

# Scripts

---

## \_ensure-venv.ps1

- NAME

```
./scripts\_ensure-venv.ps1
```

- SYNOPSIS

```
Helper utilities for FaceForge PowerShell scripts.
```

- SYNTAX

```
./scripts\_ensure-venv.ps1 [<CommonParameters>]
```

- DESCRIPTION

This file is intended to be dot-sourced by other scripts in the `scripts/` directory.  
It provides shared helpers for:

- Resolving the repository root.
- Creating and locating the repo-local Python virtual environment (`.venv`).
- Returning the correct `.`venv` Python executable path for downstream scripts.

Design goals:

- Never depend on global Python packages at runtime.
- Prefer the Windows `py` launcher to bootstrap Python 3.12 venv creation.
- Make failures actionable with clear error messages.

- PARAMETERS

```
<CommonParameters>
```

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see about\_CommonParameters (<https://go.microsoft.com/fwlink/?LinkID=113216>).

- INPUTS

- OUTPUTS

- NOTES

```
This file defines functions and does not execute any build/run tasks on its own.  
Usage pattern:  
. (Join-Path $PSScriptRoot '_ensure-venv.ps1')  
$repoRoot = Get-RepoRoot  
$venvPython = Ensure-Venv -RepoRoot $repoRoot
```

- RELATED LINKS

---

## build-core.ps1

- NAME

```
./scripts\build-core.ps1
```

- SYNOPSIS

```
Builds the FaceForge Core executable for local bundling and releases.
```

- SYNTAX

```
./scripts\build-core.ps1 [-KeepBuildHistory] [-AllowTimestampFallback] [<CommonParameters>]
```

- DESCRIPTION

```
Produces a Windows executable for FaceForge Core using PyInstaller, using the repo-local virtual environment under `venv`.
```

This script is designed to be safe and repeatable:

- Uses a repo-local venv (never relies on global site-packages after bootstrapping).
- Cleans previous `core/build` and `core/dist` output directories.
- When requested, falls back to timestamped output folders if directories are locked.
- Normalizes output to a stable path `core/dist/faceforge-core.exe` for downstream scripts.

Intended consumers:

- Local developers running the Desktop orchestrator.
- CI pipelines producing release artifacts.
- The all-in-one desktop bundler script (`scripts/build-desktop.ps1`).

- PARAMETERS

```
-KeepBuildHistory [<SwitchParameter>]  
Preserves older timestamped output folders (e.g. `core/build-YYYYMMDD-HHMMSS`, `core/dist-...`).  
By default, old timestamped output folders are pruned to keep the repo tidy.  
  
Required?          false  
Position?         named  
Default value     False  
Accept pipeline input?  false  
Aliases  
Accept wildcard characters?  false  
  
-AllowTimestampFallback [<SwitchParameter>]
```

If `core/build` or `core/dist` cannot be deleted (e.g. open file handles from Explorer or a terminal with CWD inside those folders), the default behavior is to fail with a helpful message. When this switch is provided, the script will instead build into timestamped folders.

```
Required?          false
Position?         named
Default value     False
Accept pipeline input?  false
Aliases
Accept wildcard characters?  false

<CommonParameters>
This cmdlet supports the common parameters: Verbose, Debug,
ErrorAction, ErrorVariable, WarningAction, WarningVariable,
OutBuffer, PipelineVariable, and OutVariable. For more information, see
about_CommonParameters (https://go.microsoft.com/fwlink/?LinkID=113216).
```

- INPUTS

- OUTPUTS

On success, you will have:  
- `core/dist/faceforge-core.exe`

- NOTES

Prerequisites:  
- Python 3.12.x installed (only needed to bootstrap `.venv` the first time).  
- Build dependencies are installed into `.venv` automatically.  
This script intentionally disables confirmation prompts for non-interactive execution.

----- EXAMPLE 1 -----

```
PS > ./scripts/build-core.ps1
Builds Core into `core/dist/faceforge-core.exe`, cleaning old outputs.
```

----- EXAMPLE 2 -----

```
PS > ./scripts/build-core.ps1 -AllowTimestampFallback
Builds Core even if `core/dist` is locked, by using timestamped folders.
```

- RELATED LINKS

## build-desktop.ps1

- NAME

```
./scripts\build-desktop.ps1
```

- SYNOPSIS

Builds a full FaceForge Desktop bundle (Desktop + Core sidecar).

- SYNTAX

```
./scripts\build-desktop.ps1 [-Bundles <String>] [-SkipCoreBuild] [-KeepBuildHistory] [-AllowTimestampFallback]  
[<CommonParameters>]
```

- DESCRIPTION

This script follows the project's packaging intent:

- Build FaceForge Core as a standalone executable (PyInstaller)
- Stage it into Desktop's Tauri sidecar binaries folder
- Produce Desktop installers via `tauri build`

This script is designed for repeatable local and CI builds:

- Uses the repo-local `venv` for Core builds (never global site-packages).
- Stages the Core sidecar into `desktop/src-tauri/binaries/faceforge-core.exe`.
- Runs `cargo tauri build` to produce installable artifacts.

Prerequisites (Windows):

- Rust toolchain installed (cargo).
- Tauri prerequisites (WebView2, bundler toolchains). Tauri will prompt/download some tooling.

Outputs (Windows):

- desktop/src-tauri/target/release/bundle/msi/\*.msi
- desktop/src-tauri/target/release/bundle/nsis/\*-setup.exe

- PARAMETERS

```
-Bundles <String>  
    Which bundle target(s) to build. Allowed values: all, msi, nsis.  
    Default: all.  
  
    Required?          false  
    Position?         named  
    Default value     all  
    Accept pipeline input?  false  
    Aliases  
    Accept wildcard characters?  false  
  
-SkipCoreBuild [<SwitchParameter>]  
    Skip running scripts/build-core.ps1 (expects core/dist/faceforge-core.exe to already exist).  
  
    Required?          false  
    Position?         named  
    Default value     False  
    Accept pipeline input?  false  
    Aliases  
    Accept wildcard characters?  false  
  
-KeepBuildHistory [<SwitchParameter>]  
    Forwarded to scripts/build-core.ps1; preserves old build/dist folders under core/.  
  
    Required?          false  
    Position?         named  
    Default value     False  
    Accept pipeline input?  false  
    Aliases  
    Accept wildcard characters?  false  
  
-AllowTimestampFallback [<SwitchParameter>]  
    Forwarded to scripts/build-core.ps1; if core/build or core/dist are locked, build into timestamped folders.  
  
    Required?          false  
    Position?         named  
    Default value     False  
    Accept pipeline input?  false  
    Aliases  
    Accept wildcard characters?  false
```

```
<CommonParameters>
    This cmdlet supports the common parameters: Verbose, Debug,
    ErrorAction, ErrorVariable, WarningAction, WarningVariable,
    OutBuffer, PipelineVariable, and OutVariable. For more information, see
    about_CommonParameters (https://go.microsoft.com/fwlink/?LinkID=113216).
```

- INPUTS

- OUTPUTS

- NOTES

```
Outputs (Windows):
- desktop/src-tauri/target/release/bundle/msi/*.msi
- desktop/src-tauri/target/release/bundle/nsis/*-setup.exe
```

This script sets `PositionalBinding = \$false` to discourage ambiguous invocation.

----- EXAMPLE 1 -----

```
PS > ./scripts/build-desktop.ps1
Builds Core + Desktop, producing both MSI and NSIS artifacts.
```

----- EXAMPLE 2 -----

```
PS > ./scripts/build-desktop.ps1 -Bundles nsis
Builds only the NSIS installer.
```

----- EXAMPLE 3 -----

```
PS > ./scripts/build-desktop.ps1 -SkipCoreBuild -Bundles msi
Fast path when nothing changed in Core.
```

- RELATED LINKS

---

## check-core.ps1

- NAME

```
./scripts\check-core.ps1
```

- SYNOPSIS

```
Runs FaceForge Core quality gates (format, lint, tests).
```

- SYNTAX

```
./scripts\check-core.ps1 [<CommonParameters>]
```

- DESCRIPTION

```
Installs FaceForge Core in editable mode with development dependencies into the repo-local virtual environment (`.venv`), then runs:  
- ruff format --check  
- ruff check  
- pytest
```

This is intended for local verification and CI usage. It will stop on the first failure and return a non-zero exit code.

- PARAMETERS

```
<CommonParameters>  
This cmdlet supports the common parameters: Verbose, Debug,  
ErrorAction, ErrorVariable, WarningAction, WarningVariable,  
OutBuffer, PipelineVariable, and OutVariable. For more information, see  
about_CommonParameters (https://go.microsoft.com/fwlink/?LinkID=113216).
```

- INPUTS

- OUTPUTS

```
Console output from tooling. Exits non-zero on failure.
```

- NOTES

```
Prerequisites:  
- Python 3.12.x is recommended (used only to bootstrap `.` on first run).  
This script does not rely on global Python packaging once `.` exists.
```

```
----- EXAMPLE 1 -----
```

```
PS > ./scripts/check-core.ps1  
Runs format, lint, and tests.
```

- RELATED LINKS

---

## dev-core.ps1

- NAME

```
./scripts\dev-core.ps1
```

- SYNOPSIS

```
Runs FaceForge Core locally in development mode.
```

- SYNTAX

```
./scripts\dev-core.ps1 [<CommonParameters>]
```

- DESCRIPTION

Bootstraps the repo-local virtual environment (`.venv`) if needed, installs FaceForge Core in editable mode with development dependencies, then launches the Core server.

Environment variables (optional):

- FACEFORGE\_BIND: bind host for Uvicorn/FastAPI (default: 127.0.0.1)
- FACEFORGE\_PORT: bind port for Core (default: 8787)

This script is intended for local development. For Desktop orchestration, run the Tauri app and let it start Core.

- PARAMETERS

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see about\_CommonParameters (<https://go.microsoft.com/fwlink/?LinkID=113216>).

- INPUTS

- OUTPUTS

- NOTES

This script intentionally uses `venv\Scripts\python.exe` for all Python execution. External command failures (pip install, server start) cause the script to fail fast.

----- EXAMPLE 1 -----

```
PS > ./scripts/dev-core.ps1
Starts Core at http://127.0.0.1:8787
```

----- EXAMPLE 2 -----

```
PS > $env:FACEFORGE_PORT = '43210'
./scripts/dev-core.ps1
Starts Core at http://127.0.0.1:43210
```

- RELATED LINKS

## set-version.ps1

- NAME

```
./scripts\set-version.ps1
```

- SYNOPSIS

Bumps FaceForge version across Core + Desktop manifests.

- SYNTAX

```
./scripts\set-version.ps1 -Version <String> [-WhatIf] [-Confirm] [<CommonParameters>]
```

- DESCRIPTION

Updates version strings in the following files:

- core/pyproject.toml
- core/src/faceforge\_core/app.py
- desktop/src-tauri/Cargo.toml
- desktop/src-tauri/Cargo.lock
- desktop/src-tauri/tauri.conf.json

The script supports -WhatIf / -Confirm for safe previews.

- PARAMETERS

**-Version <String>**  
The semantic version to set (e.g. 0.1.2).

Required? true  
Position? named  
Default value  
Accept pipeline input? false  
Aliases  
Accept wildcard characters? false

**-WhatIf [<SwitchParameter>]**

Required? false  
Position? named  
Default value  
Accept pipeline input? false  
Aliases  
Accept wildcard characters? false

**-Confirm [<SwitchParameter>]**

Required? false  
Position? named  
Default value  
Accept pipeline input? false  
Aliases  
Accept wildcard characters? false

**<CommonParameters>**

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see about\_CommonParameters (<https://go.microsoft.com/fwlink/?LinkID=113216>).

- INPUTS

- OUTPUTS

- NOTES

If a target file or pattern is not found, the script emits a warning and continues.

----- EXAMPLE 1 -----

```
PS > ./scripts/set-version.ps1 -Version 0.1.2  
Updates all known manifests to 0.1.2.
```

----- EXAMPLE 2 -----

```
PS > ./scripts/set-version.ps1 -Version 0.1.2 -WhatIf  
Shows what would change without modifying files.
```

- RELATED LINKS

---

## update-docs.ps1

- NAME

```
./scripts\update-docs.ps1
```

- SYNOPSIS

```
Builds HTML and PDF renderings for selected Markdown docs.
```

- SYNTAX

```
./scripts\update-docs.ps1 [-ConfigPath <String>] [-Force] [-SkipScriptsReadme] [-WhatIf] [-Confirm]  
[<CommonParameters>]
```

- DESCRIPTION

```
Reads scripts/update-docs_config.json for a list of "actions" and executes them in phases:  
1) copy      (sync files into their canonical docs/ locations)  
2) transform (e.g. Markdown -> HTML)  
3) print     (e.g. HTML -> PDF via headless Chromium)
```

This intentionally does NOT add Node.js tooling to the repo.

- PARAMETERS

-ConfigPath <String>

Required?	false
Position?	named
Default value	
Accept pipeline input?	false
Aliases	
Accept wildcard characters?	false

-Force [<SwitchParameter>]

Required?	false
-----------	-------

```

Position?          named
Default value     False
Accept pipeline input?  false
Aliases
Accept wildcard characters?  false

-SkipScriptsReadme [<SwitchParameter>]
  By default, docs/FaceForge - Dev Guide - Scripts.md is regenerated (from comment-based help) before any doc
  conversions.
  Use this switch to bypass that behavior.

Required?          false
Position?          named
Default value      False
Accept pipeline input?  false
Aliases
Accept wildcard characters?  false

-WhatIf [<SwitchParameter>]

Required?          false
Position?          named
Default value
Accept pipeline input?  false
Aliases
Accept wildcard characters?  false

-Confirm [<SwitchParameter>]

Required?          false
Position?          named
Default value
Accept pipeline input?  false
Aliases
Accept wildcard characters?  false

<CommonParameters>
  This cmdlet supports the common parameters: Verbose, Debug,
  ErrorAction, ErrorVariable, WarningAction, WarningVariable,
  OutBuffer, PipelineVariable, and OutVariable. For more information, see
  about_CommonParameters (https://go.microsoft.com/fwlink/?LinkID=113216).

```

- INPUTS

- OUTPUTS

```

----- EXAMPLE 1 -----
PS > ./scripts/update-docs.ps1

----- EXAMPLE 2 -----
PS > ./scripts/update-docs.ps1 -WhatIf

----- EXAMPLE 3 -----
PS > ./scripts/update-docs.ps1 -ConfigPath ./scripts/update-docs_config.json -Force

```

- RELATED LINKS