

How to Use wget and curl

Both wget and curl are command line tools for transferring files over the network via various network protocols like HTTP or FTP. Wget is a GNU Project by the Free Software Foundation licensed under the GNU GPL whereas Curl is an independent project licensed under a variant of the MIT license.

Curl is also based on a libcurl library, part of the same project, which makes it more suitable for use in programming various applications. It is generally more flexible and featureful whereas wget is simpler.

Simple download

Here's how to initiate a simple download with both wget and curl.

```
wget
ftp://mirrors.kernel.org/gnu/unifont/unifont-7.0.01/unifont_upper-7.0.01.ttf
```

```
curl -O
ftp://mirrors.kernel.org/gnu/unifont/unifont-7.0.01/unifont_upper-7.0.01.ttf
```

This will download the file to the current working directory with its original file name (which is what the -O option passed to curl is for). You can run curl without any options as well, but it will dump the file contents on to the screen as it downloads.

Download multiple files at once

You can also download multiple files in one go by specifying multiple URLs:

```
wget
ftp://mirrors.kernel.org/gnu/unifont/unifont-7.0.01/unifont-7.0.01.ttf
```

```
ftp://mirrors.kernel.org/gnu/unifont/unifont-7.0.01/unifont_upper-7.0.01.ttf
```

```
curl -O ftp://mirrors.kernel.org/gnu/unifont/unifont-7.0.01/unifont-7.0.01.ttf -O ftp://mirrors.kernel.org/gnu/unifont/unifont-7.0.01/unifont_upper-7.0.01.ttf
```

Specifying `-O` before each URL isn't necessary, but we found that not specifying it results in the second file being downloaded being dumped into the command line.

That said, it is also possible to specify multiple files for download in accordance to a given range of numbers or letters in the file name.

With `curl` you can also download multiple files with sequential numbers or letters in their names like `file1`, `file2`, `file3`, and so on by specifying the range in the brackets:

```
curl -O ftp://example.com/file[1-50].txt
```

Multiple nested sequences are possible too, like this:

```
curl -O http://example.com/[2002-2014]/file{a-z}.txt
```

Resume downloads

Both `wget` and `curl` allow you to resume partial downloads, which is useful if a network outage in the middle of the download interrupted it, for example, and you want to continue where you left off.

With `wget` you simply use the `-c` or `--continue` option. Some have a habit of always passing this option just in case.

```
wget -c ftp://mirrors.kernel.org/gnu/unifont/unifont-7.0.01/unifont-7.0.01.ttf
```

With `curl` you can use the `-C` – options:

```
curl -C -  
ftp://mirrors.kernel.org/gnu/unifont/unifont-7.0.01/unifont-7.  
0.01.ttf
```

That's in a nutshell how to use wget and curl. They're both powerful, curl a bit more powerful than wget, and you can find all about what they can do in their manual pages: *man wget*, and *man curl*.