

🌟 Watch the Q1 2025 update video about the new features!



JSON - Multi Language

Last updated: February 21, 2025



File format value ⓘ
multi-language-json

[Copy value](#)

[List of supported file formats](#)

One of the most common JSON format for apps and software localization is JSON file with all translations as shown below. It's easy to use because you can fetch only one file and use all translations.

File format example

```
{
  "en": {
    "home.interface.signup": "Signup",
    "home.interface.email": "Email",
    "home.interface.password": "Password"
  },
  "pl": {
    "home.interface.signup": "Zarejestruj się",
    "home.interface.email": "Adres email",
    "home.interface.password": "Hasło"
  }
}
```

You can also use [single-language-json format](#) to download only needed translation messages for the requested language key.

SimpleLocalize also recognizes nested JSON keys, as shown below.

```
{
  "en": {
    "home": {
      "interface": {
```

```

    "signup": "Signup",
    "email": "Email address",
    "password": "Password"
  }
},
"pl": {
  "home": {
    "interface": {
      "signup": "Zarejestruj się",
      "email": "Adres email",
      "password": "Hasło"
    }
  }
}
}

```

Above JSON will result with:

- `home.interface.signup` ,
- `home.interface.email` ,
- `home.interface.password` translation keys in Translation Editor.

Message interpolation

Pluralization and variables formatting is done by your translation library (e.g. [FormatJS](#)) You can use a `{key}` argument for placing a value into the message. The key is looked up in the input data, and the string is interpolated with its value.

Formatting

```

{
  "en": {
    "home": {
      "welcome": "Hello, {name}!"
    }
  }
}

```

Your variable: `name` .

Pluralization

```
{
  "en": {
    "home": {
      "message": "I have {numCats, number} cats.",
      "notifications": "You have {itemCount, plural, =0 {no ite"
    }
  }
}
```

Learn more about [ICU message syntax](#) or use our [ICU syntax playground](#) to test your messages.

Upload translation files

Upload with command-line tool

```
simplelocalize upload --apiKey <PROJECT_KEY> \
  --uploadFormat multi-language-json \
  --uploadPath ./translations.json
```

Learn more about [uploading translations with CLI](#).

Import with REST API

```
curl
  --request POST \
  --url 'https://api.simplelocalize.io/api/v2/import?uploadForm
  --header 'x-simplelocalize-token: <API_KEY>' \
  --form file=@/path/to/your/file.json
```

Learn more about [importing translations with API](#).

Download translation files

Download with command-line tool

```
simplelocalize download --apiKey <PROJECT_KEY> \  
  --downloadFormat multi-language-json \  
  --downloadPath ./translations.json
```

To download translations with nested keys use option `WRITE_NESTED`.
See [all download options](#).

```
simplelocalize download --apiKey <PROJECT_KEY> \  
  --downloadFormat multi-language-json \  
  --downloadPath ./translations.json \  
  --downloadOptions WRITE_NESTED
```

Learn more about [downloading translations with CLI](#).

Export with REST API

```
curl  
  --request GET \  
  --url https://api.simplelocalize.io/api/v4/export?downloadFor  
  --header 'x-simplelocalize-token: <API_KEY>'
```

Learn more about [exporting translations with API](#).

Was this helpful?



Documentation

Table of contents

- ▶ General
- ▶ Translation editor
- ▶ Translation hosting
- ▶ Command-line tool
- ▶ REST API
- ▶ Integrations
- ▼ File formats
 - JSON - SimpleLocalize
 - JSON - Single Language
 - [JSON - Multi Language](#)
 - Excel - translations
 - CSV - translations
 - TSV - translations
 - Markdown
 - Translation as file
 - Java properties
 - PHP Array
 - Android Strings
 - Android XML
 - XLIFF 1.2
 - Localizable Strings



 GitHub

 YouTube

 Discord

 [Email us](#)

 [Auto mode](#)

Product

[Translation Editor](#)

[Auto-translation](#)

[AI Actions](#)

[Automations](#)

[Translation Hosting](#)

[Alternative Translations](#)

[VS Code Extension](#)

[Changelog](#)

Solutions

[Auto-translate JSON](#)

[Translation management](#)

[Excel files localization](#)

[Continuous localization](#)

[React.js localization](#)

[Next.js localization](#)

[Hosting for translations](#)

[Translation feedback](#)

Resources

[About us](#)

[Brand assets](#)

[Privacy policy](#)

[Terms of service](#)

[Refund policy](#)

[Security](#)

[Service status](#)

Integrations

[JavaScript localization](#)

[Multi-language JSON](#)

[SwiftUI localization](#)

[Android Strings](#)

[PO Files](#)

[Laravel](#)

[Figma](#)

[i18next](#)

[FormatJS](#)

[Learn](#)

[Blog](#)

[Documentation](#)

[Affiliate program](#)

[Contributor program](#)

[World data resources](#)

[ICU message builder](#)

[Auto-translation calculator](#)

[Compare](#)

[POEditor](#)

[LingoHub](#)

[Localazy](#)

[Lokalise](#)

[Locize](#)

[Phrase](#)

[Localizely](#)

[Crowdin](#)

[XTM Cloud](#)

Copyright © 2025 Jakub Pomykała Software registered in Poland. All rights reserved. All trademarks are the property of their respective owners. [RSS 2.0 Sitemap \(XML\)](#) [Sitemap](#)