

7.5 Script Naming

[Go To Top](#)

[Return to Naming and File Conventions](#)

Script names should describe the action that the script performs. The team should use verb-based names for developer-facing scripts because developers run scripts to do something.

Representative script names include:

- `createRepositoryStructure.sh`
- `checkPrerequisites.sh`
- `initialisePythonEnvironment.sh`
- `generateTypes`
- `buildContainers.sh`
- `runAll.sh`
- `runNode.sh`
- `stopAll.sh`
- `prepareGitRepository.sh`

The script name should match the behaviour described in the handbook and script reference appendix. If the script checks prerequisites, it should not also modify the [Repository](#) unexpectedly. If the script builds containers, it should not silently start the full [Node](#) network unless the handbook says so.

Shell scripts should use the `.sh` suffix where the team expects developers to run them as shell scripts. Python scripts should use the `.py` suffix unless the team wraps them behind a language-neutral command. The team should document executable commands clearly so that developers know whether to run a shell script, a Python module, a build tool command, or a wrapper script.

Script names should remain stable once other scripts, documentation, or acceptance procedures depend on them. If the team renames a script, it should update all references in the handbook, README, script reference appendix, tests, and automation.

© 2026 Dido Solutions, Inc. and Jackrabbit Consulting, Inc.

From:
<https://wiki.didosolutions.com/> - **Dido Solutions, Inc Wiki**

Permanent link:
<https://wiki.didosolutions.com/fxdemo/05-part/07-naming-and-file-conventions/07-5-script-naming/start>

Last update: **2026/08/09 17:22**

