

# Package: RXKCD (via r-universe)

October 1, 2026

**Type** Package  
**Title** Get XKCD Comic from R  
**Version** 2.0.2  
**Date** 2026-10-01  
**Description** Visualize your favorite XKCD comic strip directly from R.  
Includes full-text search with BM25 ranking and semantic  
similarity search via latent semantic analysis, powered by a  
local 'DuckDB' cache.  
**URL** <https://onertipaday.github.io/RXKCD/>,  
<https://github.com/onertipaday/RXKCD>  
**BugReports** <https://github.com/onertipaday/RXKCD/issues>  
**License** GPL-2  
**Encoding** UTF-8  
**Imports** jsonlite, httr, jpeg, png, graphics, utils, DBI, duckdb,  
Matrix  
**Config/roxygen2/version** 8.1.0  
**Config/pak/sysreqs** libjpeg-dev libpng-dev libssl-dev xz-utils  
**Repository** <https://onertipaday.r-universe.dev>  
**Date/Publication** 2026-10-01 09:01:16 UTC  
**RemoteUrl** <https://github.com/onertipaday/rxkcd>  
**RemoteRef** HEAD  
**RemoteSha** 3b3d34e36022be4ae2f75f33eabc360aebd5c44c

## Contents

|                        |          |
|------------------------|----------|
| getXKCD . . . . .      | 2        |
| searchXKCD . . . . .   | 3        |
| similarXKCD . . . . .  | 3        |
| updateConfig . . . . . | 4        |
| <b>Index</b>           | <b>5</b> |

---

`getXKCD`*Get XKCD comic*

---

## Description

Fetches comic metadata and image from the XKCD JSON API.

## Usage

```
getXKCD(which = "current", display = TRUE, html = FALSE, saveImg = FALSE)
```

## Arguments

|                      |                                                                                                                                                                                |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>which</code>   | A single string or a single numeric value. Specifies which comic to retrieve. Can be "current" for the latest comic, "random" for a random comic, or a comic number. Optional. |
| <code>display</code> | A single logical value. If 'TRUE', the comic image will be displayed in the R graphics device. Optional.                                                                       |
| <code>html</code>    | A single logical value. If 'TRUE', the comic's URL will be opened in the default web browser. Optional.                                                                        |
| <code>saveImg</code> | A single logical value. If 'TRUE', the comic image will be saved to a file in the current working directory. Optional.                                                         |

## Value

A list containing details about the XKCD comic (number, title, date, image URL, alt text, link, and transcript). If 'html' is 'TRUE', the list is returned invisibly. If the comic is not found or connection fails, it returns 'NULL' invisibly. Errors will occur if 'which' is invalid or if the XKCD API cannot be reached when attempting to get a random comic.

## Examples

```
## Not run:  
getXKCD(which = 1)  
getXKCD(which = "random")  
  
## End(Not run)
```

---

|            |                           |
|------------|---------------------------|
| searchXKCD | <i>Search XKCD comics</i> |
|------------|---------------------------|

---

**Description**

Searches the local DuckDB cache using full-text search (BM25 ranking) across comic titles, alt text, and transcripts. Results are ranked by relevance. Supports stemming (e.g., "running" matches "run") and stopwords removal. Run `updateConfig()` first to populate the database and build the search index.

**Usage**

```
searchXKCD(query)
```

**Arguments**

|       |                                                                           |
|-------|---------------------------------------------------------------------------|
| query | A single string to search for in comic titles, alt text, and transcripts. |
|-------|---------------------------------------------------------------------------|

**Value**

A data frame containing matching XKCD comics with columns `num`, `date`, `title`, `alt`, and `score` (BM25 relevance), ordered by relevance. If no matches are found, an empty data frame is returned invisibly, and a message is printed. The function will stop with an error if the local database or search index is not found.

**Examples**

```
## Not run:  
updateConfig()  
searchXKCD("satellite")  
  
## End(Not run)
```

---

|             |                                              |
|-------------|----------------------------------------------|
| similarXKCD | <i>Find semantically similar XKCD comics</i> |
|-------------|----------------------------------------------|

---

**Description**

Uses pre-computed latent semantic analysis (LSA) embeddings to find comics that are semantically similar to the given query. Unlike `searchXKCD()`, which matches keywords, this function captures meaning (e.g., "feeling lonely" can match comics about isolation even if they don't contain the word "lonely"). Run `updateConfig()` first to build the embeddings.

**Usage**

```
similarXKCD(query, n = 5L)
```

**Arguments**

query            A single string describing the topic or feeling to search for.

n                A single integer specifying the number of results to return. Defaults to 5.

**Value**

A data frame with columns num, date, title, alt, and similarity (cosine similarity score, at most 1), ordered by similarity. If no similar comics are found, an empty data frame is returned invisibly. The function will stop with an error if the embeddings have not been built.

**Examples**

```
## Not run:
updateConfig()
similarXKCD("space exploration")

## End(Not run)
```

---

|              |                                          |
|--------------|------------------------------------------|
| updateConfig | <i>Update XKCD Database (Smart Sync)</i> |
|--------------|------------------------------------------|

---

**Description**

Connects to the local DuckDB. Checks for missing comics and fetches full metadata (Title, Alt Text, Transcript) from the JSON API for any gaps.

**Usage**

```
updateConfig()
```

**Value**

‘TRUE’, invisibly, if the database is successfully updated or determined to be up-to-date. The function will stop with an error if it cannot connect to ‘xkcd.com’.

**Examples**

```
## Not run:
updateConfig()

## End(Not run)
```

# Index

`getXKCD`, [2](#)

`searchXKCD`, [3](#)

`similarXKCD`, [3](#)

`updateConfig`, [4](#)