

CURL / LIBCURL

Librairie Open Source

<https://curl.haxx.se/>



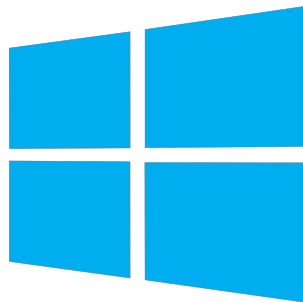
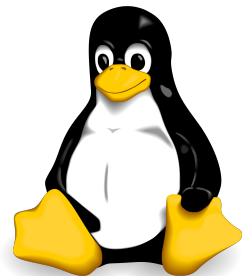
Accéder et transférer des données
en utilisant la syntaxe des URL

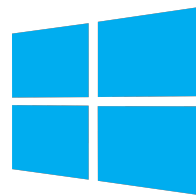
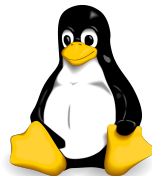


Introduction

Support - Fonctionnalités - Historique

curl://





AIX, AMIGAOS, BEOS, CHROME, DOS, DragonFlyBSD, FreeBSD, GNU Darwin, Hurd, IRIX, Android, Arch, CoreOS, CRUX, Debian, Fedora, Frugalware, Gentoo, GoboLinux, LFS, Lunar, Maemo, Mageia, OpenEmbedded, OpenWRT, PCLinuxOS, PLD, RedHat, SHR, SliTaz, SourceMage, Suse, T2, Tizen, Ubuntu, Angstrom, Midnight BSD, Minix, MiNT, NetBSD, NetWare, Open Server, OpenBSD, QNX, RISC OS, Solaris, Illumos, Tru64 Unix, UnixWare, VMS, Win32, Cygwin MinGW64, ZOS,



Multi Protocol

DICT, FILE, FTP, FTPS, Gopher, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, POP3, POP3S, RTMP, RTSP, SCP, SFTP, SMB, SMTP, SMTPS, Telnet and TFTP.



Utilisable dans vos programmes

Ada95, Basic, C, C++, Ch, Cocoa, D, Dylan, Eiffel, Euphoria, Ferite, Gambas, glib/GTK+, Haskell, ILE/RPG, Java, Lisp, Lua, Mono, .NET, Object-Pascal, O'Caml, Pascal, Perl, PHP, PostgreSQL, Python, R, Rexx, Ruby, Scheme, S-Lang, Smalltalk, SP-Forth, SPL, Tcl, Visual Basic, Visual FoxPro, Q, wxwidgets and XBLite

Depuis 2009 des ajouts comme Node.js



- Première version en 1997
- Principal et premier auteur: Daniel Stenberg
- Écrit en C
- L'origine du nom: “see url” ie. “voir l'url”



Aujourd'hui la librairie supporte

SSL/TLS through OpenSSL, GnuTLS, NSS, wolfSSL, mbed TLS, axTLS

Les cookies, user+password authentication (Basic, Plain, Digest, CRAM-MD5, NTLM, Negotiate and Kerberos)

Les proxies, le proxy tunneling, FTP Resume, HTTP/2 , SSH Agent

Et bien plus ...



Initiation

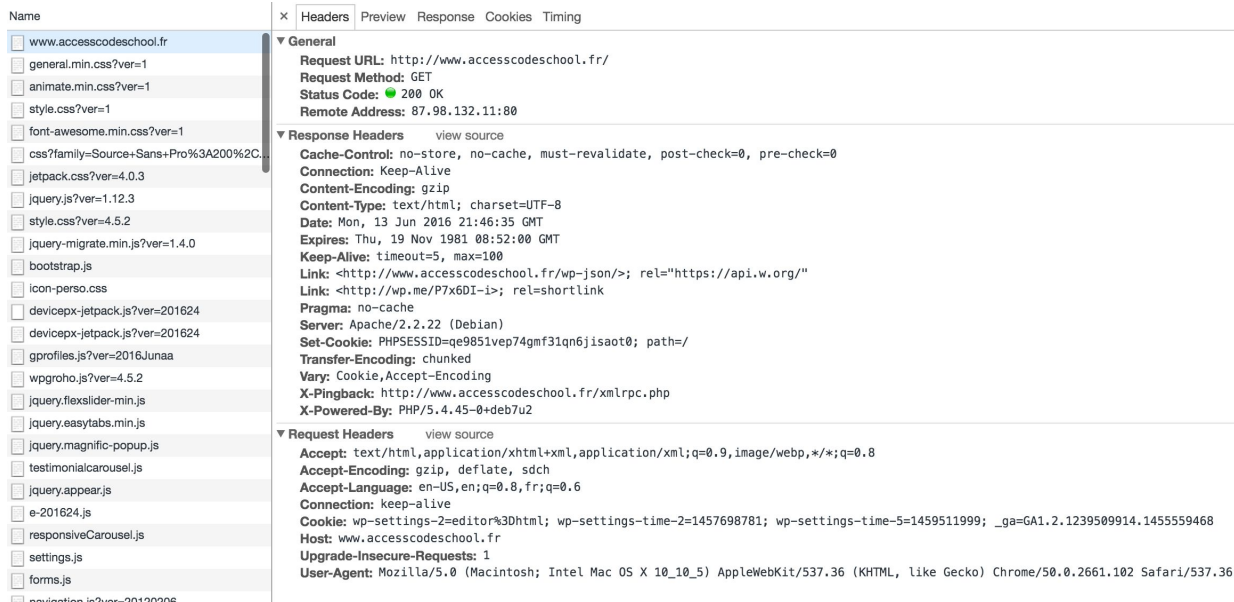
Concept - Prise en main

Protocol HTTP(s)



`http://www.accesscodeschool.fr/`

Protocol HTTP(s)



Name

- ☐ www.accesscodeschool.fr
- ☐ general.min.css?ver=1
- ☐ animate.min.css?ver=1
- ☐ style.css?ver=1
- ☐ font-awesome.min.css?ver=1
- ☐ css?family=Source+Sans+Pro%3A200%2C...
- ☐ jetpack.css?ver=4.0.3
- ☐ jquery.js?ver=1.12.3
- ☐ style.css?ver=4.5.2
- ☐ jquery-migrate.min.js?ver=1.4.0
- ☐ bootstrap.js
- ☐ icon-perso.css
- ☐ devicepx-jetpack.js?ver=201624
- ☐ devicepx-jetpack.js?ver=201624
- ☐ gprofiles.js?ver=2016Junaa
- ☐ wpgroho.js?ver=4.5.2
- ☐ jquery.flexslider-min.js
- ☐ jquery.easytabs.min.js
- ☐ jquery.magnific-popup.js
- ☐ testimonialcarousel.js
- ☐ jquery.appear.js
- ☐ e-201624.js
- ☐ responsiveCarousel.js
- ☐ settings.js
- ☐ forms.js
- ☐ navigation.js?ver=20162020R

Headers Preview Response Cookies Timing

General

Request URL: http://www.accesscodeschool.fr/
Request Method: GET
Status Code: 200 OK
Remote Address: 87.98.132.11:80

Response Headers view source

Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Connection: Keep-Alive
Content-Encoding: gzip
Content-Type: text/html; charset=UTF-8
Date: Mon, 13 Jun 2016 21:46:35 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Keep-Alive: timeout=5, max=100
Link: <http://www.accesscodeschool.fr/wp-json/>; rel="https://api.w.org/"
Link: <http://wp.me/P7x6DI-i>; rel=shortlink
Pragma: no-cache
Server: Apache/2.2.22 (Debian)
Set-Cookie: PHPSESSID=qe9851vep74gmf31qn6jisaot0; path=/
Transfer-Encoding: chunked
Vary: Cookie, Accept-Encoding
X-Pingback: http://www.accesscodeschool.fr/xmlrpc.php
X-Powered-By: PHP/5.4.45-0+deb7u2

Request Headers view source

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8,fr;q=0.6
Connection: keep-alive
Cookie: wp-settings-2=editor%3Dhtml; wp-settings-time-2=1457698781; wp-settings-time-5=1459511999; _ga=GA1.2.1239509914.1455559468
Host: www.accesscodeschool.fr
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/50.0.2661.102 Safari/537.36

Protocol HTTP(s)



▼ General

Request URL: `http://www.accesscodeschool.fr/`

Request Method: GET

Status Code:  200 OK

Remote Address: 87.98.132.11:80

Protocol HTTP(s)



▼ Request Headers [view source](#)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Accept-Encoding: gzip, deflate, sdch

Accept-Language: en-US,en;q=0.8,fr;q=0.6

Connection: keep-alive

Cookie: wp-settings-2=editor%3Dhtml; wp-settings-time-2=1457698781; wp-settings-time-5=1459511999; _ga=GA1.2.1239509914.1455559468

Host: www.accesscodeschool.fr

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/50.0.2661.102 Safari/537.36

Protocol HTTP(s)



▼ Request Headers [view source](#)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Accept-Encoding: gzip, deflate, sdch

Accept-Language: en-US,en;q=0.8,fr;q=0.6

Connection: keep-alive

Cookie: wp-settings-2=editor%3Dhtml; wp-settings-time-2=1457698781; wp-settings-time-5=1459511999; _ga=GA1.2.1239509914.1455559468

Host: www.accesscodeschool.fr

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/50.0.2661.102 Safari/537.36

Protocol HTTP(s)



▼ Request Headers [view source](#)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Accept-Encoding: gzip, deflate, sdch

Accept-Language: en-US,en;q=0.8,fr;q=0.6

Connection: keep-alive

Cookie: wp-settings-2=editor%3Dhtml; wp-settings-time-2=1457698781; wp-settings-time-5=1459511999; _ga=GA1.2.1239509914.1455559468

Host: www.accesscodeschool.fr

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/50.0.2661.102 Safari/537.36

Protocol HTTP(s)



▼ Request Headers [view source](#)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Accept-Encoding: gzip, deflate, sdch

Accept-Language: en-US,en;q=0.8,fr;q=0.6

Connection: keep-alive

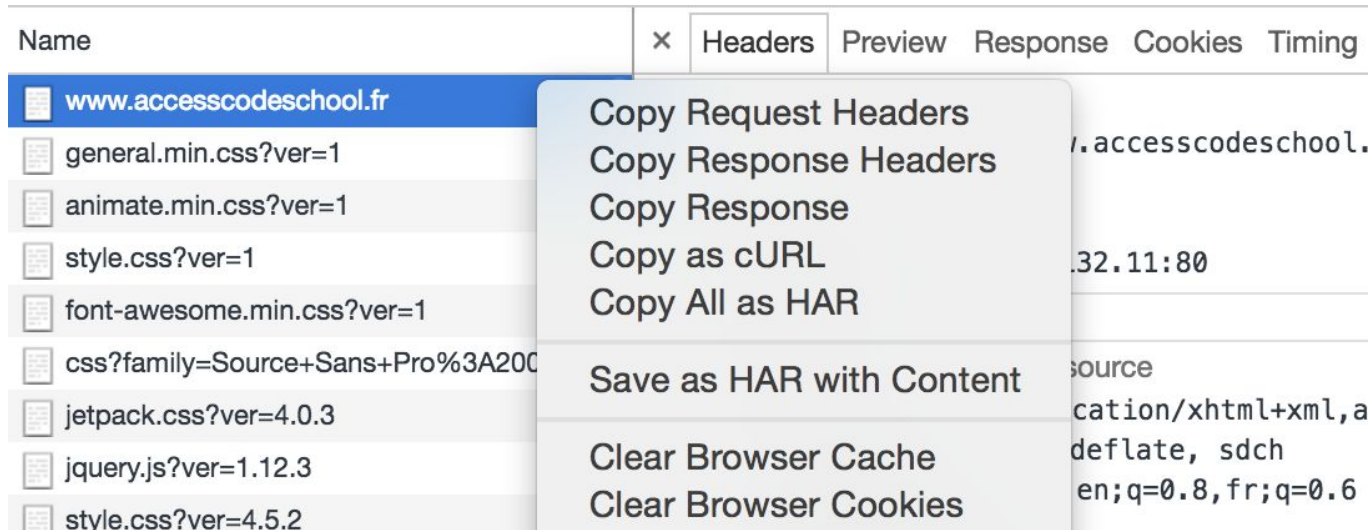
Cookie: wp-settings-2=editor%3Dhtml; wp-settings-time-2=1457698781; wp-settings-time-5=1459511999; _ga=GA1.2.1239509914.1455559468










Host: www.accesscodeschool.fr

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/50.0.2661.102 Safari/537.36

Protocol HTTP(s)



Name	Headers	Preview	Response	Cookies	Timing
 www.accesscodeschool.fr	<div>Copy Request Headers Copy Response Headers Copy Response Copy as cURL Copy All as HAR Save as HAR with Content Clear Browser Cache Clear Browser Cookies</div>				
 general.min.css?ver=1					
 animate.min.css?ver=1					
 style.css?ver=1					
 font-awesome.min.css?ver=1					
 css?family=Source+Sans+Pro%3A200					
 jetpack.css?ver=4.0.3					
 jquery.js?ver=1.12.3					
 style.css?ver=4.5.2					

Protocol HTTP(s)



```
1 curl 'http://www.accesscodeschool.fr/' \  
2     -H 'Accept-Encoding: gzip, deflate, sdch' \  
3     -H 'Accept-Language: en-US,en;q=0.8,fr;q=0.6' \  
4     -H 'Upgrade-Insecure-Requests: 1' \  
5     -H 'User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit  
6     -H 'Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/  
7     -H 'Cookie: wp-settings-2=editor%3Dhtml; wp-settings-time-2=1457698781;
```

Protocol HTTP(s)



Utilisation:

`curl [options...] <url>`

Pour la liste des options:

`curl --help`

Exemple:

`curl https://www.google.com`

`curl -o google.html https://www.google.com`

Protocol HTTP(s)



Votre meilleur ami pour trouver des exemples concrets



Exercice simple

Nom de domaine

Les noms de domaine



Un serveur peut héberger plusieurs noms de domaine.
On les nomme “virtual host”.

Voyons cela avec CURL

Les noms de domaine



Voici une commande simple:

```
curl -vv http://blog.codeur.online
```


Les noms de domaine



```
curl -vv http://blog.codeur.online
```

Curl a automatiquement ajouté "Host: blog.codeur.online" dans la requête

Les noms de domaine



Ajoutons l'option spécifiant le nom de domaine

```
curl -vv http://blog.codeur.online  
--header 'Host: julieng.student.codeur.online'
```

Les noms de domaine



Nous obtenons maintenant la page

`http://julieng.student.codeur.online`



Exercices concrets

User-Agent

User-Agent



Google Fonts

```
<link href='https://fonts.googleapis.com/css?family=Open+Sans'  
      rel='stylesheet'  
      type='text/css'  
>
```

User-Agent



Google Fonts

```
<link href='https://fonts.googleapis.com/css?family=Open+Sans'  
rel='stylesheet'  
type='text/css'  
>
```

User-Agent



```
curl 'https://fonts.googleapis.com/css?family=Open+Sans'
```

```
@font-face {  
  font-family: 'Open Sans';  
  font-style: normal;  
  font-weight: 400;  
  src: local('Open Sans'), local('OpenSans'),  
  url(https://fonts.gstatic.com/s/opensans/v13/cJZKe0uBrn4kERxqtaUH3aCWcynf_cDxXwCLxiixG1c.ttf)  
  format('truetype');  
}
```

User-Agent



```
curl --user-agent 'Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.1' 'https://fonts.googleapis.com/css?family=Open+Sans'
```

```
@font-face {  
  font-family: 'Open Sans';  
  font-style: normal;  
  font-weight: 400;  
  src: local('Open Sans'), local('OpenSans'),  
  url(https://fonts.gstatic.com/s/opensans/v13/cJZKe0uBrn4kERxqtaUH3VtXRa8TVwTICgirnJhmVJw.woff2)  
  format('woff2');  
}
```


User-Agent



```
curl --user-agent 'Mozilla/5.0 (Windows; U; MSIE 9.0; Windows NT 9.0; en-US))' 'https://fonts.googleapis.com/css?family=Open+Sans'
```

```
@font-face {  
  font-family: 'Open Sans';  
  font-style: normal;  
  font-weight: 400;  
  src: local('Open Sans'), local('OpenSans'),  
  url(https://fonts.gstatic.com/s/opensans/v13/cJZKe0uBrn4kERxqtaUH3T8E0i7KZn-EPnyo3HZu7kw.woff)  
  format('woff');  
}
```

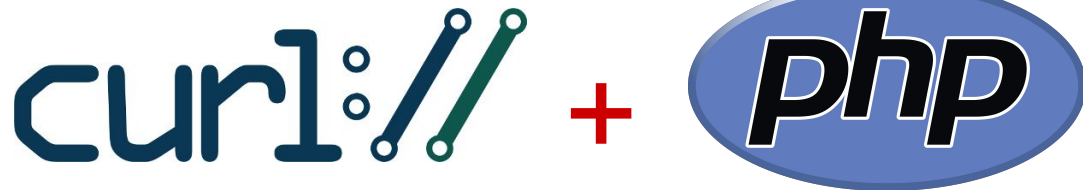


Ok, cool ...
Mais ça reste des lignes de commande
pour récupérer des fonts !!



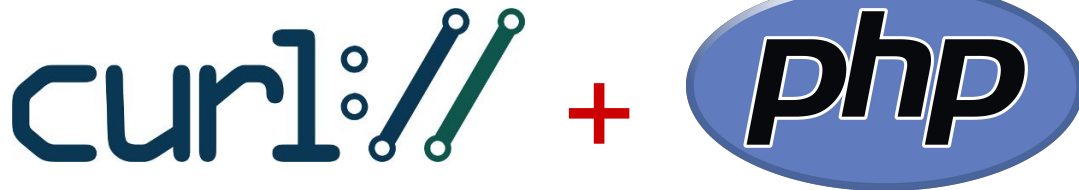
Et si on pouvait coder ça ?

Exemple en PHP



www.php.net/curl

Exemple en PHP



```
1 <?php
```

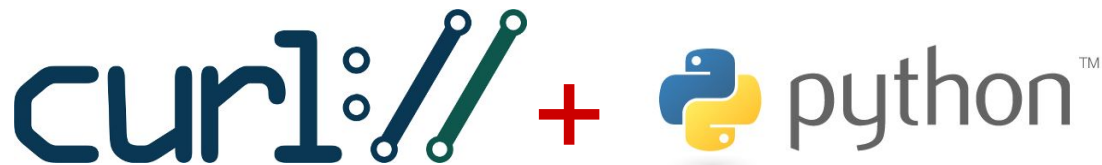
```
2  
3 $url = 'https://fonts.googleapis.com/css?family=Open+Sans';  
4 $useragent = 'Mozilla/5.0 (Windows; U; MSIE 9.0; WIndows NT 9.0; en-US)';
```

```
5  
6 $ch = curl_init();  
7 curl_setopt($ch, CURLOPT_URL, $url);  
8 curl_setopt($ch, CURLOPT_USERAGENT, $useragent);  
9 curl_setopt($ch, CURLOPT_RETURNTRANSFER, TRUE);  
10 $css = curl_exec($ch);  
11 curl_close($ch);
```

```
12  
13 echo $css;
```

```
14  
15 ?>
```

Exemple en Python



```
1 import pycurl
2 from StringIO import StringIO
3
4 url = "https://fonts.googleapis.com/css?family=Open+Sans"
5 useragent = "Mozilla/5.0 (Windows; U; MSIE 9.0; Windows NT 9.0; en-US)"
6
7 buffer = StringIO()
8 c = pycurl.Curl()
9 c.setopt(c.URL, url)
10 c.setopt(c.USERAGENT, useragent)
11 c.setopt(c.WRITEDATA, buffer)
12 c.perform()
13 c.close()
14 css = buffer.getvalue()
15 print(css)
```



Un exemple plus complexe et plus concret

curl://





Objectif:

Télécharger une sauvegarde
d'une base de donnée.



Les besoins:

Avant de s'identifier, nous devons récupérer un Cookie de session. Nous devons donc accéder à la page d'index et mémoriser le cookie.



```
curl --cookie-jar 'mescookies.txt'  
'http://julieng.student.codeur.online/phpmyadmin/index.php'
```

Les besoins:



Ensuite nous devons nous identifier en utilisant le cookie et le nom d'utilisateur et le mot de passe
Envoyés en method post.

```
curl -vv
```

```
--cookie 'mescookies.txt' --cookie-jar 'mescookies.txt'
```

```
--data 'pma_username=julieng&pma_password=mdp&server=1&target=index.php'
```

```
'http://julieng.student.codeur.online/phpmyadmin/index.php'
```

Les besoins:



Nous avons enfin les infos nécessaires, il nous suffit de reproduire la requête.

```
curl -vv  
--cookie 'mescookies.txt' --cookie-jar 'mescookies.txt'  
--data 'db=julieng&token=d6221fec8046e69ec158ff069b625dd9  
&export_type=database[...]'  
  
'http://julieng.student.codeur.online/phpmyadmin/export.php'
```



Retrouvez ces lignes de commande ainsi que l'équivalent en PHP sur github à l'adresse suivante:

<https://github.com/jubry/ACS.git>