ApproveJ Cheat Sheet

Quick reference for the public API, organized by the approval flow.

Entry Point

Method	Description
`ApprovalBuilder.approve(T value)`	Start building an approval for a value
`.named(String name)`	Custom name for the approval (used in filenames)
`.printedAs(PrintFormat)` / `.printedBy(Function)` / `.printed()`	Convert value to string
`.scrubbedOf(UnaryOperator<T> scrubber)`	Apply a scrubber to remove dynamic data
`.reviewedBy(FileReviewer)` / `.reviewedBy(String script)`	Set the file reviewer or review script
`.byValue(String previouslyApproved)`	Approve against an inline string
`.byFile()` / `.byFile(PathProvider|Path|String)`	Approve against a file next to the test (or at custom path)
`.by(Function<String, ApprovalResult>)`	Approve using a custom approver function

Print Formats

Factory Method	Description
`SingleLineStringPrintFormat.singleLineString()`	Print via `toString()` on a single line (default)
`MultiLineStringPrintFormat.multiLineString()` / `.sorted()`	Print each property on a new line (optionally sorted)
`JsonPrintFormat.json()` / `.json(ObjectMapper)`	Pretty print as JSON (requires `json-jackson` or `json-jackson3`)
`YamlPrintFormat.yaml()` / `.yaml(ObjectMapper)`	Print as YAML (requires `yaml-jackson` or `yaml-jackson3`)
`ReceivedHttpRequestPrintFormat.httpRequest()`	Print as HTTP request format (requires `http`)

Scrubbers

All factory methods are on the `Scrubbers` class unless otherwise noted.

String Scrubbers

Factory Method	Description
`strings(String first, String… more)`	Scrub specific string values
`stringsMatching(Pattern|String pattern)`	Scrub strings matching a regex pattern
`uuids()`	Scrub UUID strings

Date/Time Scrubbers

Factory Method	Description
`dateTimeFormat(String pattern[, Locale])`	Scrub dates matching a `DateTimeFormatter` pattern
ISO scrubbers — all named `Scrubbers.isoX()`: Dates: `isoLocalDates`, `isoOffsetDates`, `isoDates`, `isoOrdinalDates`, `isoWeekDates`, `basicIsoDates` Times: `isoLocalTimes`, `isoOffsetTimes`, `isoTimes` Date-times: `isoLocalDateTimes`, `isoOffsetDateTimes`, `isoZonedDateTimes`(+locale), `isoDateTimes`(+locale), `isoInstants`, `rfc1123DateTimes`	Scrub ISO-formatted dates, times, and date-times

Field / JSON / HTTP Scrubbers

Factory Method	Description
`field(Class<T>, String fieldName)`	Scrub a field value on objects of the given type
`JsonPointerScrubber.jsonPointer(String)`	Scrub a JSON node at the given JSON Pointer path
`HttpScrubbers.headerValue(String)`	Scrub an HTTP header value by name
`HttpScrubbers.hostHeaderValue()` / `.userAgentHeaderValue()`	Scrub the `Host` or `User-agent` header value

Replacements

Factory Method	Description
`Replacements.numbered([String label])`	Replace with `[label 1]`, `[label 2]`, etc.
`Replacements.labeled(String label)`	Replace with `[scrubbed label]`
`Replacements.string(String replacement)`	Replace with a fixed string
`Replacements.relativeDate()` / `.relativeDateTime()`	Replace with relative date (e.g. `[today]`, `[yesterday]`)
`Replacements.masking()`	Replace characters with `*`

Approvers & Path Providers

Factory Method	Description
`PathProviders.nextToTest()`	Place files next to the test class (default)
`PathProviders.nextToTestInSubdirectory()`	Place files in a subdirectory named after the test class
`PathProviders.approvedPath(Path|String)`	Use a specific approved file path

File Reviewers

Factory Method	Description
`Reviewers.none()`	Do nothing on mismatch (default)
`Reviewers.automatic()`	Automatically accept all received values
`Reviewers.script(String script)`	Run a script with `{receivedFile}` and `{approvedFile}` placeholders

Build Plugin Tasks

Command	Description

`gradle approvejFindLeftovers`
`mvn approvej:find-leftovers`
	List leftover approved files

`gradle approvejCleanup`
`mvn approvej:cleanup`
	Remove leftover approved files

`gradle approvejApproveAll`
`mvn approvej:approve-all`
	Approve all unapproved files

`gradle approvejReviewUnapproved`
`mvn approvej:review-unapproved`
	Review all unapproved files

Configuration Properties

Property	Environment Variable	Default
`defaultPrintFormat`	`APPROVEJ_DEFAULT_PRINT_FORMAT`	`singleLineString`
`defaultFileReviewer`	`APPROVEJ_DEFAULT_FILE_REVIEWER`	`none`
`defaultFileReviewerScript`	`APPROVEJ_DEFAULT_FILE_REVIEWER_SCRIPT`	*(none)*

Priority: env vars > project props (`src/test/resources/approvej.properties`) > user home props (`~/.config/approvej/approvej.properties`) > defaults.