

6.5 Prerequisite Validation

[Go To Top](#)

[Return to Prerequisites and Tooling](#)

The `checkPrerequisites.sh` script should validate the required tools before the developer generates [Artifacts](#), builds containers, runs [Nodes](#), or collects acceptance [Evidence](#).

The script should check:

1. Required commands are available on the path.
2. Required versions meet the [Baseline](#) expectations.
3. Required environment variables exist and point to valid locations.
4. Required [DDS](#) tooling can run.
5. Required container tooling can run.
6. Required Python support exists, if Python utilities form part of the Baseline.
7. Required runtime language tools exist, if Phase 0 Nodes or generated bindings need them.
8. Required [Repository](#) directories and [Configuration](#) templates exist.

The script should produce clear, actionable output. When a check passes, the script should report the result briefly. When a check fails, the script should explain what failed, why it matters, and what the developer should inspect or install next.

The script should return a non-zero exit code when a required prerequisite fails. It may return warnings for missing optional tools, but those warnings should not block Baseline execution unless the developer requested a workflow that depends on the optional tool.

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

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

Permanent link:
<https://wiki.didosolutions.com/fixdemo/05-part/06-prerequisites-and-tooling/06-6-troubleshooting-tool-issues/start?rev=1786320964>

Last update: 2026/08/09 17:16

