

Create a robots.txt file

☑ **If you use a site hosting service, such as Wix or Blogger**, you might not need to (or be able to) edit your robots.txt file directly. Instead, your provider might expose a search settings page or some other mechanism to tell search engines whether or not to crawl your page.

If you want to hide or unhide one of your pages from search engines, search for instructions about modifying your page visibility in search engines on your hosting service, for example, search for "wix hide page from search engines".

You can [control which files crawlers may access](#) on your site with a robots.txt file. A robots.txt file lives at the root of your site. So, for site `www.example.com`, the robots.txt file lives at `www.example.com/robots.txt`. robots.txt is a plain text file that follows the [Robots Exclusion Standard](#). A robots.txt file consists of one or more rules. Each rule blocks or allows access for a given crawler to a specified file path in that website. Unless you specify otherwise in your robots.txt file, all files are implicitly allowed for crawling.

Here is a simple robots.txt file with two rules:

```
User-agent: Googlebot  
Disallow: /nogooglebot/
```

```
User-agent: *  
Allow: /
```

```
Sitemap: http://www.example.com/sitemap.xml
```

Here's what that robots.txt file means:

1. The user agent named Googlebot is not allowed to crawl any URL that starts with `http://example.com/nogooglebot/`.
2. All other user agents are allowed to crawl the entire site. This could have been omitted and the result would be the same; the default behavior is that user agents are allowed to crawl the entire site.
3. The site's [sitemap file](#) is located at `http://www.example.com/sitemap.xml`.

See the [syntax](#) section for more examples.

Basic guidelines for creating a robots.txt file

Creating a robots.txt file and making it generally accessible and useful involves four steps:

1. [Create a file named robots.txt.](#)
2. [Add rules to the robots.txt file.](#)

3. [Upload the robots.txt file to your site.](#)
4. [Test the robots.txt file.](#)

Create a robots.txt file

You can use almost any text editor to create a robots.txt file. For example, Notepad, TextEdit, vi, and emacs can create valid robots.txt files. Don't use a word processor; word processors often save files in a proprietary format and can add unexpected characters, such as curly quotes, which can cause problems for crawlers. Make sure to save the file with UTF-8 encoding if prompted during the save file dialog.

Format and location rules:

- The file must be named robots.txt.
- Your site can have only one robots.txt file.
- The robots.txt file must be located at the root of the website host to which it applies. For instance, to control crawling on all URLs below <https://www.example.com/>, the robots.txt file must be located at <https://www.example.com/robots.txt>. It *cannot* be placed in a subdirectory (for example, at <https://example.com/pages/robots.txt>). If you're unsure about how to access your website root, or need permissions to do so, contact your web hosting service provider. If you can't access your website root, use an alternative blocking method such as [meta tags](#).
- A robots.txt file can apply to subdomains (for example, <https://website.example.com/robots.txt>) or on non-standard ports (for example, <http://example.com:8181/robots.txt>).
- A robots.txt file must be an UTF-8 encoded text file (which includes ASCII). Google may ignore characters that are not part of the UTF-8 range, potentially rendering robots.txt rules invalid.

Add rules to the robots.txt file

Rules are instructions for crawlers about which parts of your site they can crawl. Follow these guidelines when adding rules to your robots.txt file:

- A robots.txt file consists of one or more groups.
- Each group consists of multiple rules or directives (instructions), one directive per line. Each group begins with a User-agent line that specifies the target of the groups.
- A group gives the following information:
 - Who the group applies to (the user agent).

- Which directories or files that agent *can* access.
- Which directories or files that agent *cannot* access.
- Crawlers process groups from top to bottom. A user agent can match only one rule set, which is the first, most specific group that matches a given user agent.
- The default assumption is that a user agent can crawl any page or directory not blocked by a disallow rule.
- Rules are case-sensitive. For instance, disallow: /file.asp applies to https://www.example.com/file.asp, but not https://www.example.com/FILE.asp.
- The # character marks the beginning of a comment.

Google's crawlers support the following directives in robots.txt files:

- user-agent: [Required, one or more per group] The directive specifies the name of the automatic client known as search engine crawler that the rule applies to. This is the first line for any rule group. Google user agent names are listed in the [Google list of user agents](#). Using an asterisk (*) matches all crawlers except the various AdsBot crawlers, which must be named explicitly. For example:
 - # Example 1: Block only Googlebot
 - User-agent: Googlebot
 - Disallow: /
 -
 - # Example 2: Block Googlebot and Adsbot
 - User-agent: Googlebot
 - User-agent: AdsBot-Google
 - Disallow: /
 -
 - # Example 3: Block all but AdsBot crawlers
 - User-agent: *
 - Disallow: /
- disallow: [At least one or more disallow or allow entries per rule] A directory or page, relative to the root domain, that you don't want the user agent to crawl. If the rule refers to a page, it must be the full page name as shown in the browser. It must start with a / character and if it refers to a directory, it must end with the / mark.
- allow: [At least one or more disallow or allow entries per rule] A directory or page, relative to the root domain, that may be crawled by the user agent just mentioned. This is used to override a disallow directive to allow crawling of a subdirectory or page in a disallowed directory. For a single page, specify the full page name as shown in the browser. In case of a directory, end the rule with a / mark.
- sitemap: [Optional, zero or more per file] The location of a sitemap for this website. The sitemap URL must be a fully-qualified URL; Google doesn't assume or check

http/https/www.non-www alternates. Sitemaps are a good way to indicate which content Google should crawl, as opposed to which content it *can* or *cannot* crawl. [Learn more about sitemaps](#). **Example:**

- Sitemap: <https://example.com/sitemap.xml>
Sitemap: <http://www.example.com/sitemap.xml>

All directives, except sitemap, support the * wildcard for a path prefix, suffix, or entire string.

Lines that don't match any of these directives are ignored.

Read our page about [Google's interpretation of the robots.txt specification](#) for the complete description of each directive.

Upload the robots.txt file

Once you saved your robots.txt file to your computer, you're ready to make it available to search engine crawlers. There's no one tool that can help you with this, because how you upload the robots.txt file to your site depends on your site and server architecture. Get in touch with your hosting company or search the documentation of your hosting company; for example, search for "upload files infomaniak".

After you upload the robots.txt file, test whether it's publicly accessible and if Google can parse it.

Test robots.txt markup

To test whether your newly uploaded robots.txt file is publicly accessible, open a [private browsing window](#) (or equivalent) in your browser and navigate to the location of the robots.txt file. For example, <https://example.com/robots.txt>. If you see the contents of your robots.txt file, you're ready to test the markup.

Google offers two options for testing robots.txt markup:

1. The [robots.txt Tester](#) in Search Console. You can only use this tool for robots.txt files that are already accessible on your site.
2. If you're a developer, check out and build [Google's open source robots.txt library](#), which is also used in Google Search. You can use this tool to test robots.txt files locally on your computer.

Submit robots.txt file to Google

Once you uploaded and tested your robots.txt file, Google's crawlers will automatically find and start using your robots.txt file. You don't have to do anything. If you updated your robots.txt file and you need to refresh Google's cached copy as soon as possible, learn [how to submit an updated robots.txt file](#).

Useful robots.txt rules

Here are some common useful robots.txt rules:

Useful rules

Disallow crawling of the entire website	<p>Keep in mind that in some situations URLs from the website may still be indexed, even if they haven't been crawled.</p> <p>Note: This does not match the various AdsBot crawlers, which must be named explicitly.</p> <p>User-agent: *</p> <p>Disallow: /</p>
Disallow crawling of a directory and its contents	<p>Append a forward slash to the directory name to disallow crawling of a whole directory.</p> <p>Caution: Remember, don't use robots.txt to block access to private content; use proper authentication instead. URLs disallowed by the robots.txt file might still be indexed without being crawled, and the robots.txt file can be viewed by anyone, potentially disclosing the location of your private content.</p> <p>User-agent: *</p> <p>Disallow: /calendar/</p> <p>Disallow: /junk/</p>
Allow access to a single crawler	<p>Only googlebot-news may crawl the whole site.</p> <p>User-agent: Googlebot-news</p> <p>Allow: /</p> <p>User-agent: *</p> <p>Disallow: /</p>
Allow access to all but a single crawler	<p>Unnecessarybot may not crawl the site, all other bots may.</p> <p>User-agent: Unnecessarybot</p> <p>Disallow: /</p> <p>User-agent: *</p> <p>Allow: /</p>
Disallow crawling of a single web page	<p>For example, disallow the useless_file.html page.</p> <p>User-agent: *</p> <p>Disallow: /useless_file.html</p>
Block a specific image from Google Images	<p>For example, disallow the dogs.jpg image.</p> <p>User-agent: Googlebot-Image</p> <p>Disallow: /images/dogs.jpg</p>

Block all images on your site from Google Images	<p>Google can't index images and videos without crawling them.</p> <p>User-agent: Googlebot-Image</p> <p>Disallow: /</p>
Disallow crawling of files of a specific file type	<p>For example, disallow for crawling all .gif files.</p> <p>User-agent: Googlebot</p> <p>Disallow: /*.gif\$</p>
Disallow crawling of an entire site, but allow Mediapartners-Google	<p>This implementation hides your pages from search results, but the Mediapartners-Google web crawler can still analyze them to decide what ads to show visitors on your site.</p> <p>User-agent: *</p> <p>Disallow: /</p>
Use \$ to match URLs that end with a specific string	<p>User-agent: Mediapartners-Google</p> <p>Allow: /</p> <p>For example, disallow all .xls files.</p> <p>User-agent: Googlebot</p> <p>Disallow: /*.xls\$</p>