



# Exercise 1: HTTP

## Bruno Berndt Lima - 12542550

*Upload a txt or pdf file named yourName.USPnumber, with the answers for the questions.*

These exercises are about understanding how to make different kinds of HTTP. To keep it simple, we're going to make a lot of our requests to the same website: <https://httpbin.org>. httpbin is a service that accepts HTTP requests and then tells you what request you made. You may use [postman](#), [insomnia](#), [cURL](#), or other tools to make the requests. Using a text editor, include the request and response headers and the body in the answers to each question (if using postman, use the *console view*). Remember that some questions also need textual answers.

### 1. Request <https://httpbin.org>

```
> GET / HTTP/2
> Host: httpbin.org
> user-agent: insomnia/2023.1.0
> accept: */*
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
< HTTP/2 200
< date: Mon, 27 Mar 2023 21:32:37 GMT
< content-type: text/html; charset=utf-8
< content-length: 9593
< server: gunicorn/19.9.0
< access-control-allow-origin: *
< access-control-allow-credentials: true
```

### 2. Request <https://httpbin.org/anything>. [httpbin.org/anything](https://httpbin.org/anything) will look at the request you made, parse it, and echo back to you what you requested.

```
> GET /anything HTTP/2
> Host: httpbin.org
> user-agent: insomnia/2023.1.0
> accept: */*
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
< HTTP/2 200
< date: Mon, 27 Mar 2023 21:52:07 GMT
< content-type: application/json
< content-length: 349
< server: gunicorn/19.9.0
< access-control-allow-origin: *
```



< access-control-allow-credentials: true

```
{
  "args": {},
  "data": "",
  "files": {},
  "form": {},
  "headers": {
    "Accept": "*/*",
    "Host": "httpbin.org",
    "User-Agent": "insomnia/2023.1.0",
    "X-Amzn-Trace-Id": "Root=1-64221007-3f40287519a76ba000b850dc"
  },
  "json": null,
  "method": "GET",
  "origin": "189.15.227.53",
  "url": "https://httpbin.org/anything"
}
```

### 3. Make a POST request to <https://httpbin.org/anything>

```
> POST /anything HTTP/2
> Host: httpbin.org
> user-agent: insomnia/2023.1.0
> accept: */*
> content-length: 0
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
< HTTP/2 200
< date: Mon, 27 Mar 2023 21:59:42 GMT
< content-type: application/json
< content-length: 378
< server: gunicorn/19.9.0
< access-control-allow-origin: *
< access-control-allow-credentials: true
{
  "args": {},
  "data": "",
  "files": {},
  "form": {},
  "headers": {
    "Accept": "*/*",
    "Content-Length": "0",
    "Host": "httpbin.org",
    "User-Agent": "insomnia/2023.1.0",
    "X-Amzn-Trace-Id": "Root=1-642211ce-6aa93641444d59911c1bbbe6"
  },
  "json": null,
  "method": "POST",
  "origin": "189.15.227.53",
```



```
"url": "https://httpbin.org/anything"  
}
```

4. **Make a GET request to <https://httpbin.org/anything>, but this time add some query parameters (set value=panda).**

```
> GET /anything HTTP/2  
> Host: httpbin.org  
> user-agent: insomnia/2023.1.0  
> accept: */*
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
< HTTP/2 200  
< date: Mon, 27 Mar 2023 22:03:17 GMT  
< content-type: application/json  
< content-length: 349  
< server: gunicorn/19.9.0  
< access-control-allow-origin: *  
< access-control-allow-credentials: true
```

```
{  
  "args": {},  
  "data": "",  
  "files": {},  
  "form": {},  
  "headers": {  
    "Accept": "*/*",  
    "Host": "httpbin.org",  
    "User-Agent": "insomnia/2023.1.0",  
    "X-Amzn-Trace-Id": "Root=1-642212a5-56393c4927eda7cb6b471c48"  
  },  
  "json": null,  
  "method": "GET",  
  "origin": "189.15.227.53",  
  "url": "https://httpbin.org/anything"  
}
```

5. **Request google's robots.txt file ([www.google.com/robots.txt](http://www.google.com/robots.txt))**

```
> GET /robots.txt HTTP/1.1  
> Host: www.google.com  
> User-Agent: insomnia/2023.1.0  
> Accept: */*
```

\* Mark bundle as not supporting multiuse

```
< HTTP/1.1 200 OK  
< Accept-Ranges: bytes  
< Vary: Accept-Encoding
```



< Content-Type: text/plain  
< Cross-Origin-Resource-Policy: cross-origin  
< Cross-Origin-Opener-Policy-Report-Only: same-origin; report-to="static-on-bigtable"  
< Report-To: {"group":"static-on-bigtable","max\_age":2592000,"endpoints":[{"url":"https://csp.withgoogle.com/csp/report-to/static-on-bigtable"}] }  
< Content-Length: 7240  
< Date: Mon, 27 Mar 2023 22:10:36 GMT  
< Expires: Mon, 27 Mar 2023 22:10:36 GMT  
< Cache-Control: private, max-age=0  
< Last-Modified: Wed, 13 Jul 2022 19:00:00 GMT  
< X-Content-Type-Options: nosniff  
< Server: sffe  
< X-XSS-Protection: 0

User-agent: \*  
Disallow: /search  
Allow: /search/about  
Allow: /search/static  
Allow: /search/howsearchworks  
Disallow: /sdch  
Disallow: /groups  
Disallow: /index.html?  
Disallow: /?  
Allow: /?hl=  
Disallow: /?hl=\*&  
Allow: /?hl=\*&gws\_rd=ssl\$  
Disallow: /?hl=\*&\*&gws\_rd=ssl  
Allow: /?gws\_rd=ssl\$  
Allow: /?pt1=true\$  
Disallow: /imgres  
Disallow: /u/  
Disallow: /preferences  
Disallow: /setprefs  
Disallow: /default  
Disallow: /m?  
Disallow: /m/  
Allow: /m/finance  
Disallow: /wml?  
Disallow: /wml/?  
Disallow: /wml/search?  
Disallow: /xhtml?  
Disallow: /xhtml/?  
Disallow: /xhtml/search?  
Disallow: /xml?  
Disallow: /imode?  
Disallow: /imode/?  
Disallow: /imode/search?  
Disallow: /jsky?  
Disallow: /jsky/?  
Disallow: /jsky/search?  
Disallow: /pda?



Disallow: /pda/?  
Disallow: /pda/search?  
Disallow: /sprint\_xhtml  
Disallow: /sprint\_wml  
Disallow: /pqa  
Disallow: /palm  
Disallow: /gwt/  
Disallow: /purchases  
Disallow: /local?  
Disallow: /local\_url  
Disallow: /shihui?  
Disallow: /shihui/  
Disallow: /products?  
Disallow: /product\_  
Disallow: /products\_  
Disallow: /products;  
Disallow: /print  
Disallow: /books/  
Disallow: /bkshp?\*q=\*  
Disallow: /books?\*q=\*  
Disallow: /books?\*output=\*  
Disallow: /books?\*pg=\*  
Disallow: /books?\*jtp=\*  
Disallow: /books?\*jscmd=\*  
Disallow: /books?\*buy=\*  
Disallow: /books?\*zoom=\*  
Allow: /books?\*q=related:\*  
Allow: /books?\*q=editions:\*  
Allow: /books?\*q=subject:\*  
Allow: /books/about  
Allow: /booksrightsholders  
Allow: /books?\*zoom=1\*  
Allow: /books?\*zoom=5\*  
Allow: /books/content?\*zoom=1\*  
Allow: /books/content?\*zoom=5\*  
Disallow: /ebooks/  
Disallow: /ebooks?\*q=\*  
Disallow: /ebooks?\*output=\*  
Disallow: /ebooks?\*pg=\*  
Disallow: /ebooks?\*jscmd=\*  
Disallow: /ebooks?\*buy=\*  
Disallow: /ebooks?\*zoom=\*  
Allow: /ebooks?\*q=related:\*  
Allow: /ebooks?\*q=editions:\*  
Allow: /ebooks?\*q=subject:\*  
Allow: /ebooks?\*zoom=1\*  
Allow: /ebooks?\*zoom=5\*  
Disallow: /patents?  
Disallow: /patents/download/  
Disallow: /patents/pdf/  
Disallow: /patents/related/



Disallow: /scholar  
Disallow: /citations?  
Allow: /citations?user=  
Disallow: /citations?\*cstart=  
Allow: /citations?view\_op=new\_profile  
Allow: /citations?view\_op=top\_venues  
Allow: /scholar\_share  
Disallow: /s?  
Allow: /maps?\*output=classic\*  
Allow: /maps?\*file=  
Allow: /maps/d/  
Disallow: /maps?  
Disallow: /mapstt?  
Disallow: /mapslt?  
Disallow: /maps/stk/  
Disallow: /maps/br?  
Disallow: /mapabcpoi?  
Disallow: /maphp?  
Disallow: /mapprint?  
Disallow: /maps/api/js/  
Allow: /maps/api/js  
Disallow: /maps/api/place/js/  
Disallow: /maps/api/staticmap  
Disallow: /maps/api/streetview  
Disallow: /maps/\_/sw/manifest.json  
Disallow: /mld?  
Disallow: /staticmap?  
Disallow: /maps/preview  
Disallow: /maps/place  
Disallow: /maps/timeline/  
Disallow: /help/maps/streetview/partners/welcome/  
Disallow: /help/maps/indoormaps/partners/  
Disallow: /lochp?  
Disallow: /center  
Disallow: /ie?  
Disallow: /blogsearch/  
Disallow: /blogsearch\_feeds  
Disallow: /advanced\_blog\_search  
Disallow: /uds/  
Disallow: /chart?  
Disallow: /transit?  
Allow: /calendar\$  
Allow: /calendar/about/  
Disallow: /calendar/  
Disallow: /cl2/feeds/  
Disallow: /cl2/ical/  
Disallow: /coop/directory  
Disallow: /coop/manage  
Disallow: /trends?  
Disallow: /trends/music?  
Disallow: /trends/hottrends?



Disallow: /trends/viz?  
Disallow: /trends/embed.js?  
Disallow: /trends/fetchComponent?  
Disallow: /trends/beta  
Disallow: /trends/topics  
Disallow: /musica  
Disallow: /musicad  
Disallow: /musicas  
Disallow: /musicl  
Disallow: /musics  
Disallow: /musicsearch  
Disallow: /musicsp  
Disallow: /musiclp  
Disallow: /urchin\_test/  
Disallow: /movies?  
Disallow: /wapsearch?  
Allow: /safebrowsing/diagnostic  
Allow: /safebrowsing/report\_badware/  
Allow: /safebrowsing/report\_error/  
Allow: /safebrowsing/report\_phish/  
Disallow: /reviews/search?  
Disallow: /orkut/albums  
Disallow: /cbk  
Disallow: /recharge/dashboard/car  
Disallow: /recharge/dashboard/static/  
Disallow: /profiles/me  
Allow: /profiles  
Disallow: /s2/profiles/me  
Allow: /s2/profiles  
Allow: /s2/oz  
Allow: /s2/photos  
Allow: /s2/search/social  
Allow: /s2/static  
Disallow: /s2  
Disallow: /transconsole/portal/  
Disallow: /gcc/  
Disallow: /aclk  
Disallow: /cse?  
Disallow: /cse/home  
Disallow: /cse/panel  
Disallow: /cse/manage  
Disallow: /tbproxy/  
Disallow: /imesync/  
Disallow: /shenghuo/search?  
Disallow: /support/forum/search?  
Disallow: /reviews/polls/  
Disallow: /hosted/images/  
Disallow: /ppob/?  
Disallow: /ppob?  
Disallow: /accounts/ClientLogin  
Disallow: /accounts/ClientAuth



Disallow: /accounts/o8  
Allow: /accounts/o8/id  
Disallow: /topicsearch?q=  
Disallow: /xfx7/  
Disallow: /squared/api  
Disallow: /squared/search  
Disallow: /squared/table  
Disallow: /qnasearch?  
Disallow: /app/updates  
Disallow: /sidewiki/entry/  
Disallow: /quality\_form?  
Disallow: /labs/popgadget/search  
Disallow: /buzz/post  
Disallow: /compressiontest/  
Disallow: /analytics/feeds/  
Disallow: /analytics/partners/comments/  
Disallow: /analytics/portal/  
Disallow: /analytics/uploads/  
Allow: /alerts/manage  
Allow: /alerts/remove  
Disallow: /alerts/  
Allow: /alerts/\$  
Disallow: /ads/search?  
Disallow: /ads/plan/action\_plan?  
Disallow: /ads/plan/api/  
Disallow: /ads/hotels/partners  
Disallow: /phone/compare/?  
Disallow: /travel/clk  
Disallow: /hotelfinder/rpc  
Disallow: /hotels/rpc  
Disallow: /commercesearch/services/  
Disallow: /evaluation/  
Disallow: /chrome/browser/mobile/tour  
Disallow: /compare/\*/apply\*  
Disallow: /forms/perks/  
Disallow: /shopping/suppliers/search  
Disallow: /ct/  
Disallow: /edu/cs4hs/  
Disallow: /trustedstores/s/  
Disallow: /trustedstores/tm2  
Disallow: /trustedstores/verify  
Disallow: /adwords/proposal  
Disallow: /shopping?\*  
Disallow: /shopping/product/  
Disallow: /shopping/seller  
Disallow: /shopping/ratings/account/metrics  
Disallow: /shopping/ratings/merchant/immersivedetails  
Disallow: /shopping/reviewer  
Disallow: /about/careers/applications/  
Disallow: /about/careers/applications-a/  
Disallow: /landing/signout.html





Disallow: /webmasters/sitemaps/ping?  
Disallow: /ping?  
Disallow: /gallery/  
Disallow: /landing/now/ontap/  
Allow: /searchhistory/  
Allow: /maps/reserve  
Allow: /maps/reserve/partners  
Disallow: /maps/reserve/api/  
Disallow: /maps/reserve/search  
Disallow: /maps/reserve/bookings  
Disallow: /maps/reserve/settings  
Disallow: /maps/reserve/manage  
Disallow: /maps/reserve/payment  
Disallow: /maps/reserve/receipt  
Disallow: /maps/reserve/sellersignup  
Disallow: /maps/reserve/payments  
Disallow: /maps/reserve/feedback  
Disallow: /maps/reserve/terms  
Disallow: /maps/reserve/m/  
Disallow: /maps/reserve/b/  
Disallow: /maps/reserve/partner-dashboard  
Disallow: /about/views/  
Disallow: /intl/\*/about/views/  
Disallow: /local/cars  
Disallow: /local/cars/  
Disallow: /local/dealership/  
Disallow: /local/dining/  
Disallow: /local/place/products/  
Disallow: /local/place/reviews/  
Disallow: /local/place/rap/  
Disallow: /local/tab/  
Disallow: /localservices/\*  
Allow: /finance  
Allow: /js/  
Disallow: /nonprofits/account/  
Disallow: /fbx  
Disallow: /viewer

#### # AdsBot

User-agent: AdsBot-Google  
Disallow: /maps/api/js/  
Allow: /maps/api/js  
Disallow: /maps/api/place/js/  
Disallow: /maps/api/staticmap  
Disallow: /maps/api/streetview

# Crawlers of certain social media sites are allowed to access page markup when  
google.com/imgres\* links are shared. To learn more, please contact images-robots-  
allowlist@google.com.

User-agent: Twitterbot  
Allow: /imgres



User-agent: facebookexternalhit  
Allow: /imgres

Sitemap: <https://www.google.com/sitemap.xml>

**6. Make a GET request to <https://httpbin.org/anything> and set the header User-Agent: elephant.**

```
> GET /anything HTTP/2  
> Host: httpbin.org  
> user-agent: elephant;  
> accept: */*
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
< HTTP/2 200  
< date: Mon, 27 Mar 2023 22:18:18 GMT  
< content-type: application/json  
< content-length: 341  
< server: gunicorn/19.9.0  
< access-control-allow-origin: *  
< access-control-allow-credentials: true
```

```
{  
  "args": {},  
  "data": "",  
  "files": {},  
  "form": {},  
  "headers": {  
    "Accept": "*/*",  
    "Host": "httpbin.org",  
    "User-Agent": "elephant;",  
    "X-Amzn-Trace-Id": "Root=1-6422162a-28d9c4712d928ce540075bdf"  
  },  
  "json": null,  
  "method": "GET",  
  "origin": "189.15.227.53",  
  "url": "https://httpbin.org/anything"  
}
```

**7. Make a DELETE request to <https://httpbin.org/anything>**

```
> DELETE /anything HTTP/2  
> Host: httpbin.org  
> user-agent: insomnia/2023.1.0  
> accept: */*
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!



```
< HTTP/2 200
< date: Mon, 27 Mar 2023 22:19:25 GMT
< content-type: application/json
< content-length: 352
< server: gunicorn/19.9.0
< access-control-allow-origin: *
< access-control-allow-credentials: true

{
  "args": {},
  "data": "",
  "files": {},
  "form": {},
  "headers": {
    "Accept": "*/*",
    "Host": "httpbin.org",
    "User-Agent": "insomnia/2023.1.0",
    "X-Amzn-Trace-Id": "Root=1-6422166d-6af8c43b22e7aaa55b42df0b"
  },
  "json": null,
  "method": "DELETE",
  "origin": "189.15.227.53",
  "url": "https://httpbin.org/anything"
}
```

8. Make a POST request to <https://httpbin.org/anything> with the JSON body `{"value": "panda"}`. The Content-Type header must be different from `application/json`, some tools will fill this header automatically.

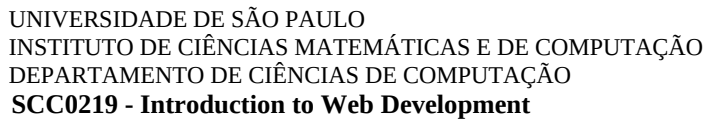
```
> POST /anything HTTP/2
> Host: httpbin.org
> user-agent: insomnia/2023.1.0
> accept: */*
> content-length: 18
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
| {"value": "panda"}
```

\* We are completely uploaded and fine

```
< HTTP/2 200
< date: Mon, 27 Mar 2023 22:21:59 GMT
< content-type: application/json
< content-length: 423
< server: gunicorn/19.9.0
< access-control-allow-origin: *
< access-control-allow-credentials: true
```



9. Make the same POST request as the previous exercise, but set the Content-Type header to `application/json` (because POST requests need to have a content type that matches their body). Look at the `json` field in the response to see the difference from the previous one.

```
{
  "args": {},
  "data": "{\n\t\t\"value\": \"panda\"\n\t}",
  "files": {},
```



```
"form": {},  
"headers": {  
  "Accept": "*/*",  
  "Content-Length": "23",  
  "Content-Type": "application/json",  
  "Host": "httpbin.org",  
  "User-Agent": "insomnia/2023.1.0",  
  "X-Amzn-Trace-Id": "Root=1-64221768-71c471d816a6e3b32b929df4"  
},  
"json": {  
  "value": "panda"  
},  
"method": "POST",  
"origin": "189.15.227.53",  
"url": "https://httpbin.org/anything"  
}
```

#### 10. Make a GET request to <https://httpbin.org/anything> and set the header Accept-Encoding: gzip (What happens? Why?)

```
> GET /anything HTTP/2  
> Host: httpbin.org  
> user-agent: insomnia/2023.1.0  
> accept-encoding: gzip;  
> accept: */*
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
< HTTP/2 200  
< date: Mon, 27 Mar 2023 22:25:53 GMT  
< content-type: application/json  
< content-length: 382  
< server: gunicorn/19.9.0  
< access-control-allow-origin: *  
< access-control-allow-credentials: true
```

```
{  
  "args": {},  
  "data": "",  
  "files": {},  
  "form": {},  
  "headers": {  
    "Accept": "*/*",  
    "Accept-Encoding": "gzip",  
    "Host": "httpbin.org",  
    "User-Agent": "insomnia/2023.1.0",  
    "X-Amzn-Trace-Id": "Root=1-642217f1-3202e31658882ef116100051"  
  },  
  "json": null,  
  "method": "GET",  
  "origin": "189.15.227.53",  
}
```



```
"url": "https://httpbin.org/anything"  
}
```

**11. Put a bunch of a JSON in a file and then make a POST request to <https://httpbin.org/anything> with the JSON in that file as the body**

```
> POST /anything HTTP/2  
> Host: httpbin.org  
> user-agent: insomnia/2023.1.0  
> content-type: application/json  
> accept: */*  
> content-length: 602
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
| {"widget": {  
|   "debug": "on",  
|   "window": {  
|     "title": "Sample Konfabulator Widget",  
|     "name": "main_window",  
|     "width": 500,  
|     "height": 500  
|   },  
|   "image": {  
|     "src": "Images/Sun.png",  
|     "name": "sun1",  
|     "hOffset": 250,  
|     "vOffset": 250,  
|     "alignment": "center"  
|   },  
|   "text": {  
|     "data": "Click Here",  
|     "size": 36,  
|     "style": "bold",  
|     "name": "text1",  
|     "hOffset": 250,  
|     "vOffset": 100,  
|     "alignment": "center",  
|     "onMouseUp": "sun1.opacity = (sun1.opacity / 100) * 90;"  
|   }  
| }}  
| }}
```

\* We are completely uploaded and fine

```
< HTTP/2 200  
< date: Mon, 27 Mar 2023 22:37:25 GMT  
< content-type: application/json  
< content-length: 1753  
< server: gunicorn/19.9.0
```



< access-control-allow-origin: \*

< access-control-allow-credentials: true

```
{
  "args": {},
  "data": "{\n  \"widget\": {\n    \"debug\": \"on\",\n    \"window\": {\n      \"title\": \"Sample Konfabulator Widget\",\n      \"name\": \"main_window\",\n      \"width\": 500,\n      \"height\": 500\n    },\n    \"image\": {\n      \"src\": \"Images/Sun.png\",\n      \"name\": \"sun1\",\n      \"hOffset\": 250,\n      \"vOffset\": 250,\n      \"alignment\": \"center\",\n      \"text\": {\n        \"data\": \"Click Here\",\n        \"size\": 36,\n        \"style\": \"bold\",\n        \"name\": \"text1\",\n        \"hOffset\": 250,\n        \"vOffset\": 100,\n        \"alignment\": \"center\",\n        \"onMouseUp\": \"sun1.opacity = (sun1.opacity / 100) * 90;\"\n      }\n    }\n  },\n  \"files\": {},\n  \"form\": {},\n  \"headers\": {\n    \"Accept\": \"*/*\",\n    \"Content-Length\": \"602\",\n    \"Content-Type\": \"application/json\",\n    \"Host\": \"httpbin.org\",\n    \"User-Agent\": \"insomnia/2023.1.0\",\n    \"X-Amzn-Trace-Id\": \"Root=1-64221aa5-2cf9a6cd3ceec7372c5295d6\"\n  },\n  \"json\": {\n    \"widget\": {\n      \"debug\": \"on\",\n      \"image\": {\n        \"alignment\": \"center\",\n        \"hOffset\": 250,\n        \"name\": \"sun1\",\n        \"src\": \"Images/Sun.png\",\n        \"vOffset\": 250\n      },\n      \"text\": {\n        \"alignment\": \"center\",\n        \"data\": \"Click Here\",\n        \"hOffset\": 250,\n        \"name\": \"text1\",\n        \"onMouseUp\": \"sun1.opacity = (sun1.opacity / 100) * 90;\",\n        \"size\": 36,\n        \"style\": \"bold\",\n        \"vOffset\": 100\n      },\n      \"window\": {\n        \"height\": 500,\n        \"name\": \"main_window\",\n        \"title\": \"Sample Konfabulator Widget\",\n        \"width\": 500\n      }\n    }\n  }\n},
```



```
"method": "POST",  
"origin": "189.15.227.53",  
"url": "https://httpbin.org/anything"  
}
```

## 12. Make a PUT request to <https://httpbin.org/anything>

```
> PUT /anything HTTP/2  
> Host: httpbin.org  
> user-agent: insomnia/2023.1.0  
> accept: */*  
> content-length: 0
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
< HTTP/2 200  
< date: Mon, 27 Mar 2023 22:30:04 GMT  
< content-type: application/json  
< content-length: 377  
< server: gunicorn/19.9.0  
< access-control-allow-origin: *  
< access-control-allow-credentials: true
```

```
{  
  "args": {},  
  "data": "",  
  "files": {},  
  "form": {},  
  "headers": {  
    "Accept": "*/*",  
    "Content-Length": "0",  
    "Host": "httpbin.org",  
    "User-Agent": "insomnia/2023.1.0",  
    "X-Amzn-Trace-Id": "Root=1-642218ec-595c843a34457ba46b3e9e92"  
  },  
  "json": null,  
  "method": "PUT",  
  "origin": "189.15.227.53",  
  "url": "https://httpbin.org/anything"  
}
```

## 13. Request <https://www.twitter.com>. Using your tool (postman, cURL, etc), show the response headers. Is the HTML returned the same you get in a browser? Why?

```
> GET / HTTP/2  
> Host: www.twitter.com  
> user-agent: insomnia/2023.1.0  
> accept: */*
```

\* TLSv1.3 (IN), TLS handshake, Newsession Ticket (4):





- \* TLSv1.3 (IN), TLS handshake, Newsession Ticket (4):
- \* old SSL session ID is stale, removing

< HTTP/2 301  
< perf: 7626143928  
< location: https://twitter.com/  
< cache-control: no-cache, no-store, max-age=0  
< content-length: 0  
< x-transaction-id: a5da0f26c45ea74a  
< x-response-time: 120  
< x-connection-hash: 2519916eb2e8b3b96c513880a80cfce350df68aediae2dd6821a2ba4ef72cb928  
< date: Mon, 27 Mar 2023 22:31:16 GMT  
< server: tsa\_d

- \* Connection #2 to host www.twitter.com left intact
- \* Issue another request to this URL: 'https://twitter.com/'
- \* Trying 104.244.42.129:443...
- \* Connected to twitter.com (104.244.42.129) port 443 (#3)
- \* ALPN, offering h2
- \* ALPN, offering http/1.1
- \* TLSv1.3 (OUT), TLS handshake, Client hello (1):
- \* TLSv1.3 (IN), TLS handshake, Server hello (2):
- \* TLSv1.3 (IN), TLS handshake, Encrypted Extensions (8):
- \* TLSv1.3 (IN), TLS handshake, Certificate (11):
- \* TLSv1.3 (IN), TLS handshake, CERT verify (15):
- \* TLSv1.3 (IN), TLS handshake, Finished (20):
- \* TLSv1.3 (OUT), TLS change cipher, Change cipher spec (1):
- \* TLSv1.3 (OUT), TLS handshake, Finished (20):
- \* SSL connection using TLSv1.3 / TLS\_AES\_256\_GCM\_SHA384
- \* ALPN, server accepted to use h2
- \* Server certificate:
- \* subject: C=US; ST=California; L=San Francisco; O=Twitter, Inc.; CN=twitter.com
- \* start date: Dec 10 00:00:00 2022 GMT
- \* expire date: Dec 10 23:59:59 2023 GMT
- \* subjectAltName: host "twitter.com" matched cert's "twitter.com"
- \* issuer: C=US; O=DigiCert Inc; CN=DigiCert TLS RSA SHA256 2020 CA1
- \* SSL certificate verify ok.
- \* Using HTTP2, server supports multiplexing
- \* Connection state changed (HTTP/2 confirmed)
- \* Copying HTTP/2 data in stream buffer to connection buffer after upgrade: len=0
- \* Using Stream ID: 1 (easy handle 0x8f00463e000)

> GET / HTTP/2  
> Host: twitter.com  
> user-agent: insomnia/2023.1.0  
> accept: \*/\*

- \* TLSv1.3 (IN), TLS handshake, Newsession Ticket (4):
- \* TLSv1.3 (IN), TLS handshake, Newsession Ticket (4):
- \* old SSL session ID is stale, removing



```
< HTTP/2 200
< date: Mon, 27 Mar 2023 22:31:17 GMT
< perf: 7626143928
< expiry: Tue, 31 Mar 1981 05:00:00 GMT
< pragma: no-cache
< server: tsa_d
< set-cookie: guest_id_ads=v1%3A167995627712637747; Max-Age=63072000; Expires=Wed,
    26 Mar 2025 22:31:17 GMT; Path=/; Domain=.twitter.com; Secure; SameSite=None
< set-cookie: guest_id_marketing=v1%3A167995627712637747; Max-Age=63072000;
    Expires=Wed, 26 Mar 2025 22:31:17 GMT; Path=/; Domain=.twitter.com; Secure;
    SameSite=None
< set-cookie: gt=1640481643545673728; Max-Age=10800; Expires=Tue, 28 Mar 2023 01:31:17
    GMT; Path=/; Domain=.twitter.com; Secure
< set-cookie: personalization_id="v1_wMI2WP4ws0Ary1v8rhKCzg=="; Max-Age=63072000;
    Expires=Wed, 26 Mar 2025 22:31:17 GMT; Path=/; Domain=.twitter.com; Secure;
    SameSite=None
< set-cookie: guest_id=v1%3A167995627712637747; Max-Age=63072000; Expires=Wed, 26
    Mar 2025 22:31:17 GMT; Path=/; Domain=.twitter.com; Secure; SameSite=None
< set-cookie: ct0=; Max-Age=-1679956276; Expires=Thu, 01 Jan 1970 00:00:01 GMT; Path=/;
    Domain=.twitter.com; Secure; SameSite=Lax
< content-type: text/html; charset=utf-8
< x-powered-by: Express
< cache-control: no-cache, no-store, must-revalidate, pre-check=0, post-check=0
< last-modified: Mon, 27 Mar 2023 22:31:17 GMT
< x-frame-options: DENY
< x-transaction-id: 2d9ca12deddd20b5
< x-xss-protection: 0
< x-content-type-options: nosniff
< content-security-policy: connect-src 'self' blob: https://*.pscp.tv https://*.video.pscp.tv
    https://*.twimg.com https://api.twitter.com https://api-stream.twitter.com https://ads-
    api.twitter.com https://aa.twitter.com https://caps.twitter.com https://pay.twitter.com
    https://sentry.io https://ton.twitter.com https://twitter.com https://upload.twitter.com
    https://www.google-analytics.com https://accounts.google.com/gsi/status
    https://accounts.google.com/gsi/log https://app.link https://api2.branch.io https://bnc.lt
    wss://*.pscp.tv https://vmap.snappytv.com https://vmapstage.snappytv.com
    https://vmaprel.snappytv.com https://vmap.grabyo.com https://dhdsnappytv-vh.akamaihd.net
    https://pdhdsnappytv-vh.akamaihd.net https://mdhdsnappytv-vh.akamaihd.net
    https://mdhdsnappytv-vh.akamaihd.net https://mpdhdsnappytv-vh.akamaihd.net
    https://mmdhdsnappytv-vh.akamaihd.net https://mdhdsnappytv-vh.akamaihd.net
    https://mpdhdsnappytv-vh.akamaihd.net https://mmdhdsnappytv-vh.akamaihd.net
    https://dwo3ckksxb0v.cloudfront.net https://media.riffsy.com https://*.giphy.com
    https://media.tenor.com https://c.tenor.com ; default-src 'self'; form-action 'self'
    https://twitter.com https://*.twitter.com; font-src 'self' https://*.twimg.com; frame-src 'self'
    https://twitter.com https://mobile.twitter.com https://pay.twitter.com https://cards-
    frame.twitter.com https://accounts.google.com/ https://client-api.arkoselabs.com/
    https://iframe.arkoselabs.com/ https://recaptcha.net/recaptcha/
    https://www.google.com/recaptcha/ https://www.gstatic.com/recaptcha/; img-src 'self' blob:
    data: https://*.cdn.twitter.com https://ton.twitter.com https://*.twimg.com
    https://analytics.twitter.com https://cm.g.doubleclick.net https://www.google-analytics.com
    https://maps.googleapis.com https://www.periscope.tv https://www.pscp.tv
```



```
https://media.riffsy.com https://*.giphy.com https://media.tenor.com https://c.tenor.com
https://*.pscp.tv https://*.periscope.tv https://prod-periscope-profile.s3-us-west-
2.amazonaws.com https://platform-lookaside.fbsbx.com https://scontent.xx.fbcdn.net
https://scontent-sea1-1.xx.fbcdn.net https://*.googleusercontent.com; manifest-src 'self';
media-src 'self' blob; https://twitter.com https://*.twimg.com https://*.vine.co https://*.pscp.tv
https://*.video.pscp.tv https://dhdsnappytv-vh.akamaihd.net https://pdhdsnappytv-
vh.akamaihd.net https://mdhdsnappytv-vh.akamaihd.net https://mdhdsnappytv-
vh.akamaihd.net https://mpdhdsnappytv-vh.akamaihd.net https://mmdhdsnappytv-
vh.akamaihd.net https://mdhdsnappytv-vh.akamaihd.net https://mpdhdsnappytv-
vh.akamaihd.net https://mmdhdsnappytv-vh.akamaihd.net
https://dwo3ckksxlb0v.cloudfront.net; object-src 'none'; script-src 'self' 'unsafe-inline'
https://*.twimg.com https://recaptcha.net/recaptcha/ https://www.google.com/recaptcha/
https://www.gstatic.com/recaptcha/ https://client-api.arkoselabs.com/ https://www.google-
analytics.com https://twitter.com https://app.link https://accounts.google.com/gsi/client
https://appleid.cdn-apple.com/appleauth/static/jsapi/appleid/1/en_US/appleid.auth.js 'nonce-
NDM1ZDg1MGEtY2I3Mi00NDA4LTkxZDEtMTA5NjRhYjhiOGNm'; style-src 'self'
'unsafe-inline' https://accounts.google.com/gsi/style https://*.twimg.com; worker-src 'self'
blob;; report-uri https://twitter.com/i/csp_report?a=O5RXE%3D%3D%3D&ro=false
< strict-transport-security: max-age=631138519
< cross-origin-opener-policy: same-origin-allow-popups
< cross-origin-embedder-policy: unsafe-none
< x-response-time: 158
< x-connection-hash: f612b8dfb3aa4610920894ceeac4fb3a530d1376885dbd5c0a6e976af0425246
```

O HTML retornado é o mesmo do browser.

#### 14. Make any request to <https://httpbin.org/anything> and just set some nonsense headers (like panda: elephant). What happens?

```
> GET /anything HTTP/2
> Host: httpbin.org
> user-agent: insomnia/2023.1.0
> panda: elephant;
> accept: */*
```

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

```
< HTTP/2 200
< date: Mon, 27 Mar 2023 22:50:01 GMT
< content-type: application/json
< content-length: 376
< server: gunicorn/19.9.0
< access-control-allow-origin: *
< access-control-allow-credentials: true
```

```
{
  "args": {},
  "data": "",
  "files": {},
  "form": {},
```



```
"headers": {  
  "Accept": "*/*",  
  "Host": "httpbin.org",  
  "Panda": "elephant;",  
  "User-Agent": "insomnia/2023.1.0",  
  "X-Amzn-Trace-Id": "Root=1-64221d99-2f68c8b3418263f860972055"  
},  
"json": null,  
"method": "GET",  
"origin": "189.15.227.53",  
"url": "https://httpbin.org/anything"  
}
```

**15. Request <https://httpbin.org/status/404> and <https://httpbin.org/status/200>. Where your tool (postman, cURL, etc) shows the request results?**

```
> GET /status/404 HTTP/2  
> Host: httpbin.org  
> user-agent: insomnia/2023.1.0  
> accept: */*  
  
< HTTP/2 404  
< date: Mon, 27 Mar 2023 22:51:25 GMT  
< content-type: text/html; charset=utf-8  
< content-length: 0  
< server: gunicorn/19.9.0  
< access-control-allow-origin: *  
< access-control-allow-credentials: true  
  
> GET /status/200 HTTP/2  
> Host: httpbin.org  
> user-agent: insomnia/2023.1.0  
> accept: */*  
  
< HTTP/2 200  
< date: Mon, 27 Mar 2023 22:52:05 GMT  
< content-type: text/html; charset=utf-8  
< content-length: 0  
< server: gunicorn/19.9.0  
< access-control-allow-origin: *  
< access-control-allow-credentials: true
```

**16. Request <https://httpbin.org/anything> and set a username and password. Many tools will make it automatically: if the site you are using needs authentication, it can send this information to you. Find out how to do it in the tool you are using.**

```
> GET /anything HTTP/2  
> Host: httpbin.org  
> user-agent: insomnia/2023.1.0
```



> authorization: Basic YWRtaW46MTIzNDU=

> accept: \*/\*

\* Connection state changed (MAX\_CONCURRENT\_STREAMS == 128)!

< HTTP/2 200

< date: Mon, 27 Mar 2023 22:55:33 GMT

< content-type: application/json

< content-length: 397

< server: gunicorn/19.9.0

< access-control-allow-origin: \*

< access-control-allow-credentials: true

```
{
  "args": {},
  "data": "",
  "files": {},
  "form": {},
  "headers": {
    "Accept": "*/*",
    "Authorization": "Basic YWRtaW46MTIzNDU=",
    "Host": "httpbin.org",
    "User-Agent": "insomnia/2023.1.0",
    "X-Amzn-Trace-Id": "Root=1-64221ee5-7797a545297af43f61af0c8c"
  },
  "json": null,
  "method": "GET",
  "origin": "189.15.227.53",
  "url": "https://httpbin.org/anything"
}
```

*Exercises based on Julia Evans Curl exercises.*