store -q –referrers-closure 'realpath ./result/bin/S4

nixos

$nixos-rebuild switch –rollbackflake.nix >flake.nix ▉

$sudo nixos-rebuild switch –flake .#syryuhds –impure

flake

[flake.nix]$nix build   nix build github:syryuauros/myflakes/masterflake.nix >flake.nix ▉

$nix run   nix run github:syryuauros/myflakes/master
$ nix run nixpkgs#gnuplot

$realpath ./result/bin/S4 $nix-store -q –tree 'realpath ./result/bin/S4'

$nix flake metadata github:edolstra/dwarffs

1

$nix flake show github:edolstra/dwarffs

$nix-store -q –referrers-closure 'realpath ./result/bin/S4

$ nix registry add nixpkgs github:NixOS/nixpkgs/5272327b81ed355bbed5659b8d303cf2979b6953

$nix-info -m

"`https://github.com/`<owner>/<project>/commit/<hash>"
example
`https://github.com/nixos/nixpkgs/commit/ede02b4ccb13557b95058d66146640a2b0bb198f`

$nix build (default package) $nix build .#s4 .#gnuplot (default package   package .)

flake override

ex) (openblas.override { singleThreaded = true;})

or let openblas-single-thread = openblas.override { singleThreaded = true; };

nix-shell

nix-shell -p 'lua.withPackages(ps: with ps; [busted luafilesystem])'

nixpkgs

confirm packages for nix-build, nix-instantiate, nix-channel

1 $nixrepl [Ret] nix-repl>:l <nixpkgs> [Ret] nix-repl>keyword(ex:rust) + TAB
2 /nix/var/nix/profiles/per-user/root/channels/nixos/pkgs/top-level/all-packages.nix
3 $cd 'nix-instantiate –eval -E '<nixpkgs>'' [Ret] cat > all-packages.nix
4 $NIX_PATH,   all-packages.nix

confirm packages for nix-env

1 $nix-env -qa 'keyword.*' or $nix-env -qaPs 'keyword'
2 $cd ~/.nix-defexpr/channels_root/nixos/pkgs/top-level, all-packages.nix

confirm packages for flakes

1 flake.lock   » "nixpkgs" "rev" term » github.com/[owner]/[repo]/tree/[rev]

nix repl

nix-repl>

:help
 TAB
:doc keyword

pictures

./img/3_nix/flake input url.png

./img/3_nix/real repo flake indicate.png

./img/3_nix/real repo flake indicate_2.png

./img/3_nix/real repo flake indicate_3.png

./img/3_nix/repo find from flake.lock.png

./img/3_nix/repo find from flake.lock2.png

./img/3_nix/repo find from flake.lock3.png

./img/3_nix/how to find packages_1.png

./img/3_nix/how to find packages_2.png

.