# just Command runner

## CONCEPTS

just is a command runner, not a build system: recipes always run, no file timestamps are checked. Indent recipe bodies consistently with spaces or tabs.

**justfile** found upwards from cwd
↓ `just <recipe>`
**Dependencies** run first, once each
↓ then
**Recipe lines** each in its own shell
↓ `cwd = justfile_dir`
**Result** stops at first failure

Every recipe line runs in a fresh shell, so cd and shell variables do not carry to the next line. Use a shebang recipe or && for multi-step state.

## RUNNING

**`just`**
Run the default (first) recipe

**`just`** *`<recipe>`* **`[`***`<args>`***`...]`**
Run one or more recipes in order

**`just --list / just -l`**
List recipes with doc comments

**`just --dry-run / just -n`**
Print recipe lines without running them

**`just --show`** *`<recipe>`* **`/ just -s`** *`<recipe>`*
Print a recipe's source

**`just --init`**
Write a starter justfile in the project root

**`just`** *`<var>=<value> <recipe>`*
Override a variable for this run

**`just --justfile`** *`<file>`* **`/ just -f`** *`<file>`*
Use a specific justfile

## RECIPE SYNTAX

**A recipe**

```
# doc comment shown by --list
build:
    cargo build --release
```

*`<name>`***`:`** *`<dep> <dep>`*
Run dependencies before the recipe

## RECIPE SYNTAX (cont.)

*`<name>`***`: (`***`<dep>`** **`"`***`<arg>`***`")`**
Pass an argument to a dependency

**`@`***`<name>`***`:`**
Don't echo this recipe's lines

**`@`***`<command>`*
Don't echo one line

**`-`***`<command>`*
Keep going if this line fails

**`{{`***`<var>`***`}}`**
Insert a variable or expression into a line

*`<name> <param>`***`:`**
Required parameter

*`<name> <param>`***`=`***`<default>`***`:`**
Parameter with a default

*`<name>`* **`+`***`<files>`***`:`**
Variadic: one or more values

*`<name>`* **`*`***`<flags>`***`:`**
Variadic: zero or more values

**List recipes when run bare**

```
default:
    @just --list
```

## VARIABLES

*`<name>`* **`:= "`***`<value>`***`"`**
Define a variable

**`export`** *`<NAME>`* **`:= "`***`<value>`***`"`**
Export to recipe environments

**`set dotenv-load`**
Load .env into the environment

## SETTINGS

**`set shell := ["bash", "-euo", "pipefail", "-c"]`**
Choose the shell for lines and backticks

## ATTRIBUTES

**`[private]`**
Hide from --list (also names starting with _)

## SCRIPTS & MODULES

**Shebang recipe**

```
stats:
    #!/usr/bin/env python3
    import sys
    print(sys.version)
```