HttpClient 4 - Отправить Custom Cookie

* [HTTP Client](https://www.codeflow.site/ru/tag/HTTP%20Client)

**1. Обзор**

Этот учебник будет посвящен **тому, как отправить пользовательский файл cookie с помощью Apache HttpClient 4**.

Если вы хотите копнуть глубже и узнать другие интересные вещи, которые вы можете сделать с помощью HttpClient, перейдите по ссылке [**основное руководство по HttpClient**](https://www.codeflow.site/httpclient-guide).

**2. Настроить управление файлами cookie на HttpClient**

**2.1. HttpClient после 4.3**

В более новом HttpClient 4.3 мы будем использовать API-интерфейс Fluent Builder, отвечающий за создание и настройку клиента.

Во-первых, нам нужно создать хранилище cookie и настроить наш пример cookie в магазине:

BasicCookieStore cookieStore = **new** BasicCookieStore();

BasicClientCookie cookie = **new** BasicClientCookie("JSESSIONID", "1234");

cookie.setDomain(".github.com");

cookie.setPath("/");

cookieStore.addCookie(cookie);

Затем мы можем настроить это хранилище cookie на HttpClient и отправить запрос:

@Test

**public** **void** **whenSettingCookiesOnTheHttpClient\_\_thenCookieSentCorrectly**()

**throws** ClientProtocolException, IOException {

BasicCookieStore cookieStore = **new** BasicCookieStore();

BasicClientCookie cookie = **new** BasicClientCookie("JSESSIONID", "1234");

cookie.setDomain(".github.com");

cookie.setPath("/");

cookieStore.addCookie(cookie);

HttpClient client = HttpClientBuilder.create().setDefaultCookieStore(cookieStore).build();

**final** HttpGet request = **new** HttpGet("http://www.github.com");

response = client.execute(request);

assertThat(response.getStatusLine().getStatusCode(), equalTo(200));

}

Очень важным элементом является domain **\_\_, который устанавливается в cookie -**без установки правильного домена клиент вообще не будет отправлять cookie \*\* !

**2.2. HttpClient до 4.3**

В более старых версиях HttpClient (до 4.3) хранилище файлов cookie было установлено непосредственно в *HttpClient*

@Test

**public** **void** **givenUsingDeprecatedApi\_\_whenSettingCookiesOnTheHttpClient\_\_thenCorrect**()

**throws** ClientProtocolException, IOException {

BasicCookieStore cookieStore = **new** BasicCookieStore();

BasicClientCookie cookie = **new** BasicClientCookie("JSESSIONID", "1234");

cookie.setDomain(".github.com");

cookie.setPath("/");

cookieStore.addCookie(cookie);

DefaultHttpClient client = **new** DefaultHttpClient();

client.setCookieStore(cookieStore);

HttpGet request = **new** HttpGet("http://www.github.com");

response = client.execute(request);

assertThat(response.getStatusLine().getStatusCode(), equalTo(200));

}

Кроме способа построения клиента, нет никаких отличий от предыдущего примера.

**3. Установить Cookie по запросу**

Если настройка cookie для всего HttpClient невозможна, мы можем настроить запросы с помощью cookie отдельно:

@Test

**public** **void** **whenSettingCookiesOnTheRequest\_\_thenCookieSentCorrectly**()

**throws** ClientProtocolException, IOException {

BasicCookieStore cookieStore = **new** BasicCookieStore();

BasicClientCookie cookie = **new** BasicClientCookie("JSESSIONID", "1234");

cookie.setDomain(".github.com");

cookie.setPath("/");

cookieStore.addCookie(cookie);

instance = HttpClientBuilder.create().build();

HttpGet request = **new** HttpGet("http://www.github.com");

HttpContext localContext = **new** BasicHttpContext();

localContext.setAttribute(HttpClientContext.COOKIE\_\_STORE, cookieStore);

//localContext.setAttribute(ClientContext.COOKIE\_\_STORE, cookieStore);//before 4.3

response = instance.execute(request, localContext);

assertThat(response.getStatusLine().getStatusCode(), equalTo(200));

}

**4. Установите Cookie на запрос низкого уровня**

Низкоуровневой альтернативой установки cookie в HTTP-запросе будет установка его в виде необработанного заголовка:

@Test

**public** **void** **whenSettingCookiesOnARequest\_\_thenCorrect**()

**throws** ClientProtocolException, IOException {

instance = HttpClientBuilder.create().build();

HttpGet request = **new** HttpGet("http://www.github.com");

request.setHeader("Cookie", "JSESSIONID=1234");

response = instance.execute(request);

assertThat(response.getStatusLine().getStatusCode(), equalTo(200));

}

Это, конечно, намного более **подвержено ошибкам, чем работа со встроенной поддержкой cookie**- например, обратите внимание, что в этом случае мы больше не задаем домен - что неверно.

**5. Заключение**

В этой статье показано, как **работать с HttpClient для отправки пользовательских cookie-файлов, управляемых пользователем**.

Обратите внимание, что это не то же самое, что позволить HttpClient иметь дело с файлами cookie, установленными сервером, но вместо этого он контролирует клиентскую сторону вручную на низком уровне.

Реализация всех этих примеров и фрагментов кода **может быть найдена в [my[my project github](https://github.com/eugenp/tutorials/tree/master/httpclient" \l "readme)**- это проект на основе Eclipse, поэтому он должен легко импортировать и запускать как есть.