



FolderContents

What is in there? How many of what?

FolderContents watches Downloads and adds **Finder Comments** that show what new folders and files contain.

Target Users

FolderContents helps with daily downloads triage. Canva, Etsy, GitHub, or any folder whose name stopped being enough.

What It Shows

FolderContents scans selected items locally and writes the answer into Finder Comments. It does not upload files, require an account, or build a cloud index.

Folders get compact type summaries in the Finder Comments column. Images get pixel dimensions. PDFs get page counts. ZIP files get file counts. CSV, TSV, and JSONL selections get line counts.

```
StickerPack
  png:42, jpg:18, pdf:3

AudioBundle
  mp3:27, mp4:8

ReceiptsExport.zip
  files:167

DataExport.csv
  lines:100
```

Set And Forget

- 1 Enable Watch Downloads. New top-level Downloads items get Finder Comments automatically.
- 2 View Downloads in Finder list view with the Comments column visible.

Manual Checks

Drag items onto FolderContents.app in Finder or the Dock. The optional **Quick Actions > FolderContents** workflow has fewer features; see App And Quick Action.

Install Or Refresh

Turn on Watch Downloads for the no-extra-effort path. The app adds Finder Comments as new top-level Downloads items arrive.

Finder may ask once for permission because FolderContents uses Finder itself to write Finder-visible comments.

Show Finder Comments

Finder may hide Comments. In a Finder list view, right-click the header row beside Name, Date Modified, Size, or Kind, then turn on Comments. FolderContents hints appear beside each item.

Settings

Show Downloads

Show Downloads opens Finder. On first use, a dialog explains how to show the Comments column: switch to List view, Control-click a column heading, and select Comments.

Collapsible Sections

Settings, examples, and Recent Activity each have a disclosure arrow. All start open; choices are remembered across launches. Collapsing Recent Activity hides the log without stopping the watcher. The window stays within the screen's usable height; scroll to reach the bottom controls.

Comment Options

The three checkboxes appear below the divider inside Settings, in this order.

1. Hoist project files - on by default. Put project-kind clues such as **package.json: 1**, **Cargo.toml: 1**, and **xcodeproj: 1** before ordinary type counts. Turn this off to lead with the most common content type instead. Project files still contribute to ordinary counts. Custom HoistByExample rules remain independent.

2. Hoist first line of .txt files to Finder Comments - on by default. Write **text: Meeting notes** on a selected TXT file. A blank first line falls back to the second line. Two blank opening lines produce no preview. This checkbox controls TXT-file previews, not folder project markers or AppendByExample previews. The Quick Action does not process selected TXT files.

3. Append License info - off by default. Add a compact license title after the other folder details when it is useful. Otherwise, keep comments short so Finder's middle truncation leaves more room for type counts. **MIT License** becomes **MIT**; **The MIT License (MIT)** becomes **MIT (MIT)**. The word license (or licence) is removed throughout the title, along with a leading the.

The app checks the folder's own LICENSE, License.txt, or License.md; the app option also controls nested license hints matched by example rules. A blank first line uses the second. Ignore rules still win. Directly selected license files retain their own compact preview even when folder license hints are off.

Shared Settings

The app saves the checkboxes in `~/Documents/FolderContents/Settings/Settings.plist`. The keys are **HoistProjectFiles** (true), **HoistTextFirstLine** (true), and **AppendLicenseInfo** (false). Existing saved choices are preserved. The local app and Quick Action both read the project and license options; TXT previews remain app-only.

Changing a checkbox does not immediately rewrite existing Finder Comments. Drag items onto the app, run the Quick Action on supported selections, or let the watcher process a new download to apply the current choices. Sort Finder by Comments to group items by their leading project kind or major content type. Batch progress belongs in the workflow dialog; it is not added to the comment.

Example Lists And Quick Action

Ignore, Hoist, Append

The app creates three example folders under **Documents/FolderContents**. The filenames of your examples define the rules; their contents do not matter.

IgnoreByExample: examples with an extension ignore that extension; examples without one ignore that filename.

HoistByExample: matching filenames appear as named counts before the common file types.

AppendByExample: matching filenames appear after the common file types. Extensionless and text-file matches can show their first line; other matches show counts.

Ignore takes precedence over Append, and Append takes precedence over Hoist. Edits to example lists are picked up while the app is running. Append previews work independently of the TXT checkbox; license hints additionally require **Append License info**. The Quick Action does not use these example lists.

App And Quick Action

The Finder Quick Action supports folder counts, project-marker hoists, and many image, media, Office, font, database, HTML, and EPUB summaries. It reads **HoistProjectFiles** and **AppendLicenseInfo** from the shared plist each run. The app additionally supports selected PDF and ZIP counts, NDJSON, TXT previews, and example lists. Drag items onto the app for these features.

The workflow processes only the selected supported items; it does not trigger the Downloads watcher or rescan other items. Its dialog provides batch progress. The two versions currently have separate formatters, so type-count limits and tie ordering can differ. The shared settings align project and license choices, but do not yet guarantee identical comment text.

The shared Documents plist is verified with the local Debug app. A sandboxed App Store build needs access to the same Documents location before it can share this file with the workflow.

Advanced

File types

FolderContents chooses a compact Finder Comment tag based on the selected item. These common file types are the easiest clues to recognize at a glance.

Types	Comments
Folders	png:100, pdf:1
PNG, JPG/JPEG, WebP, AVIF, TIFF, HEIC/HEIF	size:1920 x 1080
Animated GIF	size:1920 x 1080, frames:42, duration:3.2s
MP3, FLAC, AIFF, AAC, M4A, WAV	duration:3:42
MP4	duration:3:42
MKV	duration:1:42:08, size:1920 x 1080
EPUB	title:Dune, author:Frank Herbert, sections:24
PDF	pages:12
PPTX	slides:42
XLSX	sheets:3
HTML / HTM	title:'Order history'
CSV, TSV, JSONL, NDJSON	lines:100
TXT (when enabled in the app)	text:Meeting notes
Anki / Anki2	notes:60, cards:60
SQLite, SQLite3, DB	rows:1204 or tables:8
TTF, OTF, TTC, OTC	family:Inter, style:Bold, glyphs:2910
ZIP	files:167
Mac apps	version:1.1, sandbox:no, x86_64:1 (when applicable)
Bundles and project packages	workflow:1, xcodeproj:1

First-slot signals

Some names look like noise to most people. For developers, makers, and technical teams, each one identifies the kind of folder immediately. With **Hoist project files** on, the app and workflow put project markers ahead of ordinary file types. The final x86_64 clue applies to selected Mac apps independently of this option.

SKILL.md	Codex skill
mcp.json	MCP configuration
CLAUDE.md	Claude Code project instructions
AGENTS.md	Codex agent instructions
Dockerfile	Docker container image
Containerfile	Docker or Podman container image
*.xcodproj	Xcode project
Package.swift	Swift package
manifest.json	Packaged app, Chrome extension, or web project
package.json	JavaScript or Node project
pyproject.toml	Modern Python project
setup.cfg	Python package configuration
Cargo.toml	Rust project
Makefile	Command-line build workflow
Podfile	CocoaPods-based Apple project
go.mod	Go module
requirements.txt	Python dependencies
docker-compose.yml / compose.yml	Container composition
x86_64:1	Mac app without Apple silicon support