Nix Reproducible packages

CONCEPTS

The nix command and flakes are experimental. Enable them once: experimental-features = nix-command flakes in ~/.config/nix/nix.conf.

flake.nix inputs and outputs
↓ lock
flake.lock pinned revisions
↓ evaluate
Derivation .drv build recipe
↓ build
/nix/store hash-addressed path

An installable is flakeref#attr, e.g. nixpkgs#hello or .#myapp. A bare . means the flake in the current directory. Flakes only see files tracked by git: git add new files.

RUN AND TRY

nix run *<installable>* **[--** *<args>***]**
Run an app from a flake

nix shell *<installable>*...
Shell with packages on PATH

nix shell *<installable>* **-c** *<cmd>*
Run one command with the package

nix run nixpkgs#*<pkg>*
Run a nixpkgs package

nix run .
Run this flake's default app

nix run github:*<owner>***/***<repo>*
Run a flake from GitHub

BUILD

nix build
Build default package of ./flake

nix build *<installable>*
Build a given installable

nix build .#*<name>*
Build a named package of this flake

BUILD (cont.)

nix build -L
Show full build logs

DEVELOP

nix develop
Shell with the dev environment

nix develop -c *<cmd>*
Run a command in that environment

FLAKES

nix flake init
Flake from the default template

nix flake show [*<flakeref>***]**
List a flake's outputs

nix flake check
Evaluate and run flake checks

nix flake update
Update all inputs in the lock

nix flake update *<input>*
Update a single input

PROFILE

nix profile add *<installable>*
Install into your profile

nix profile list
List installed packages

nix profile remove *<name>*
Uninstall a package

nix profile upgrade --all
Upgrade installed packages

SEARCH AND INSPECT

nix search nixpkgs *<regex>*
Search packages by name or description

STORE AND CLEANUP

nix store gc
Delete unreachable store paths