

Twilio Documentation Style Guide 2.0

Updated August 22, 2022

This is the primary reference for the key elements of Twilio documentation written and presentational style. It is not intended to show you how to write Twilio documentation, but to help you ensure that your writing is consistent with other Twilio docs. It focuses on the key elements of textual formatting, grammar, and usage. Common sources of error are included to indicate the correct usage; there are examples of right and wrong usage too. A dictionary is provided that shows how we apply certain specific words and phrases.



Note: This guide will be updated as usage issues emerge and are resolved. Please contact [Developer Education](#) if you have additions you would like to make.

Formatting

Page titles

- Page titles should be capitalized on all major words, e.g., “Push Inbound Messages and Status Updates to Salesforce”.
- Use the present tense and active verbs:
 - **YES** “Push Inbound Messages and Status Updates to Salesforce”.
 - **NO** “Pushing Inbound Messages and Status Updates to Salesforce”.
- Do not add end punctuation.
- Keep titles short: 50-60 characters works best for SEO.
- URL slugs (set in the promote tab) should *only* use lower-case characters – capitalized characters in a slug will cause the Wagtail page to 404.

Headings

- Headings must be presented in sentence-case, ie. initial capital only, e.g., “Set up your account”.
 - Proper nouns should be capitalized, e.g., “Make a new UsageRecord resource”.
- Make explicit what the section is about or what it will tell the reader.
- Use the present tense and active verbs.
- Do not add end punctuation.
- Never use the ampersand (&) character in a heading unless you are referring to an entity that includes this character, e.g., “Twilio signs mega deal with Holland & Barrett”.
- Only use headlines sizes to represent document structure:
 - H1 (Markdown: #) — Page Title.
 - H2 (Markdown: ##) — Major page sections. Generates left-hand nav entry.
 - H3 (Markdown: ###) — Sub-sections. Generates left-hand nav entry.
 - H4 (Markdown: ####) — Only if required. Use for headings you DON’T want to show up in the left-hand nav (menu overload)
 - H5, H6 — Do not use.
- Do not include `code` formatting in headlines, even for the names of tools and files.

Paragraphs

- Do not begin paragraphs with an indentation.
- Wagtail’s various editors will space paragraphs for you, so you only need hit the **Enter** key once between paragraphs.

Ordered and unordered lists

- Always use ordered (numbered) or unordered (bulleted) lists rather than long (four or more) comma- or semicolon-separated lists within a paragraph.
- Use ordered (numbered) lists for sequential items, such as steps in a procedure.
- Use unordered (bulleted) lists for related items which have no set sequence.
 - But if you are listing items, e.g., properties of an object, try to list them in alphabetical order to help readers zoom in on the ones they are interested in.
- Lists items that are sentences or phrases should end with a period. This applies to all items, not just the last one
- Lists that comprise single words, or phrases fewer than three words should have no end punctuation on any line.
- Don't start a line with a numeral: rewrite to avoid this or spell out the number.
- Do not use semicolons, commas or conjunctions ("and" or "or") at the end of list items.
- Items should begin with a capital unless there is a clear reason not to: for example, a list of commands which would be entered lowercase.
- Keep the items in the list consistent: if most of them are of the form <item> <detail>, for example, then they should *all* be of the form <item> <detail>.
- For lists entries of the form <item> <detail>, the <item> should be in bold and <detail> should be presented as a sentence, ie. capitalized and unemphasized.
 - Do not add end punctuation, specifically a colon, to the <item>.
- Indented lists can be of either type and need not match the main list.
- With nested lists, try not to indent more than two indents. If you need to do more, consider breaking the list into a series of headlined lists.

Italics

- Use italics for the titles of books, magazines, newspapers, movies, TV shows.
 - The titles of shorter works — blog posts, articles, reports — should not be italicized: place the title in double quotes.
- Use italics for general emphasis, but use it sparingly. Ask yourself:
 - Do you really need to emphasize the text?
 - Would a **Callout** better draw attention to the text?

Bold

- Use bold for user interface items that readers will interact with, ie. the title text of buttons, and the names of tabs, menus, text-entry fields, checkboxes and such.
- The first mention of a resource in API reference documentation.
- The titles of Twilio documents when referenced by other documents:
 - **YES** "For more information, please see [How to Limit Rate Calls](#)".
 - **NO** "For more information, please see our guidance on [limiting rate calls](#)".
 - **NO** "For more information, please see [How to Limit Rate Calls](#)".

Code

- Use code format for text or other values that a user will enter into a user interface, e.g., the contents of a text field, as a command line tool argument, or as an element in code.
 - Do not include quote marks unless the user also has to enter quote marks.
- Filenames and filename extensions, e.g., "Now open `test.md` in your text editor".
 - File extensions when presented as part of a file name, as in the example above, should be lower case.
 - Do not use code formatting when discussing file types, ie. not file extension, and ensure the name is capitalized as per the **Abbreviations** rules:
 - **YES** "For this tutorial you will need a PNG image or a CSV file".
 - **NO** "For this tutorial you will need a .png image file".
- Computer code identifiers when referenced in a document, e.g., "We pass the variable `unit` into the function `consumeUnits()`".
- Code values with specific meanings: `true`, `false`, `null`, `nil`, `NaN`.
- Do not include the command line prompt, e.g., `$` or `%`, when presenting commands for users to type.

- Placeholders, e.g., for data or values that a developer will themselves enter, should be written in braces (“<” and “>”):
 - **YES** “Don’t forget, you will need to replace <AUTH_TOKEN> with your own Twilio Auth Token.”
- Constants should be presented in UPPER_SNAKE_CASE.
- Inset lines should be inset by two spaces.
- Always double-check any SHAs presented alongside code download links.
 - Use `shasum -a 256 <file_name>` to get the SHA of the file and compare it to the SHA in the text.
- CLI tool names — but make sure you get the correct name. For example, we offer a software tool called Twilio CLI, which is installed using the Node Package Manager bundle `twilio-cli` and is run by end-users as `twilio`. Make sure you use these as appropriate, ie. only use `twilio-cli` when referring to the NPM package, or `twilio` when specifying commands and their arguments.
 - **YES** “Twilio CLI is available as `twilio-cli` from Node Package Manager: `npm -g install twilio-cli`”.
- HTTP verbs, GET, POST, etc.
- Do not include code formatting in headlines, even for the names of tools and files.

File types and formats

- When talking about file types, ie. the format of a given file, use the name of the creator application, if it’s a unique format, or the file type for generic formats. Only use file extensions in this context if it would be ambiguous if you didn’t:
 - **YES** “For this tutorial you will need to download the following JPEG file.”
 - **YES** “For this tutorial you will need to download the following Photoshop file.”
 - **NO** “For this tutorial you will need to download the following .psd file.”
 - **NO** “For this tutorial you will need to download the following .jpg file.”
- Do not use code formatting when discussing file types, ie. not file extension, and ensure the name is capitalized as per the [Abbreviations](#) rules:
 - **YES** “For this tutorial you will need a PNG image”.
 - **NO** “For this tutorial you will need a .png image file”.

Capitalization

- Proper nouns, e.g., product names and unique Twilio API entities, e.g., Account SID and Auth Token.
- Job titles, e.g., Product Manager.
- Agency names, e.g., Department of Justice.
- Fully capitalize abbreviations and initialisms which are pronounced letter by letter:
 - **YES** FAQ, IRS, PBS, EPA, STFU, YAML, NFL, 2400 BCE.
 - **NO** ASCII, RAM, ROM, POTUS.
 - **NOTE 1** There are exceptions where the product name is an abbreviation but specifically started by the owner to be of a specific case, e.g., npm.
 - **NOTE 2** There are exceptions where an abbreviation is not pronounced letter by letter, but is customarily written out in capitals, e.g., JSON, NASA, SIM.
 - **NOTE 3** There are exceptions where an abbreviation commonly includes a lower case ‘e’ in place of ‘embedded’ or ‘electronic’, e.g., eUICC, eSIM, email.
- Initially capitalize abbreviations and initialisms which are pronounced as words, not letter by letter:
 - **YES** Nascar, Defra.
- API resource names, e.g., “Request a new Message resource”.
 - UpperCamelCase multi-word resources, e.g., “Request a new UsageRecord resource”.
- The titles of books, magazines, newspapers, movies, TV shows and documents.
- The first word of every item in a bulleted or numbered list.
- The first word of a sentence.
- Do not capitalize the word after a colon — only a period marks a new sentence.
- Do not capitalize words that seem special or authoritative but are not, e.g., “government agencies” not “Government Agencies”.
- Do not capitalize things that are used by Twilio but are not unique to it — unless you are referring to specific API resources or Twilio products of the same name. These include:
 - webhooks
 - phone numbers
 - API keys
 - API documentation
- Always capitalize Twilio product names:
 - Always capitalize “Console” when referring to the Twilio Console.
 - Always capitalize “App” when referring to the SendGrid App.

Numbers

- Numbers zero through nine should be written in words, e.g., “You will need to enter three values” or “You will make a three-way call”.
- Numbers above nine should be written numerically, e.g., “You will need to enter 11 values.”
- However, be consistent, even if it means overriding the above rules: if one number in a group requires a numeral, ensure all group members also use a numeral, e.g., “One article has 16 pages, one has 7 pages and a third has only 5 pages”.
- Only include commas in values greater than 999:
 - **YES** “12,345 units”.
 - **YES** “8,976 units”.
- Don’t start a sentence with a numeral: rewrite to avoid this or spell out the number, no matter how it would normally be written:
 - **YES** “Twelve apps are included in the sample code directory”.
 - **NO** “12 apps are included in the sample code directory”.

Dates and Times

- Use the ‘month day, year’ format, e.g., “November 23, 1963”.
 - If you need to include the name of the day, add it at the start and separate it with a comma, e.g., “Saturday, November 23, 1963”.
- Use ‘hour:minute’ format, e.g., “4:15”.
- Use the 12-hour clock, and add “pm” or “am” in lowercase only with no space following the minute:
 - **YES** “4:15pm”.
 - **NO** “16:15”.
 - **NO** “4:15 pm”.
- If you need to include a timezone, add it after the time, e.g., “1:45pm +BST”.
- When referring to decades, include the century. Do not include an apostrophe before the final ‘s’:
 - **YES** “Home computing was really popular in the 1980s”.
 - **YES** “Madonna was one of 1980s’ most popular performers”.
 - **NO** “Twilio’s share price rocketed through the 2010’s”.

Ranges

- Use ‘to’ notation for more general ranges, e.g., “lines 10 to 20 of your code could be refactored”.
- Use dash notation, with an ‘en dash’, for specific numeric ranges, e.g., “1–10 lines of code”.
 - Mac: **Option + -**
 - PC (with numeric pad): **Alt + 0150**
- Ranges can run FROM one thing TO another, or one value THROUGH another value, or take values BETWEEN x AND y — do not mix these forms.
- Add “inclusive” after a range if you need to be explicit that all of the values expressed are included.

Strokes

- When using strokes (/) for either/or options or alternative names for the same entity, do not place spaces on either side of the stroke:
 - **YES** “Charity/501(c)(3) organizations should take advantage of A2P 10DLC.”
 - **NO** “Charity / 501(c)(3) organizations should take advantage of A2P 10DLC.”

Bits and Bytes

- When discussing memory and storage sizes, and data rates:
 - Use these prefixes as needed: uppercase ‘K’ for kilo (not lowercase ‘k’); uppercase ‘M’ for mega; lowercase ‘m’ for milli.
 - Use lowercase ‘b’ for bits and uppercase ‘B’ for bytes when used with a prefix; otherwise write out the word.
 - For speeds, use ‘/s’ for per second.
 - Examples:

- **YES** “1024KB equals 1MB”.
- **YES** “Your application will require as little as 32 bytes of memory”.
- **YES** “Modems could only reach speeds of 56Kb/s in those days”.
- Do not put a space between the number and the units.
- Memory and storage are often confused, in documentation as well as practical usage: ‘memory’ is a short-term store, ie. Ram; “storage” is a long-term store, e.g., hard drive, SSD, Flash, etc.

Links

- Try to include links where you can to content, but ensure the linked text is relevant.
- Try to include links to sections in the document where it makes sense to do so, e.g., to a section beyond sections that experienced users may not need to read.
- Do not include links to the same document multiple times, unless you are linking (by anchor) to specific parts of the linked document.
- Never include the spaces on either side of the linked text in the link.
- Be descriptive: link text should indicate what the links action will perform or what information is being linked to, e.g., “Why not [view our getting started guide?](#)”
- Avoid using just “here” or “learn more” as your link text.
- Never include `https://twilio.com/` in manually entered link destinations, especially when linking to page anchors, as it causes the entire page to be reloaded by the browser.
 - Make sure in-page links only contain, e.g., “#
- Link to section of the page by referencing an anchor, e.g., `#open-your-account`.
 - To find a heading’s anchor, copy its link in a Wagtail preview and delete everything up to the #.
 - Wagtail automatically adds anchors to headings of H3 or above.
- When linking to other Twilio pages, use Wagtail’s internal links where possible as these will deal with changes of page location.

Quotation marks

- Do not use quote marks in place of other formatting. For example, a variable name in a block of code should always be presented in `code`, not in quotes; text that a user should key in should also be in code format.
- Full sentence quotations should include the end punctuation within the quotes marks, but quoted phrases should not.
 - **YES** Jeff said: “We will continue to grow as a company.”
 - **YES** “We will continue to grow as a company,” said Jeff.
 - **YES** Twilio will “continue to grow”, its CEO revealed today.
- The placement of question marks specifically should be led by the context:
 - **YES** “If you have ever asked yourself the question “What is the difference between SMS and MMS?”, we have the answer for you”
 - **YES** “What did you think about her statement that “No one uses SMS these days?””
- When quoting other documents, use double-quotes.
- When using quotes to highlight a colloquialism, use single quotes.
- Always use typography — “curly” — quote marks:
 - Mac left single: **Option +]**
 - Mac right single: **Option + Shift +]**
 - Mac left double: **Option + [**
 - Mac right double: **Option + Shift + [**
 - HTML left single: `‘`;
 - HTML right single: `’`;
 - HTML left double: `“`;
 - HTML right double: `”`;

Abbreviations

- Abbreviations may be common terminology, but not to developers who may be new to our products and services so always spell out abbreviations for specialist jargon.
- Do not spell out abbreviations that are more recognizable than the full words, e.g., SMS, NASA, unless you specifically need to do so.

- If you expect to refer to a phrase throughout the text and want to save space, write the text in full and include the abbreviation in brackets afterward, e.g., “Home Service Gateway (HSG)”.
 - On long documents, it is polite to reacquaint the reader with an abbreviation’s meaning at a later point. Use the same format as above.
 - Always capitalize the words forming the abbreviation, as in the example above. If the first letter of a word is *not* part of the abbreviation, or a single word contributes multiple letters to the abbreviation, capitalize the letters that are. For example, “To trigger this action use the SVC (SuperVisor Call) mnemonic”.
- Avoid the possessive form when using an abbreviation — unless the acronym refers to a person or organisation.
 - **YES** “The purpose of the FAQ.”
 - **NO** “The FAQ’s purpose.”

Notes

- When you are including a key point as a note, first consider whether it should really appear in a [Callout](#). If a point is sufficiently important to highlight as a note, i.e, any content prefixed with something like “Note that...” or “Please note...”, it will almost always be better presented as a Callout: it will be more visible and therefore more likely to draw the attention of the reader. If the point is not sufficiently important to be a Callout, it probably does not require emphasis.
- If you do include a non-Callout note, prefix the paragraph with **Note** in bold and then write the body of the note as a new, unemphasized sentence:
 - **YES** “**Note** Beware that all new users may not have accepted the account terms yet.”
 - **NO** “**Note** that all new users may not have accepted the account terms yet.”
 - **NO** “Please note that all new users may not have accepted the account terms yet.”
- When editing an article in Wagtail, consider migrating **Note** paragraphs to [Callouts](#).
 - If you do this, make sure you remove the original note.

Callouts

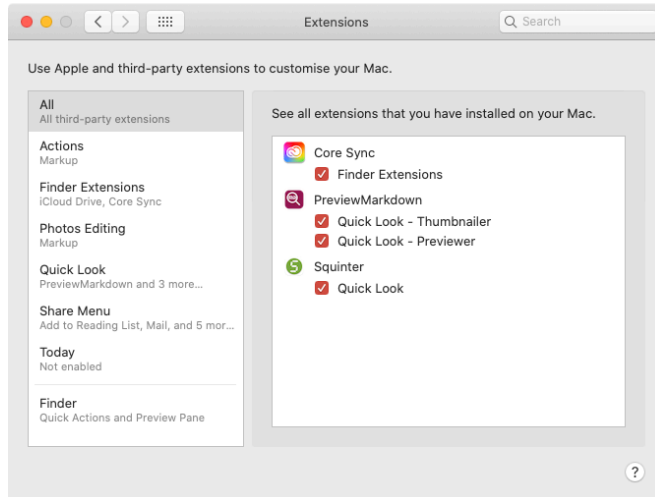
- Wagtail Callout panels are the best way to highlight key points you specifically want the reader to be aware of.
 - Use **Info** to present valuable information, to direct readers to related content, or to tell they can skip one or more steps.
 - Use **Warning** to warn readers about potential issues they may experience, or to alert them to a product’s preview or beta status.
 - Use **Danger** to warn readers about deprecated endpoints and APIs and potentially risky actions, such as including authentication information in production apps.
 - Callouts stand out with color flashes that clearly indicate their level of importance:
- Do not include highlight words, e.g., **Note**, in the callout body — the callout itself is sufficient to indicate the text is something for the reader to specially note.

Pictures

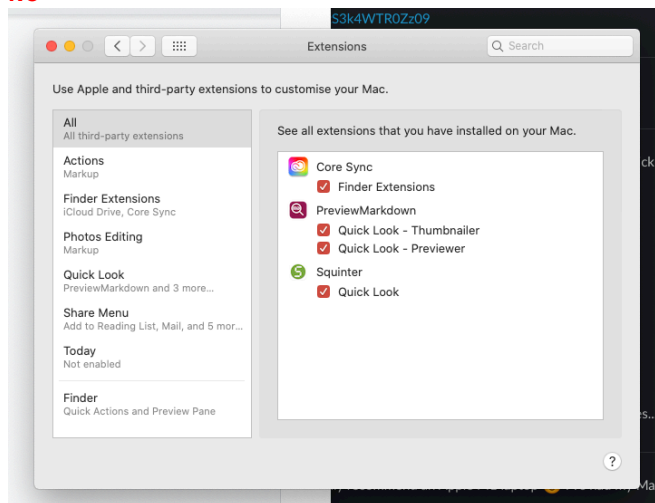
- Screenshots should be **avoided** unless they really do make a point more clearly than text can, or the UI is very complex.
- Screenshots can quickly go out of date because UIs change frequently.
- Most users will be working in the UI while reading your guide, so they can usually see and find what you are writing about.
- The following cases do not need a screenshot:
 - Install and uninstall wizards — unless there is something really unusual about a lot of the settings.
 - Login pages.
 - Standard dialog windows (open, save).
 - Long drop-down lists, e.g., to select a language or a country.
 - Individual buttons — use text.
 - Anything that is already described in detail or included in other screenshots.
 - Screens containing details about the version, rights, etc. If a new version comes out, you may need to change numbers or letters on every single screenshot.
 - Pieces of code, configs, etc. Instead, insert these as formatted text – it looks better, and users can easily copy them.
- When you do use a screenshot:
 - Do not include any window shadow in the grab.
 - Do not take Mac grabs in Dark Mode.
 - Make sure you obscure, e.g., blur out, confidential information, e.g., API keys, in a screenshot before uploading.

- Crop down to the key user interface element to show what the user will control, but make sure you don't lose context. Don't zero in on the control, but on the dialog box or page section it sits inside.
- Only use PNG, GIF or JPG files.
- Do not use screenshots showing the entire screen unless that is what you need to show.
- If you need to show a resizable app window, set it to the standard size of 1280 x 800 and then screen grab.
- If you are grabbing a window or dialog, make sure you get its entire frame, but do not include small parts of the background:

• **YES**



• **NO**



- Take a grab using macOS Screenshot app, in **Applications > Utilities**. Click on the **Capture Selected Window** button in the Screenshot UI.
- Include a clear but brief caption for the image's ALT tag — this is very important for users with visual impairment who rely on text-to-speech.

Tables

- Table headers should be in upper sentence case, i.e. "SDK Release Version".
- Table headers should be in bold, if you are generating them manually (Wagtail rich text and markdown editors will do this for you).
- Do not forget to include appropriate formatting for table entries
 - For example, if a column is listing URL parameters, apply `code` formatting.
- Only columns containing narrow text — e.g., ticks or Yes/No entries, or numbers — should be centered.
 - All other columns should be left aligned.
- Cells containing full sentences should end with a period — all other columns should not.

Punctuation and Grammar

General

- Punctuation is used to indicate the rhythm of a sentence and to help the reader understand its meaning.
- If you are unsure whether to punctuate at a certain point, speak the sentence out loud. Wherever you pause for breath or emphasis, your sentence needs punctuation.
- Do not use a comma to join two independent clauses; use a semicolon. Or start a new sentence.
- You also need to punctuate in order to clarify meaning or eliminate ambiguity. For example, a single comma completely changes the meaning of these five words:
 - “High schools which teach English” — refers solely to the subset of schools that teach the specified language.
 - “High schools, which teach English” — references *all* schools, *all* of whom happen to teach the specified language.
- Do not over-punctuate. If a sentence contains a lot of punctuation, it should probably be broken into multiple sentences.
 - Avoid including many sub-clauses as they can make a sentence very hard to read. Break such sentences down into shorter, separate sentences.

Spelling

- We use **American English** as the basis for our spelling — it’s a good idea to set your writing tools’ dictionaries accordingly. For example:
 - “Dialing” not “dialling”.
 - “Favor” not “favour”.
 - “Install” not “instal”.
 - “Color” not “colour”.
 - “Write someone” not “write to someone”.
 - “Named for someone” not “named after someone”.
 - Always use “ize” rather than “ise”.

Sentences

- Sentences contain a verb.
- Ensure that sentences are not ambiguous, e.g., they don’t confuse objects and subjects. Be aware that a prepositional phrase will include an object, not the subject of your sentence:
 - “After washing my hair, Mark told me about Julie’s arrest.” — Did I wash my hair, or did Mark do it for me? The sentence indicates that Mark did, but this may not be what the writer intended, e.g., “After I washed my hair, Mark told me about Julie’s arrest”.
- Keep sentences short. If you include too many sub-clauses, the reader may lose track.
 - This is especially important for developers whose primary language is not English.
- Separate sentences with a single space, not two or more.

Commas

- We use the Oxford comma in short lists, ie. a comma before the final ‘and’. For example:
 - ‘Apps are available for iOS, Android, and Windows.’
 - ‘The departments affected are Personnel, Events, and Mergers and Acquisitions.’
- Take care with lists that contain sub-lists (as in the example above) that it remains clear which parts are grouped together, e.g., “Mergers and Acquisitions” not “Events and Mergers”. If necessary re-order the list to avoid ambiguity.
- For complex lists, separate major items using semicolons or (better) use an unordered (bulleted) list.
- When using “however”, note the difference between these two examples:
 - “However hard you try, you never quite succeed.”
 - “However, you will never succeed if you don’t try.”
- Before the name of an entity whom you are addressing directly:
 - **YES** “Thank you so much, David.”
 - **NO** “That’s all folks!”

Apostrophes

- Apostrophes indicate a missing character or a possessive case.
 - Remember: “it’s” is short for “it is” whereas “its” is the possessive.
- Apostrophes are useful to give language a greater feeling of informality and you should take advantage of this to make the docs more engaging and accessible:
 - “We can’t delete your account” is less formal or harsh than “We cannot delete your account”.
- Possessive plurals take the apostrophe after the final “s”, e.g., “The cows’ eyes followed me as I strolled across the field.”
- Do not include apostrophes in plurals unless you are using the possessive form. This includes abbreviations:
 - **YES** “I just bought 100 PCs”.
 - **NO** “I just bought 100 PC’s”.

Dashes

- Use dashes to separate groups of words, typically semi-independent sub-clauses.
- Use long dashes (“em dash”) — and do not use two hyphens (--) :
 - Mac: **Option + Shift + -**
 - PC (with numeric pad): **Alt + 0151**
- Make sure you replace hyphens and hyphen pairs used as dashes with em dashes.
- Dashes should be separated from surrounding words with a single space:
 - **YES** “You know — I’m sure you do — that Twilio has great APIs”.
 - **NO** “You know—I’m sure you do—that Twilio has great APIs”.

Brackets (Parentheses)

- Brackets should only be used to separate out words or very brief phrases which act as clarifications of the previous text, or to indicate a definition of an abbreviation. In fact, using em-dashes in place of brackets is the preferred formatting.
- Do not use brackets for sentences or (worse) whole paragraphs. Create a new sentence, use a semicolon, make a new paragraph or consider presenting the content as a [Callout](#) instead.

Ellipses

- Do use ellipses (“...”) to indicate omitted text in a quote or code in technical content.
- Always attach an ellipsis to the previous word and put a single space between the ellipsis and the next word:
 - **YES** “The project requires resistors, capacitors, diodes... the list of parts is endless”
- Avoid using ellipses to suggest something unspoken:
 - **NO** “But of course we’d never do that, would we...”

Exclamations

- Avoid overuse of exclamations — they really should only be used for bangs and crashes, not as a general signifier of informality.
- Only use one exclamation at a time:
 - **YES** “Well done! You’ve completed this tutorial”.
 - **NO** “Well done!!! You’ve completed this tutorial”.
- But that’s for written documentation. For things like Slack comments and email, knock yourself out!!!

Quantities

- We write about an amount of a substance and a number of things, but never the other way round:
 - **YES** “This pie recipe uses a considerable amount of flour”.
 - **YES** “These biscuits contain a number of ingredients”.
- Generally use “fewer” for quantities of things, “less” for quantities of substances:
 - **YES** “Twilio has fewer than ten competitors”.
 - **MAYBE** “Twilio has less than ten competitors”.
 - **NO** “I put fewer than ten grams of sugar in my coffee”.

- **YES** “I put fewer than ten spoons of sugar in my coffee”.
- **YES** “I put less than ten grams of sugar in my coffee”.

Special Situations

Style

- The Twilio writing style is informal but authoritative. We aim to be clear and concise, and therefore accessible to developers of all abilities, whether English is their first language or not.
- Always write in a plain-English style, not an academic or technical style. Say what you need to say in as few words as possible.
- For fuller guidance, please see the [Twilio Documentation Writers' Guide](#).

Person

- Use second-person pronouns (you, your) and avoid plural first person (we, us)
 - We/us is confusing in documentation, where sometimes "we" refers to the company, Twilio, and other times "we" is used to be friendly and suggest "we, you the reader and I the writer."
- See the Microsoft Grammar Guide for more information on Person: <https://docs.microsoft.com/en-us/style-guide/grammar/person>

Inclusion

- Do not include terminology which is, or might be perceived to be, derogatory to members of a minority group, a race, a nation, or a gender identity.
- Do not include terminology which directly or indirectly associates a negative action or event with membership of a minority group, a race, a nation, or a gender identity.
- Do not use language which is not in itself derogatory but is nevertheless not inclusionary because it implies an inequality between racial groups, minorities or gender identities.
 - Replace such terms with neutral alternatives that do not include the at-issue word or term.
- Avoid gender-specific pronouns.
 - Use neutral terms instead, such as “they” and “them”.
- Avoid using example user identities — these can easily imply unintentional bias and do not readily translate to other nationalities.
 - First names: Joey, Charlie.
 - Surnames: Grey, Green.
 - If you must include example user identities, use gender- and race-neutral names.

User interface components

- We discuss user interfaces whenever we refer to the Console or to a third-party UI, such as a cloud service or an app, be it desktop or mobile.
- The names of UI items, e.g., buttons, menus, menu items, tabs, screens or pages, should be presented in **bold**, e.g., “Select the **SIMs** tab.”
- UI pathways may be used to save space where the pathway is straightforward, ie. it will be immediately clear to the reader what they should click next. Use > to separate items on the pathway:
 - e.g., **File > Preferences... > Text Settings > Input.**
- Sample or actual content that the user must type should be presented in `code`, e.g., “In the **Variable** field, type in `date_and_time`.”
 - Don't include quotes in string values unless they should be included by the reader.
- Match the case of the item to its presentation in the UI. If a button is titled START, write it as **START**.
- UI items have specific names — use them:
 - **Menu** — any list of options that pops up or drops down when the user clicks on an object.
 - **Field** — any area in which the user keys in alphanumeric characters.
 - **Tab Bar** — a horizontal array of titles; clicking on any one will present a page or sub-page of options.
 - **Tab** — an item in a Tab Bar.

- **Checkbox** — a single, independent item that can be ticked, crossed or filled (or unticked) with a click.
- **Radio buttons** — a group of related checkboxes that list mutually overriding settings.
- **Switch** — a checkbox that is presented as an on/off switch.
- **Toggle** — alternative term for Switch.
- **Search** — any text-entry field used for searching/finding
- When discussing operations at the command line, refer to directories not to folders.
- When discussing operations within a GUI, refer to folders not directories.
- Avoid using qualitative terms when referencing UI elements, unless they are part of an element’s displayed identifier:
 - **YES** “Enter Twilio in the search box”.
 - **YES** “Enter Twilio in the **Fast Search** field” — because that’s what it’s called.
 - **NO** “Enter Twilio in the quick find box” — who says it’s quick?

Twilio Entities

- API resources should be presented in upper camel case, e.g., “The ExportConfig resource”.
 - Take care when resource names match a commonly used term: e.g., “Create a Message resource to send a message”.
- Set the first body copy reference to an API resource in bold the first time it is mentioned.
- When linking to API documentation from a resource, do not include “resource” in the link:
 - **YES** “The [ExportConfig](#) resource will allow you to receive daily data updates”.
 - **NO** “The [ExportConfig resource](#) will allow you to receive daily data updates”.
- Take care with single-word resources to make it clear when you are discussing a resource and when you are using the word generically, e.g., “Remember, you can schedule a new job by posting a Job resource”.

Specific words and phrases

- “US” not “U.S.”, “U.S.A.” or “USA”.
- “Ascii” not “ASCII” (as it’s pronounced ‘askey’).
- “GitHub” not “Github”, but “git” (for the command line tool) not “Git”.
- “End-user” not “end user” (because they don’t use an end).
- Use “for example” in preference to “e.g.”, “eg.” or “e.g.”
 - But if using the abbreviated form, use “e.g.”
- Always use “ize” rather than “ise”.
- “Adapter” not “adaptor”.
- “Mac” not “MAC” when referring to Apple computers.
- “MAC” not “Mac” when referring to Ethernet adapter addresses.
- “Boolean” not “boolean” — it’s a proper noun — but when discussing code you may refer to a “bool”.
- Terminal — Use the lowercase form, “terminal”, to represent generic terminals, ie. desktop instances of the command line interface. Only use the capitalized form, “Terminal”, when you are specifically referring to the macOS app of the same name or a similar branded product on another platform.
- Companies are singular:
 - **YES** “Twilio has launched an amazing new product”.
 - **NO** “Twilio have launched an amazing new product”.
 - **YES** “Twilio, you say? Yes, it makes great products”.
 - **NO** “Twilio, you say? Yes, they make great products”.
- “Real time” is two words.
- Use “can not” or (best) “can’t”, not “cannot”.
- Do not describe processes as “simple” or “easy”, and products as “easy to use”. These terms are not inclusive and can be viewed as patronising.
- “Third-party” not “third party” — unless you are actually discussing party number three.
- Always place “etc” after a comma and include a period: “, etc.”
- “Changelog” not “change-log” or “change log”.
 - Make sure page URL slugs match this style, please!
- “macOS” not “Mac OS X”, “OSX” or “OS X”
- For JSON data, use the term “object” for a key-value pair, and “property name” (not “field” or “key”) and “property value”.
 - You can use “name” and “value” as shorthand for them within the same paragraph, or if the context is clear.
 - **YES** “The JSON you get back from the API has a property with the name `url` — its value is the location of the asset.”
 - **NO** “you’ll receive JSON from the API. Look for the key `status` and check its content to see if your request was successful.”

Additional Resources

- [Important Considerations and Information When Posting on the Twilio Blog](#)
- [Twilio Documentation Writers' Guide](#)