



南開大學
Nankai University

互联网数据库开发个人作业 1

姓名：房书睿

学号：2213459

学院：密码与网络空间安全学院

专业：信息安全法学

2025 年 12 月 18 日

Contents

1 实验要求	2
2 调研网页请求的不同方式	2
2.1 捕捉 GET 请求	2
2.2 POST 请求	5
2.3 OPTION 请求	8
2.4 PUT 请求	10
2.5 DELETE 请求的捕获	10
3 使用 JQuery 实现事件触发	11
4 实现浏览器插件	12
5 总结与收获	13

1 实验要求

- 针对任意网页，调研其不同方式请求，至少包括 get、post 请求，写出或截图其请求及相应数据包；
- 针对任意网页，使用 Jquery，能够触发某一事件，写出至少三条语句，截图响应前后不同的状态；
- 完成一个浏览器插件，功能不限，文档中写明功能及代码。

2 调研网页请求的不同方式

本实验使用 Google Chrome 浏览器，调研的网站是 <https://www.instagram.com/> 使用浏览器开发者工具调研具体的请求

2.1 捕捉 GET 请求

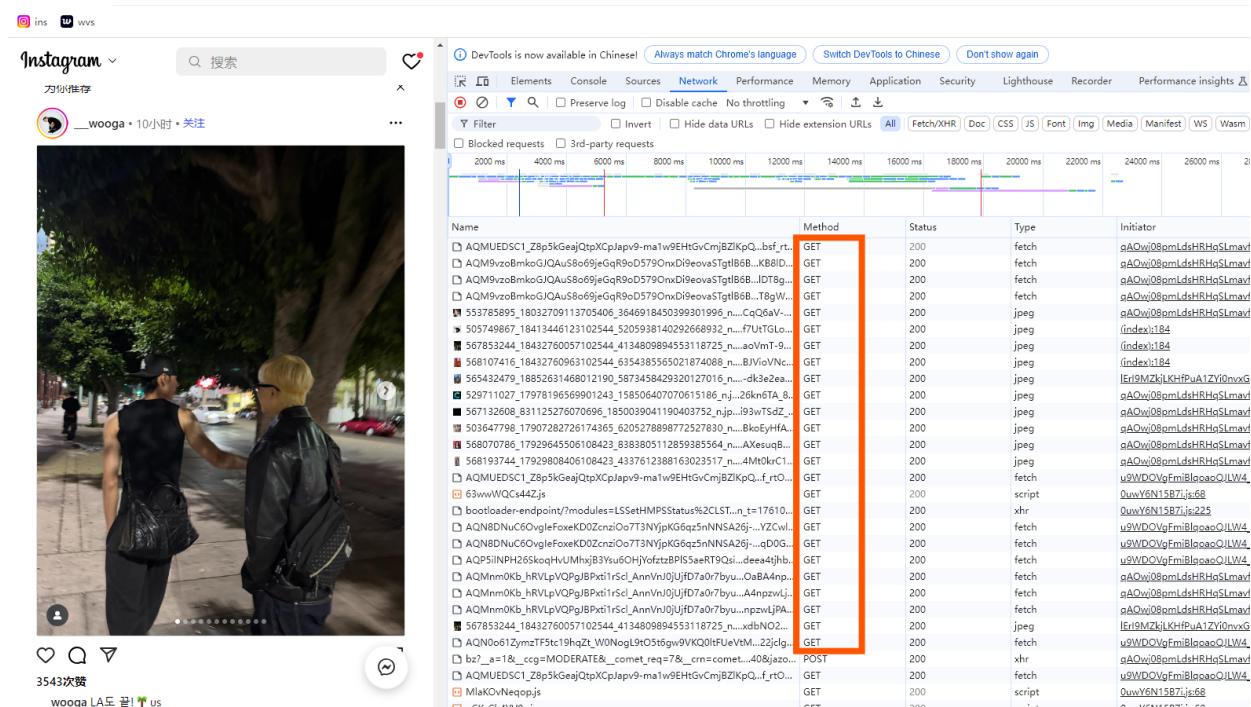


图 2.1: 查看 GET 请求

在刚刚进入网站时，由于此时用户还没有与网站产生交互信息，可以看到大部分请求都是 GET。随机点一个 GET 请求，看一下详细信息。

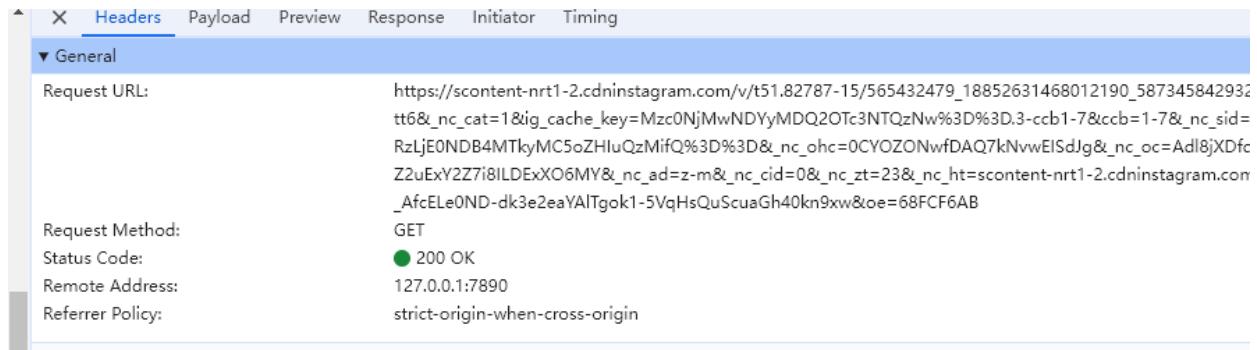


图 2.2: General

General 信息显示了访问的 URL, 请求方式, HTTP 状态码, 客户端实际连接的远程服务器 IP 地址和端口号, 还有控制 Referer 请求头.



图 2.3: Response Headers

相应头信息代表了服务器对此次请求的响应, 具体内容如下:

accept-ranges: bytes 表示服务器支持按字节范围请求资源, 允许断点续传;

access-control-allow-origin: https://www.instagram.com 表示该资源允许来自 Instagram 网站的跨域请求;

alt-svc: h3=":443"; ma=86400 表示服务器支持 HTTP/3 协议, 建议在 86400 秒内使用;

cache-control: max-age=1209600, no-transform 表示资源可缓存 14 天且不得被代理转换;

content-digest: adler32=1407458626 提供内容的 Adler-32 校验值用于完整性验证;

content-length: 66550 表示响应体大小为 66,550 字节;

content-type: image/jpeg 表明返回内容为 JPEG 图像格式;

cross-origin-resource-policy: cross-origin 允许资源被任意跨源请求加载;

date: Tue, 21 Oct 2025 07:22:38 GMT 是响应生成的服务器时间;

last-modified: Sat, 18 Oct 2025 19:05:25 GMT 表示资源最后修改时间, 用于缓存验证;

no-vary-search: key-order, params=("_nc_gid" "oh" "oe" "_nc_oc" "_nc_ohc" "_nc_cat" "_nc_ht") 表示缓存时忽略这些参数的顺序与存在;

timing-allow-origin: * 允许所有来源获取该资源的性能计时数据;

vary: Origin 表示响应可能因请求来源不同而变化, 影响缓存键;

x-additional-error-detail: 为自定义空字段, 未提供额外错误信息;

x-crypto-project: 0 可能为功能开关标识, 当前值为 0;

x-fb-connection-quality: EXCELLENT; q=0.9, rtt=4, rtx=0, c=365, mss=1380, tbw=1168286, tp=-1, tpl=-1, uplat=2, ullat=-1 是 Facebook 的连接质量指标, 显示当前网络质量极佳;

x-fb-ptm-uuid: 6F44012ACAE7DBB0FF7CC982BFC4B94 是用于请求追踪的唯一标识符;

x-robots-tag: noarchive, noindex 指示搜索引擎不要缓存或索引该资源。

▼ Request Headers	
:authority:	scontent-nrt1-2.cdninstagram.com
:method:	GET
:path:	/v/t51.82787-15/565432479_18852631468012190_5873458429320127016_n.jpg?stp=dst-jpegr_e35_p480x480_tt68_nc_cat=1&ig_cache_key=Mzc0NjMwNDYyMDQ2OTc3NTQzNw%3D%3D.3-ccb1-7&ccb=1-7&_nc_sid=58cdad&efg=eylZW5jb2RlX3RhZyl6InhwaWRzljE0NDB4MTkyMC5oZHIuQzMifQ%3D%3D&_nc_ohc=0CYOZONwfDAQ7kNvwElSdJg&_nc_oc=Adl8jXDfqOUejwb00AxZV4IPGKjMx5XrzfynB1lxG5bNSZ2uExY2Z7i8lDExX06MY&_nc_ad=z-m&_nc_cid=0&_nc_zt=23&_nc_ht=scontent-nrt1-2.cdninstagram.com&_nc_gid=GY2lsICk_yUBGwP0YCpVg&oh=0_&fcElE0ND-dk3e2eaYAItgok1-5VqHsQuScuaGh40kn9xw&oe=68FCF6AB
:scheme:	https
Accept:	image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Accept-Encoding:	gzip, deflate, br, zstd
Accept-Language:	zh-CN,zh;q=0.9,ja;q=0.8
Origin:	https://www.instagram.com
Priority:	i
Referer:	https://www.instagram.com/
Sec-Ch-Ua:	"Not)A;Brand";v="99", "Google Chrome";v="127", "Chromium";v="127"
Sec-Ch-Ua-Mobile:	?0
Sec-Ch-Ua-Platform:	"Windows"
Sec-Fetch-Dest:	image
Sec-Fetch-Mode:	cors
Sec-Fetch-Site:	cross-site
User-Agent:	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.0.0 Safari/537.36

图 2.4: Request Headers

请求头信息含义如下：

authority: scontent-nrt1-2.cdninstagram.com 指定请求的目标主机，即 Instagram 内容分发网络中的图片服务器；

:method: GET 表示该 HTTP/2 伪头部字段定义了请求方法为获取资源；

:path: /v/t51.82787-15/... 包含请求的完整路径与查询参数，用于定位特定缩略图并携带缓存、设备、加密等标识；

:scheme: https 表明使用 HTTPS 协议进行安全传输；

accept: image/avif,image/webp,... 声明客户端支持的图像格式优先级，偏好现代格式如 AVIF/WebP；

accept-encoding: gzip, deflate, br, zstd 表示客户端可接受多种内容编码方式，以实现压缩传输优化；

accept-language: zh-CN,zh;q=0.9,ja;q=0.8 表明用户首选简体中文，其次日语，用于内容本地化；

origin: https://www.instagram.com 标识跨域请求来源，用于 CORS 安全策略判断；

priority: i 是新兴优先级信号，表示此图像请求具有高重要性；

referer: https://www.instagram.com/ 指出用户从 Instagram 主站跳转而来，用于统计与防盗链；

sec-ch-ua: "Not)A;Brand";v="99", "Google Chrome";v="127", "Chromium";v="127" 向服务器通报浏览器品牌与版本，用于兼容性适配；

sec-ch-ua-mobile: ?0 表示用户代理非移动端（即桌面端）；

sec-ch-ua-platform: "Windows" 说明操作系统为 Windows；

sec-fetch-dest: image 指明本次请求目标是一个图像资源；

sec-fetch-mode: cors 表示请求通过 CORS 模式发起，常见于跨域资源加载；

sec-fetch-site: cross-site 表明请求是从当前源以外的第三方站点发起的跨站请求；

user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) ... 提供浏览器详细信息，包括平台、内核及 Chrome 版本。

2.2 POST 请求

搜索一个 ins 博主，发现出现了很多 post 请求信息。

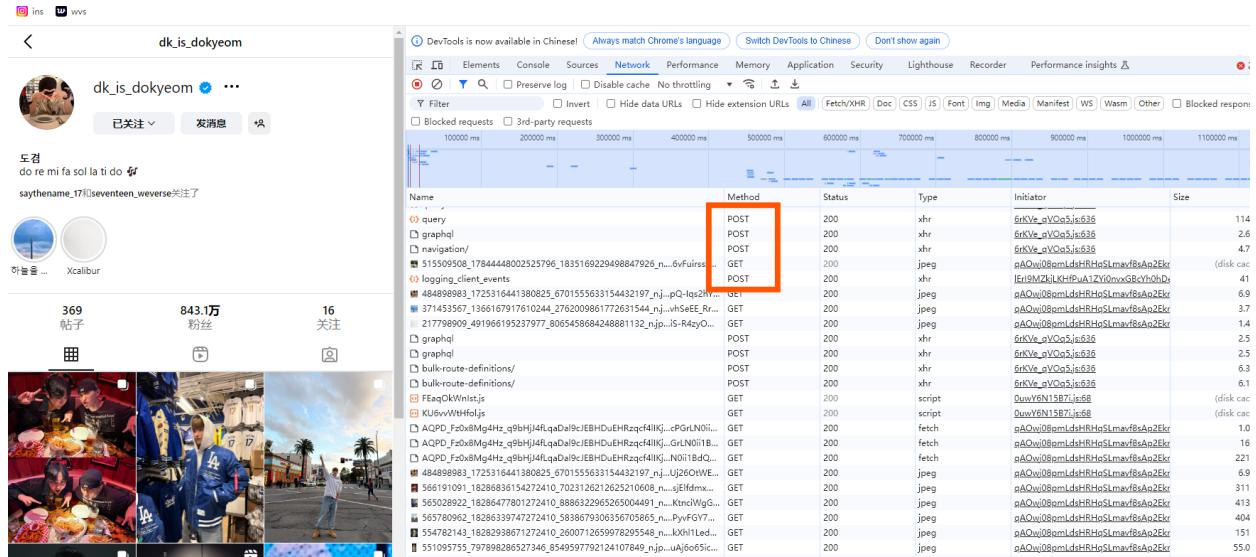


图 2.5: 捕捉 POST 请求

对第一个进行详细分析。

General	
Request URL:	https://www.instagram.com/graphql/query
Request Method:	POST
Status Code:	200 OK
Remote Address:	127.0.0.1:7890
Referrer Policy:	strict-origin-when-cross-origin

图 2.6: General

对不同的部分进行分析: set-cookie: csrf_token=nBY55us1YBuzubNgiprKtUxvjgxStfI; Domain=.instagram.com; expires=Wed, 25-Nov-2026 07:42:59 GMT; Max-Age=34560000; Path=/; Secure 设置了一个 CSRF 令牌用于防止跨站请求伪造攻击，适用于整个 Instagram 域；

HttpOnly; Path=/; SameSite=Lax; Secure set-cookie: rur="VLL5475689997795541792568579:01feba55a03; Domain=.instagram.com;

HttpOnly; Path=/; SameSite=Lax; Secure 设置了区域用户路由标识符，带有 HttpOnly 标志防止 JavaScript 访问；

set-cookie: ds_user_id=75689997795; Domain=.instagram.com; expires=Mon, 19-Jan-2026 07:42:59 GMT; Max-Age=7776000; Path=/; SameSite=None; Secure 设置了用户的唯一标识符，允许跨站访问但需 HTTPS；

set-cookie: th_eu_pref="" 多次出现，设置相同的值为空字符串并立即过期，用于清除特定域下的偏好设置；

strict-transport-security: max-age=31536000 强制浏览器在未来一年内通过 HTTPS 访问该站点，增强安全性；

Response Headers	
Access-Control-Allow-Credentials:	true
Access-Control-Allow-Origin:	https://www.instagram.com
Access-Control-Expose-Headers:	X-Stack, Content-Type, X-IG-Set-WWW-Claim
Alt-Svc:	h3=":443"; ma=86400
Cache-Control:	private, no-cache, no-store, must-revalidate
Content-Encoding:	zstd
Content-Language:	zh-cn
Content-Length:	113127
Content-Security-Policy:	report-uri https://www.instagram.com/security/csp_report/; default-src 'self' https://www.instagram.com https://*.instagram.com https://*.cdninstagram.com https://*.facebook.com https://*.fbsbx.com http://*.giphy.com; font-src data: https://*.fbcdn.net https://*.instagram.com https://*.cdninstagram.com https://www.instagram.com https://*.cdninstagram.com https://*.fbcdn.net; manifest-src 'self' https://*.fbcdn.net https://www.instagram.com https://*.www.instagram.com https://*.cdninstagram.com https://*.fbcdn.net https://*.facebook.net https://*.fbcdn.net 'unsafe-inline' 'unsafe-eval' blob; https://www.instagram.com 'unsafe-inline'; connect-src 'self' https://instagram.com https://www.instagram.com https://graph.instagram.com https://*.graph.instagram.com https://i.instagram.com/graphql_www https://*.cdninstagram.com https://api.instagram.com https://i.instagram.com https://*.i.instagram.com https://i.threads.net https://i.threads.com https://*.od.threads.net https://*.od.threads.com wss://www.chat.instagram.com https://*.facebook.com https://*.fbcdn.net https://*.facebook.net chrome-experiment-worker-src 'self' blob: https://www.instagram.com; frame-src 'self' https://instagram.com https://www.instagram.com https://staticxx.facebook.com https://www.facebook.com https://web.facebook.com https://connect.facebook.com; object-src 'none'; upgrade-insecure-requests
Content-Type:	application/json; charset=utf-8
Cross-Origin-Embedder-Policy-Report-Only:	require-corp;report-to="coep"
Cross-Origin-Opener-Policy:	same-origin-allow-popups;report-to="coop"
Date:	Tue, 21 Oct 2025 07:42:59 GMT
Expires:	Sat, 01 Jan 2000 00:00:00 GMT
Pragma:	no-cache

图 2.7: Response Headers

vary: Accept-Language, Cookie, Accept-Encoding 指示缓存机制应根据语言、Cookie 和编码的不同进行区分存储；

x-aed: 605 可能是自定义头，表示某种内部状态或诊断信息；
x-content-type-options: nosniff 禁止浏览器猜测内容类型，防止 MIME 类型嗅探攻击；
x-fb-connection-quality: EXCELLENT; q=0.9, rtt=1, rtx=0, c=126, mss=1380, tbw=96848, tp=-1, tpl=-1, uplat=1468, ullat=10 提供 Facebook 的连接质量指标，显示当前网络质量极佳；
x-frame-options: SAMEORIGIN 防止页面被嵌入到不同源的框架中，仅允许同源嵌套。

▼ Request Headers	
:authority:	www.instagram.com
:method:	POST
:path:	/graphql/query
:scheme:	https
Accept:	*/*
Accept-Encoding:	gzip, deflate, br, zstd
Accept-Language:	zh-CN,zh;q=0.9,ja;q=0.8
Content-Length:	2150
Content-Type:	application/x-www-form-urlencoded
Cookie:	csrftoken=nBY55uslYBuzubNgiprKtUxvjgxnStfI; ds_user_id=75689997795; ig_did=28500A23-EC09-4741-8509-F5319BE81558; sessionid=75689997795%3AXYrp0epAAWCinK%3A2%3AAAYgVN1sdztPr-2Apq3O_yp1HtrZXNzm1itf2kavpw; mid=aPczWQALAAQgfb-3zGK0gYZAy2-X; wd=559x911; rur=VLL\05475689997795\0541792568563:01fe2fa4cff17ac0e91a9d8ec1ab71a2631977ac8bf3277e0b6b074baaadccf572139c4"
Origin:	https://www.instagram.com
Priority:	u=1, i
Referer:	https://www.instagram.com/
Sec-Ch-Prefers-Color-Scheme:	light
Sec-Ch-Ua:	"Not)A;Brand";v="99", "Google Chrome";v="127", "Chromium";v="127"
Sec-Ch-Ua-Full-Version-List:	"Not)A;Brand";v="99.0.0", "Google Chrome";v="127.0.6533.120", "Chromium";v="127.0.6533.120"
Sec-Ch-Ua-Mobile:	?0
Sec-Ch-Ua-Model:	""
Sec-Ch-Ua-Platform:	"Windows"
Sec-Ch-Ua-Platform-Version:	"15.0.0"
Sec-Fetch-Dest:	empty
Sec-Fetch-Mode:	cors
Sec-Fetch-Site:	same-origin
User-Agent:	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.0.0 Safari/537.36
X-Asbd-Id:	359341

图 2.8: Request Headers

sec-ch-prefers-color-scheme: light 表示客户端偏好使用浅色主题界面；

sec-ch-ua: "Not)A;Brand";v="99", "Google Chrome";v="127", "Chromium";v="127" 向服务器通报浏览器品牌与主版本号，用于兼容性判断；

sec-ch-ua-full-version-list: "Not)A;Brand";v="99.0.0.0", "Google Chrome";v="127.0.6533.120", "Chromium";v="127.0.6533.120" 提供更详细的浏览器完整版本信息；

sec-ch-ua-mobile: ?0 表示用户代理来自桌面设备而非移动端；

sec-ch-ua-model: "" 设备型号为空，桌面浏览器通常不提供具体设备型号；

sec-ch-ua-platform: "Windows" 指明操作系统为 Windows；

sec-ch-ua-platform-version: "15.0.0" 传递平台版本信息，可能为模拟或简化值；

sec-fetch-dest: empty 表示该请求由 JavaScript 发起，目标类型为空（非资源加载）；

sec-fetch-mode: cors 指示请求采用跨域资源共享（CORS）模式；

sec-fetch-site: same-origin 表明请求目标与当前页面同源；

user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) ... 标准用户代理字符串，标识 Chrome 127 在 Windows 10 上运行；

x-asbd-id: 359341 Facebook/Instagram 内部用于追踪服务路由的 ASBD(Application-Specific Backend Distribution) ID；

x-bloks-version-id: d472af6df5cc606197723ed51adaa088... 标识 Bloks 平台(Meta 的无代码框架) 当前版本，用于客户端一致性校验；

x-csrftoken: nBY55uslYBuzubNgiprKtUxvjgxnStfI 携带 CSRF 安全令牌，防止跨站请求伪造；

x-fb-friendly-name: PolarisProfilePostsQuery 提供可读的 GraphQL 查询名称，便于后端监控与调试；

x-fb-lsd: LOCKgZKso8JPX_68hRgDA5 Meta 内部使用的反 CSRF 令牌 (LSD token)，用于增强请求安全性；

x-ig-app-id: 936619743392459 标识请求来自 Instagram 官方 Web 应用的特定客户端 ID；

`x-root-field-name: xdt_api_v1_feed_user_timeline_graphql_connection` 指定 GraphQL 查询的根字段名称，用于路由解析。

2.3 OPTION 请求

Name	Method	Status
realtime?x-dgw-appid=936619743392459&x-dgw-appvers...5-830a-c84...		Finished
init_script/?worker_type=MODULE&_hs=20382.HCSV2%3...3&use_broa...	GET	(pending)
chat?sid=7919135616925695&cid=1ed77ca6-f300-4d6b-817e-45d9b3e...	GET	101
streamcontroller?x-dgw-appid=936619743392459&x-dgw...6-a698-1a6...	GET	101
realtime?x-dgw-appid=936619743392459&x-dgw-appvers...5-830a-c84...	GET	101
AQNN9ko2mBN742o3EONiAKNjCXCDrFhSJNhrUoZjGyp8pucmUK...YjQ...	GET	200
AQN0o61ZymzTF5tc19hqZt_W0NogL9tO5t6gw9VKQ0ltFUeVtM...2jclg&...	GET	200
logging_client_events	OPTIONS	200
bz?_a=1&_ccg=Moderate&_comet_req=7&_crn=comet....40&jazo...	POST	200
logging_client_events	POST + <u>Preflight</u>	200

图 2.9: OPTION 请求

General 信息与上面两个请求方式类似，Request Headers response headers 有所不同。

▼ Response Headers	
Access-Control-Allow-Headers:	x-asbd-id,x-web-session-id
Access-Control-Allow-Methods:	GET, POST, DELETE, OPTIONS
Access-Control-Allow-Origin:	*
Access-Control-Max-Age:	300
Alt-Svc:	h3=":443"; ma=86400
Cache-Control:	private, no-cache, no-store, must-revalidate
Content-Length:	0
Content-Type:	application/json; charset=UTF-8
Date:	Tue, 21 Oct 2025 07:41:23 GMT
Expires:	Sat, 01 Jan 2000 00:00:00 GMT
Facebook-Api-Version:	v18.0
Pragma:	no-cache
Strict-Transport-Security:	max-age=31536000; preload; includeSubDomains
Vary:	Origin
X-Fb-Aed:	605
X-Fb-Connection-Quality:	EXCELLENT; q=0.9, rtt=8, rtx=14, c=88, mss=1380, tbw=30278, tp=-1, tpi=-1, uplat=152, ullat=0
X-Fb-Debug:	WcgQpWlgOW51tmxQaYUv2Nyc9jgBNQejeOC078I+yCqOWIS3gueL9ruGAKjfh5uw6LvnM2YkjxQX2AR2BZH2A==
X-Fb-Request-Id:	AOxri4vhk0APglVP6sqUzD
X-Fb-Rev:	1028698734
X-Fb-Trace-Id:	GyWCvfGDLG8
X-Stack:	www

图 2.10: Response Headers

`access-control-allow-headers: x-asbd-id,x-web-session-id` 允许客户端在跨域请求中携带指定的自定义请求头字段；

`access-control-allow-methods: GET, POST, DELETE, OPTIONS` 指明该资源支持的跨域请求方法类型；

`access-control-allow-origin: *` 允许任何来源的跨域访问，适用于公开 API，但需注意安全风险；

access-control-max-age: 300 设置预检请求 (preflight) 结果缓存时间为 300 秒，减少重复 OPTIONS 请求；

alt-svc: h3=":443"; ma=86400 表示服务器支持 HTTP/3 协议，建议客户端在未来 86400 秒内使用端口 443 进行连接；

cache-control: private, no-cache, no-store, must-revalidate 禁止缓存响应内容，确保每次请求都重新验证，用于保护敏感或动态数据；

content-length: 0 表示响应体为空，通常用于仅返回状态码的操作（如删除或成功确认）；

content-type: application/json; charset=UTF-8 指定返回内容为 UTF-8 编码的 JSON 格式，符合现代 Web API 规范；

date: Tue, 21 Oct 2025 07:41:23 GMT 记录服务器生成响应的时间戳；

expires: Sat, 01 Jan 2000 00:00:00 GMT 显式设置过期时间为过去时间，强化禁止缓存策略；

facebook-api-version: v18.0 标识本次请求调用的是 Facebook Graph API 的第 18.0 版本，便于版本管理与兼容；

pragma: no-cache 为兼容 HTTP/1.0 客户端而设置，进一步强调不缓存指令；

strict-transport-security: max-age=31536000; preload; includeSubDomains 启用 HSTS 策略，要求一年内所有子域名必须通过 HTTPS 访问，并支持预加载；

vary: Origin 指示缓存系统应根据请求来源 (Origin) 不同分别缓存响应，配合 CORS 使用；

x-fb-aed: 605 Meta 内部使用的调试标识符，可能对应特定数据中心或服务节点；

x-fb-connection-quality: EXCELLENT; q=0.9, rtt=8, rtx=14, c=88, mss=1380, tbw=30278, tp=-1, tpl=-1, uplat=152, ullat=0 提供详细的网络质量指标，显示当前连接质量优良；

x-fb-debug: WcgQpWIg0W...== 包含用于后端调试的跟踪令牌，仅在开发或授权环境下有效；

x-fb-request-id: A0xri4ivhkOAPglYP6sqUzD 为本次请求分配的唯一 ID，用于日志追踪与问题排查；

x-fb-rev: 1028698734 表示服务器当前代码版本号 (revision)，便于定位部署版本；

x-fb-trace-id: GyWCvfGDLG8 分布式追踪 ID，用于跨服务链路监控与性能分析；

x-stack: www 标识请求经过的服务栈类型，此处表示来自标准 Web 前端服务。

▼ Request Headers	
:authority:	graph.instagram.com
:method:	OPTIONS
:path:	/logging_client_events
:scheme:	https
Accept:	/*
Accept-Encoding:	gzip, deflate, br, zstd
Accept-Language:	zh-CN,zh;q=0.9,ja;q=0.8
Access-Control-Request-Headers:	x-asbd-id,x-web-session-id
Access-Control-Request-Method:	POST
Origin:	https://www.instagram.com
Priority:	u=1, i
Referer:	https://www.instagram.com/
Sec-Fetch-Dest:	empty
Sec-Fetch-Mode:	cors
Sec-Fetch-Site:	same-site
User-Agent:	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.0.0 Safari/537.36

图 2.11: Request Headers

authority: graph.instagram.com 指定请求的目标主机为 Instagram 的图形 API 服务器；

:method: OPTIONS 表示该请求为预检请求 (preflight)，用于 CORS 安全机制中探测服务器允许的跨域策略；
 :path: /logging_client_events 指明请求路径为客户端事件日志上报接口；
 :scheme: https 表示通信协议为 HTTPS，确保传输安全；
 accept: */* 表示客户端可接受任意内容类型，常见于 API 请求；
 accept-encoding: gzip, deflate, br, zstd 声明支持多种压缩编码方式，以优化传输效率；
 accept-language: zh-CN,zh;q=0.9,ja;q=0.8 表示用户首选简体中文，其次中文和日语，用于内容本地化；
 access-control-request-headers: x-asbd-id,x-web-session-id 在预检请求中声明实际请求将携带的自定义请求头字段；
 access-control-request-method: POST 指明实际请求将使用 POST 方法；
 origin: https://www.instagram.com 标识请求来源，用于服务器判断是否允许跨域访问；
 priority: u=1, i 表示该请求具有较低优先级 (u=1)，且为非关键资源 (i=idle)；
 referer: https://www.instagram.com/ 指出用户从 Instagram 主站发起请求，用于统计与安全校验；
 sec-fetch-dest: empty 表示该请求由脚本发起，目标资源类型为空，符合预检请求特征；
 sec-fetch-mode: cors 指示请求采用跨域资源共享 (CORS) 模式；
 sec-fetch-site: same-site 表明请求目标与来源同属一个注册域名 (如 instagram.com 及其子域)，属于同站请求；
 user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) ... 标识客户端为运行在 Windows 10 上的 Chrome 127 浏览器。

2.4 PUT 请求

PUT 请求通常用于更新资源。PUT 通常要求客户端提交完整的资源信息，它会替换服务器端的整个资源。

我在评论区发表了一条评论，但是没有检测到 PUT 请求，而是 POST 请求。

查询资料后得知，在主流社交平台的 API 设计中：

创建资源 → POST 到集合路径

例：POST /posts, POST /comments, POST /likes

更新资源 → PUT 或 PATCH 到具体资源路径

例：PUT /comments/123

删除资源 → DELETE /comments/123

如果我连续点击“发送评论”两次，Instagram 通常会发送两个 POST 请求 → 生成两条评论（或通过防重机制阻止）。这正是 POST 的预期行为。而如果用 PUT，系统会认为用户在“更新同一个资源”，可能导致逻辑混乱。

2.5 DELETE 请求的捕获

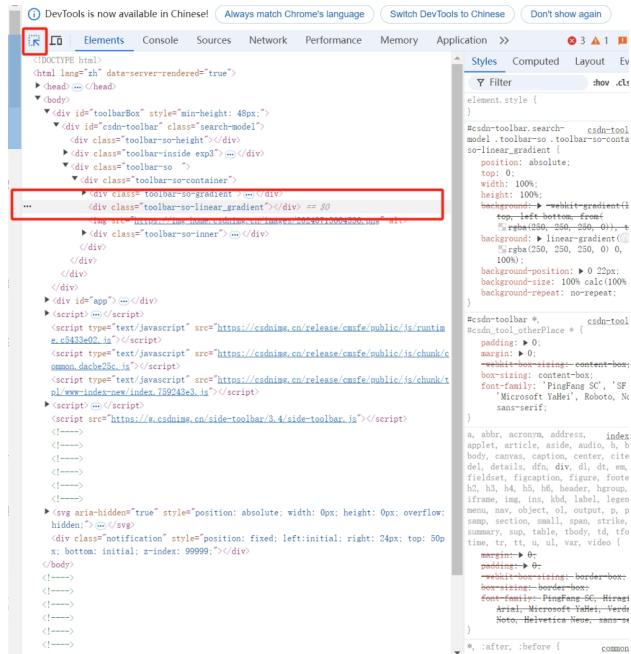
我删除了刚刚发送的评论，此时也显示的是 POST 请求而不是 DELETE。从 RESTful 设计原则来看，删除资源理应使用 DELETE 方法。但在实际前端开发中，尤其是像 Instagram 这样的大型 SPA，经常看到“删除操作”也通过 POST 发起。这看似违背规范，实则有其合理的技术和安全考量。虽然

DELETE 在语义上更准确，但 Instagram 使用 POST 删除评论是出于实际工程、安全和架构的综合考量。

3 使用 JQuery 实现事件触发

我在 `instagram` 上没有找到自带的 JQuery 插件，查询过后发现 CSDN 自带插件，于是利用 CSDN 完成 JQuery 事件触发。

可以通过修改 `div` 元素的内容，实现一个基本的事件触发。



例如此处的 `div` 对应的就是 csdn 导航栏的内容。



在 `console` 选项中输入 `JQuery` 代码，就可以轻易地改变导航栏的内容。

我们输入

```
$(".toolbar-so-linear-gradient").html("Hello");
```

之后的导航栏变成了



4 实现浏览器插件

我设计了一个页面倒计时插件，可以设置不同的时间，记录当前页面使用时长。快结束的时候弹出窗口进行提醒，最后触发结束时钟。

具体功能如下：

- 在任意网页右下角显示可控的倒计时浮层，支持开始、暂停、恢复、结束。
- 通过扩展弹窗输入分钟与秒，面向当前活动标签页生效。
- 记忆上次输入的时间，提升重复使用效率。

架构概览：

1 Manifest V3 扩展，主要由三个部分构成：

- 弹窗：负责用户输入与操作，发送消息到活动标签页。
- 内容脚本：在页面注入并渲染倒计时浮层，执行业务逻辑。
- 清单：定义权限、弹窗、内容脚本注入规则。

2 组件设计

- 弹窗界面
 1. 文件：popup.html、popup.css、popup.js
 2. 功能：输入分钟/秒并触发开始；提供暂停、恢复、结束操作；提示状态；读写 chrome.storage.local
- 内容脚本
 1. 文件：content.js、content.css
 2. 功能：在页面右下角创建浮层；管理计时状态与渲染；处理来自弹窗的消息；提供浮层上的暂停、恢复、结束按钮。
- 清单
 1. 文件：manifest.json
 2. 内容：manifest_version=3; action.default_popup 指向弹窗；content_scripts 注入到 <all_urls>；声明 tabs、storage 权限

实现效果如下：



可以在 chrome 浏览器中导入我们的插件，在右上角可以设置自己想要设定的倒计时时间，也可以随时暂停、复位等等。

可以随时查看剩余时间。



结束的时候也会显示提醒。



5 总结与收获

通过本次实验，我深入理解了网页中 GET 与 POST 请求的区别，并熟练使用浏览器开发者工具捕获和分析网络请求数据包，掌握了请求头、参数传递及响应内容的查看方法。

在前端动态操作方面，我通过 jQuery 实践了元素的隐藏、内容修改和事件绑定等功能。相比原生 JavaScript，jQuery 语法简洁、兼容性好，显著提升了开发效率，尤其在处理动态内容更新时优势明显。

此外，我还完成了一个简单的浏览器插件开发，了解了插件的基本结构（如 `manifest.json`、`popup` 和 `content script`）以及如何向网页注入自定义功能。这让我认识到浏览器扩展在增强用户体验方面的灵活性和实用性。

整体上，本次实验将理论与实践紧密结合，有效提升了我在前端调试、交互开发和功能拓展等方面的综合能力，为后续更复杂的 Web 开发任务打下了良好基础。