

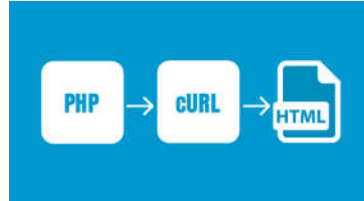
Примеры использования cURL в PHP

14.11.2017 136954 1 комментарий В закладки

27 7

Содержание

- 1 GET запрос
- 2 POST запрос
- 3 PUT запрос
- 4 DELETE запрос
- 5 Запрос через проху
- 6 Отправка файлов на другой сервер
- 7 Скачивание файлов
- 8 Отправить и получить cookie
- 9 Имитация браузера
- 10 HTTP авторизация
- 11 Получить HTTP код ответа сервера
- 12 Если CURL возвращает false



cURL PHP – это библиотека предназначенная для получения и передачи данных через такие протоколы, как HTTP, FTP, HTTPS. Библиотека используется для получения данных в виде XML, JSON и непосредственно в HTML, парсинга, загрузки и передачи файлов и т.д.

1 GET запрос

```
1 $ch = curl_init('https://example.com');
2 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
3 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
4 curl_setopt($ch, CURLOPT_HEADER, false);
5 $html = curl_exec($ch);
6 curl_close($ch);
7
8 echo $html;
```

PHP

GET-запрос с параметрами

```
1 $get = array(
2     'name' => 'Alex',
3     'email' => 'mail@example.com'
4 );
5
6 $ch = curl_init('https://example.com?' . http_build_query($get));
7 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
8 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
9 curl_setopt($ch, CURLOPT_HEADER, false);
10 $html = curl_exec($ch);
11 curl_close($ch);
12
13 echo $html;
```

PHP

2 POST запрос

```
1 $array = array(
2     'login' => 'admin',
3     'password' => '1234'
4 );
5
6 $ch = curl_init('https://example.com');
7 curl_setopt($ch, CURLOPT_POST, 1);
8 curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($array, '', '&'));
9 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
10 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
11 curl_setopt($ch, CURLOPT_HEADER, false);
12 $html = curl_exec($ch);
13 curl_close($ch);
14
15 echo $html;
```

PHP

Отправка JSON через POST-запрос

ИНСТРУМЕНТЫ

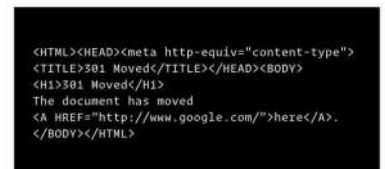
[Разбор JSON](#)

ЕЩЕ НА ЭТУ ТЕМУ

[POST-запрос через file_get_contents\(\)](#)

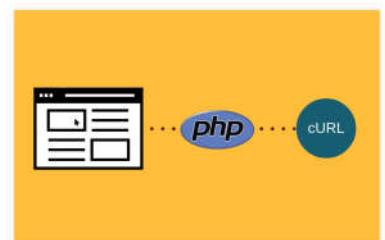
Функция `file_get_contents($filename)` — читает содержимое файла в строку, если в `$filename` указать URL, то функция...

1388 0

[CURL - если сервер отдает редирект](#)

Бывает так что сервер перенаправляет на другой URL. Например Google, если перейти на `https://google.com` с IP из РФ он...

10278 +2

[Отправка формы через PHP CURL](#)

К примеру, на сайте есть форма, после её отправки требуется все данные (`$_GET`, `$_POST` и `$_FILES`) транслировать на...

1500 +1

```
1 $data = array(
2     'name' => 'Маффин',
3     'price' => 100.0
4 );
5
6 $ch = curl_init('https://example.com');
7 curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type:application/json'));
8 curl_setopt($ch, CURLOPT_POST, 1);
9 curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($data, JSON_UNESCAPED_UNICODE));
10 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
11 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
12 curl_setopt($ch, CURLOPT_HEADER, false);
13 $res = curl_exec($ch);
14 curl_close($ch);
15
16 $res = json_encode($res, JSON_UNESCAPED_UNICODE);
17 print_r($res);
```

3 ► PUT запрос

```
1 $ch = curl_init('https://example.com');
2 curl_setopt($ch, CURLOPT_PUT, true);
3 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
4 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
5 curl_setopt($ch, CURLOPT_HEADER, false);
6 $html = curl_exec($ch);
7 curl_close($ch);
8
9 echo $html;
```

PHP

4 ► DELETE запрос

HTTP-метод **DELETE** используется в REST API для удаления объектов.

```
1 $ch = curl_init('https://example.com');
2 curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'DELETE');
3 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
4 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
5 curl_setopt($ch, CURLOPT_HEADER, false);
6 curl_exec($ch);
7 curl_close($ch);
```

PHP

5 ► Запрос через проxy

```
1 $proxy = '165.22.115.179:8080';
2
3 $ch = curl_init('https://example.com');
4 curl_setopt($ch, CURLOPT_TIMEOUT, 400);
5 curl_setopt($ch, CURLOPT_PROXY, $proxy);
6 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
7 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
8 curl_setopt($ch, CURLOPT_HEADER, false);
9 $html = curl_exec($ch);
10 curl_close($ch);
```

PHP

6 ► Отправка файлов на другой сервер

Отправка файлов осуществляется методом **POST**:

До PHP 5.5

```
1 $ch = curl_init('https://example.com');
2 curl_setopt($ch, CURLOPT_POST, 1);
3 curl_setopt($ch, CURLOPT_POSTFIELDS, array('photo' => '@' . __DIR__ . '/image.png'));
4 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
5 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
6 curl_setopt($ch, CURLOPT_HEADER, false);
7 $html = curl_exec($ch);
8 curl_close($ch);
```

PHP

С PHP 5.5 следует применять [CURLFile](http://php.net/manual/ru/function.curl-file-create.php) (<http://php.net/manual/ru/function.curl-file-create.php>).

```
1 $curl_file = curl_file_create(__DIR__ . '/image.png', 'image/png', 'image.png');
2
3 $ch = curl_init('https://example.com');
4 curl_setopt($ch, CURLOPT_POST, 1);
5 curl_setopt($ch, CURLOPT_POSTFIELDS, array('photo' => $curl_file));
6 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
```

PHP

```

7 | curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
8 | curl_setopt($ch, CURLOPT_HEADER, false);
9 | $res = curl_exec($ch);
10 | curl_close($ch);

```

Также через curl можно отправить сразу несколько файлов:

```

1 | $curl_files = array(                                     PHP
2 |     'photo[0]' => curl_file_create(__DIR__ . '/image.png', 'image/png', 'image.png'),
3 |     'photo[1]' => curl_file_create(__DIR__ . '/image-2.png', 'image/png', 'image-2.png')
4 | );
5 |
6 | $ch = curl_init('https://example.com');
7 | curl_setopt($ch, CURLOPT_POST, 1);
8 | curl_setopt($ch, CURLOPT_POSTFIELDS, $curl_files);
9 | curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
10 | curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
11 | curl_setopt($ch, CURLOPT_HEADER, false);
12 | $res = curl_exec($ch);
13 | curl_close($ch);

```

Ещё файлы можно отправить методом **PUT**, например так загружаются файлы в [REST API Яндекс Диска](#).

```

1 | $file = __DIR__ . '/image.jpg';                         PHP
2 | $fp = fopen($file, 'r');
3 |
4 | $ch = curl_init('https://example.com');
5 | curl_setopt($ch, CURLOPT_PUT, true);
6 | curl_setopt($ch, CURLOPT_UPLOAD, true);
7 | curl_setopt($ch, CURLOPT_INFILESIZE, filesize($file));
8 | curl_setopt($ch, CURLOPT_INFILE, $fp);
9 | curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
10 | curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
11 | curl_setopt($ch, CURLOPT_HEADER, false);
12 | curl_exec($ch);
13 | curl_close($ch);

```

7 ► Скачивание файлов

Curl позволяет сохранить результат сразу в файл, указав указатель на открытый файл в параметре **CURLOPT_FILE**.

```

1 | $file_name = __DIR__ . '/file.html';                     PHP
2 | $file = @fopen($file_name, 'w');
3 |
4 | $ch = curl_init('https://example.com');
5 | curl_setopt($ch, CURLOPT_FILE, $file);
6 | curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
7 | curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
8 | curl_setopt($ch, CURLOPT_HEADER, false);
9 | curl_exec($ch);
10 | curl_close($ch);
11 |
12 | fclose($file);

```

Второй вариант:

```

1 | $ch = curl_init('https://example.com');                 PHP
2 | curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
3 | curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
4 | curl_setopt($ch, CURLOPT_HEADER, false);
5 | $html = curl_exec($ch);
6 | curl_close($ch);
7 |
8 | file_put_contents(__DIR__ . '/file.html', $html);

```

8 ► Отправить и получить cookie

Чтобы CURL сохранял куки в файле достаточно прописать его путь в параметрах **CURLOPT_COOKIEFILE** и **CURLOPT_COOKIEJAR**.

```

1 $ch = curl_init('https://example.com');
2 curl_setopt($ch, CURLOPT_COOKIEFILE, __DIR__ . '/cookie.txt');
3 curl_setopt($ch, CURLOPT_COOKIEJAR, __DIR__ . '/cookie.txt');
4 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
5 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
6 curl_setopt($ch, CURLOPT_HEADER, false);
7
8 $html = curl_exec($ch);
9 curl_close($ch);

```

PHP

Передать значение кук можно принудительно через параметр `CURLOPT_COOKIE`.

```

1 $ch = curl_init('https://example.com');
2 curl_setopt($ch, CURLOPT_COOKIE, 'PHPSESSID=61445603b6a0809b061080ed4bb93da3');
3 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
4 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
5 curl_setopt($ch, CURLOPT_HEADER, false);
6
7 $html = curl_exec($ch);
8 curl_close($ch);

```

PHP

9 ▶ Имитация браузера

На многих сайтах есть защита от парсинга. Она основана на том что браузер передает серверу `user agent`, `referer`, `cookie`. Сервер проверяет эти данные и возвращает нормальную страницу. При подключение через `curl` эти данные не передаются и сервер отдает ошибку 404 или 500. Чтобы имитировать браузер нужно добавить заголовки:

```

1 $headers = array(
2     'cache-control: max-age=0',
3     'upgrade-insecure-requests: 1',
4     'user-agent: Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0
5     'sec-fetch-user: ?1',
6     'accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0
7     'x-compress: null',
8     'sec-fetch-site: none',
9     'sec-fetch-mode: navigate',
10    'accept-encoding: deflate, br',
11    'accept-language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7',
12 );
13
14 $ch = curl_init('https://example.com');
15 curl_setopt($ch, CURLOPT_COOKIEFILE, __DIR__ . '/cookie.txt');
16 curl_setopt($ch, CURLOPT_COOKIEJAR, __DIR__ . '/cookie.txt');
17 curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
18 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
19 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
20 curl_setopt($ch, CURLOPT_HEADER, true);
21 $html = curl_exec($ch);
22 curl_close($ch);
23
24 echo $html;

```

PHP

10 ▶ HTTP авторизация

Basic Authorization

Если на сервере настроена [HTTP авторизация](#), например с помощью `.htpasswd`, подключится к нему можно с помощью параметра `CURLOPT_USERPWD`.

```

1 $ch = curl_init('https://example.com');
2 curl_setopt($ch, CURLOPT_USERPWD, 'login:password');
3 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
4 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
5 curl_setopt($ch, CURLOPT_HEADER, false);
6 $html = curl_exec($ch);
7 curl_close($ch);
8
9 echo $html;

```

PHP

OAuth авторизация

```

1 $ch = curl_init('https://example.com');
2 curl_setopt($ch, CURLOPT_HTTPHEADER, array('Authorization: OAuth TOKEN'));
3 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
4 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
5 curl_setopt($ch, CURLOPT_HEADER, false);
6 $html = curl_exec($ch);
7 curl_close($ch);
8

```

PHP

```
9 | echo $html;
```

11 ► Получить HTTP код ответа сервера

```
1 | $ch = curl_init('https://example.com');  
2 | curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);  
3 | curl_setopt($ch, CURLOPT_HEADER, false);  
4 | curl_exec($ch);  
5 | $http_code = curl_getinfo($ch, CURLINFO_HTTP_CODE);  
6 | curl_close($ch);  
7 |  
8 | echo $http_code; // Выведет: 200
```

PHP

12 ► Если CURL возвращает false

Какая произошла ошибка можно узнать с помощью функции `curl_errno()`.

```
1 | $ch = curl_init('https://example.com');  
2 | curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);  
3 | curl_setopt($ch, CURLOPT_HEADER, false);  
4 | curl_exec($ch);  
5 | $res = curl_exec($ch);  
6 |  
7 | var_dump($res); // false  
8 |  
9 | if ($errno = curl_errno($ch)) {  
10 |     $message = curl_strerror($errno);  
11 |     echo "cURL error ({$errno}): \n {$message}"; // Выведет: cURL error (35): SSL connect error  
12 | }  
13 |  
14 | curl_close($ch);
```

PHP

📅 14.11.2017, обновлено 08.04.2022 👁 136954

Комментарии 1



Andrey Jelovovskii
21 апреля 2022 в 22:13

^ 0 v

Всем спасибо! Все в одном месте. Очень помогло, спасибо!

Ответить



[Авторизуйтесь](#), чтобы добавить комментарий.