tmuxinator

tmux session templates

CONCEPTS

A project is a YAML file that describes one tmux session: windows, panes and the commands each runs.

Project file name, root, hooks

- **Window** layout, root
  - **Pane** command

tmuxinator only builds the session; once it runs, you work in plain tmux.

PROJECTS

`tmuxinator new` *`<project>`*
Create a project and open it in $EDITOR

`tmuxinator edit` *`<project>`*
Edit an existing project file

`tmuxinator list`
List configured projects

`tmuxinator open` *`<project>`*
Open a project, creating it if missing

`tmuxinator copy` *`<old>`* *`<new>`*
Duplicate a project

`tmuxinator delete` *`<project>`*
Remove a project file

`tmuxinator new` *`<project>`* *`<session>`*
Build a project from a running tmux session

START AND STOP

`tmuxinator start` *`<project>`*
Create the session and attach

`tmuxinator start`
Start from ./.tmuxinator.yml

`tmuxinator stop` *`<project>`*
Stop a project's session

`tmuxinator stop-all`
Stop every active project

`tmuxinator start` *`<project>`* **`-n`** *`<name>`*
Use a custom session name

`tmuxinator start` *`<project>`* **`-p`** *`<file>`*
Use a specific config file

START AND STOP (cont.)

`tmuxinator start` *`<project>`* **`--append`**
Add windows to the current session

PROJECT FILE: TOP LEVEL

`name:` *`<name>`*
Session name (no dots)

`root: ~/code/app`
Directory every window starts in

`windows:`
List of window definitions

`startup_window:` *`<name>`*
Window selected on start

`startup_pane:` *`<index>`*
Pane selected in that window

`pre_window:` *`<cmd>`*
Run before every pane's commands

`attach: false`
Create the session without attaching

WINDOWS AND PANES

`- editor: vim`
Window with a name and one command

`layout: main-vertical`
Pane layout for the window

`panes:`
List of panes in the window

`- logs: tail -f log/dev.log`
Named pane running a command

`root:` *`<dir>`*
Window-specific root directory

HOOKS AND TEMPLATES

`on_project_first_start:` *`<cmd>`*
Run only when the session is created

SETUP AND DEBUG

`tmuxinator doctor`
Check tmux, editor and shell setup

`tmuxinator debug` *`<project>`*
Print the shell commands it would run

`tmuxinator version`
Show the installed version