

twemojis v1.1

Jost Rossel

February 8, 2021

Abstract

Simple wrapper which allows to use Twitter's open source emojis through LaTeX commands. This works on images, so no fancy unicode-font stuff is needed and it *should work*[™] on every installation.

1 Available Commands

`\twemoji` basically includes a PDF graphic of the emoji via `\includegraphics`. Per default `\includegraphics` is given the parameter `height=\defaultHeight`, where `\defaultHeight` is set to `\fontcharht\font'X` (i.e. the height of the letter X). This means the standard height is fixed document-wide. `\twemoji{smile}` with default settings looks like this: 😊.

The default height can be changed via `\setlength{\defaultHeight}{...}`, this affects all `\twemoji` usages. With the height set to `0.6em` `\twemoji{smile}` looks like this: 😊.

The look of a single emoji can be changed via an optional parameter (everything `\includegraphics` allows works). Example 🧐: `\twemoji[height=\defaultHeight, angle=45, trim={1mm 1mm 1mm 1mm}, clip]{smile}`.¹

`\twemoji` works perfectly fine in a Tikz picture:



`\texttwemoji` adapts itself to the text height (`\twemoji` does not). Hence, `\texttwemoji{smile}` looks like 😊, but `{\Large\texttwemoji{smile}}` looks like 😊. It does not take optional parameters.

¹`\defaultHeight` needs to be explicitly set, as the usage of the optional parameter overwrites the default one.

2 Finding Emoji Names











There are three ways to get the emoji you want. If you use a string—as a parameter for either command—that does not match any defined one, the command will be ignored.

1. The unicode value
 - in hexadecimal,
 - with small letters,
 - and without leading zeros.

If the emojis use combined values (e.g., for skin color) separate them via `-`. The value can be found, for example, here (not every unicode character is supported, only those you typically find on an “Emoji Keyboard”).

2. The official unicode CLDR names of the emoji. These can also be found here or here.
3. The easiest names to remember (used by Signal, WhatsApp, Discord, etc) like `:smile:` can also be used, but only support a subset (e.g. no variations like skin colors). These can be found, for example, here (without the `:`).

Examples.

```
 \twemoji{1f483}
 \twemoji{woman dancing}
 \twemoji{1f604}
 \twemoji{grinning face with smiling eyes}
 \twemoji{smile}
 \twemoji{1f487-1f3fb-200d-2642-fe0f}
 \twemoji{man getting haircut: light skin tone}
 \twemoji{1f608}
 \twemoji{smiling face with horns}
 \twemoji{smiling_imp}
```

3 Implementation Notice

As the emojis are PDF-based and use transparency, they include groups. I was not able to change the inkscape export so that that does not happen. Hence, pdf_{flat}ex throws a warning about multiple groups on one PDF page. This package disables said warning: `\pdfsuppresswarningpagegroup=1` as it is of no concern for the inputted emojis. It **could** happen, that a PDF you input is faulty and you don't notice due to said suppression. You can enable the warning with `\pdfsuppresswarningpagegroup=0`.

4 License

4.1 LaTeX Package

Copyright © 2021 Jost Rossel

LaTeX Project Public License, version 1.3 or later: <http://www.latex-project.org/lppl.txt>

4.2 Emojis

Copyright © 2019 Twitter, Inc and other contributors

Graphics licensed under CC-BY 4.0: <https://creativecommons.org/licenses/by/4.0/>

4.2.1 Attribution

I'm no lawyer, so take this section with a grain of salt.

As the emojis themselves are licensed under CC-BY they require attribution but are open to be distributed and modified anyway you like. I'm not sure whether the attribution in this file is enough for a file (e.g. PDF) generated with this package to be covered as well.

I see two different possibilities to attribute the emojis in a compiled document:

- Add attribution to the metadata of the document.
For example with <https://www.ctan.org/pkg/hyperxmp> in PDF files.
- Add attribution in the document directly (e.g. on the last page of a presentation).

I consider this option to be the safest.