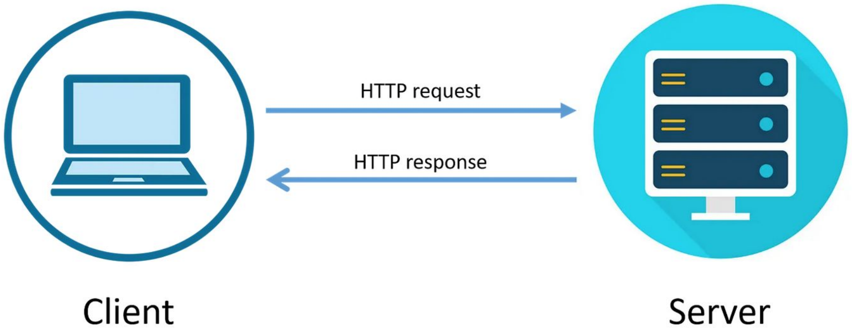
HTTP Request Protocol

Understanding HTTP Requests and Responses

Introduction to HTTP

- HTTP (Hypertext Transfer Protocol) is the foundation of data communication on the web.
- It is a request-response protocol between a client (browser) and a server.
- HTTP is stateless, meaning each request is independent of others.



HTTP Request Methods

- **GET:** Retrieve data from the server.
- **POST:** Send data to the server to create a resource.
- **PUT:** Replace a resource on the server.
- **DELETE:** Remove a resource from the server.
- **PATCH:** Partially update a resource on the server.
- **HEAD:** Retrieve headers without the body.
- **OPTIONS:** Describe communication options for the target resource.
- **CONNECT:** Establish a tunnel to the server (used for HTTPS).
- **TRACE:** Echoes the received request for debugging purposes.

Anatomy of an HTTP Request

An HTTP Request consists of:

1. **Request Line:**
   - Method (e.g., GET, POST)
   - URL (e.g., /index.html)
   - HTTP Version (e.g., HTTP/1.1)
2. **Headers:**
   - Key-value pairs providing metadata (e.g., Content-Type, Authorization).
3. **Body (optional):**
   - Data sent to the server (e.g., JSON, form data)

Example of an HTTP Request

```
GET https://jsonplaceholder.typicode.com/albums/1/photos HTTP/2

Host: jsonplaceholder.typicode.com
Accept-Language: en-US
Cache-Control: no-cache
```

- **Request Line:** `GET jsonplaceholder.typicode.com/albums/1/photos HTTP/2`
- **Headers:** `Host`, `Accept-Language`, `Cache-Control`
- **Body:** No body sent in this example

Anatomy of an HTTP Response

An HTTP Response consists of:

1. **Status Line:**
   - HTTP Version (e.g., HTTP/2)
   - Status Code (e.g., 200)
   - Status Message (e.g., OK)
2. **Headers:**
   - Key-value pairs providing metadata (e.g., Content-Type, Server).
3. **Body (optional):**
   - Data returned by the server (e.g., HTML, JSON).

Example of an HTTP Response

```
HTTP/2 200 OK
Content-Type: application/json
Server: cloudflare
Content-Length: 1820
[
  {"albumId":1,"id":1,"title":"...","url":"...","thumbnailUrl": "..."},
  ...
]
```

- **Status Line:** `HTTP/2 200 OK`
- **Headers:** `Content-Type`, `Server`, `Content-Length`
- **Body:** JSON data on all albums with photos

HTTP Status Codes

- **1xx (Informational):** Request received, continuing process.
- **2xx (Success):** Request successfully received, understood, and accepted.
  - `200 OK`, `201 Created`, `204 No Content`
- **3xx (Redirection):** Further action needed to complete the request.
  - `301 Moved Permanently`, `304 Not Modified`
- **4xx (Client Error):** Request contains bad syntax or cannot be fulfilled.
  - `400 Bad Request`, `404 Not Found`, `403 Forbidden`
- **5xx (Server Error):** Server failed to fulfill a valid request.
  - `500 Internal Server Error`, `503 Service Unavailable`

Common HTTP Headers

- **Request Headers:**
  - `Host`: Domain name of the server.
  - `User-Agent`: Client software details.
  - `Authorization`: Credentials for authentication.
  - `Content-Type`: Type of data in the body (e.g., `application/json`).
- **Response Headers:**
  - `Content-Type`: Type of data in the body.
  - `Server`: Software used by the server.
  - `Set-Cookie`: Sets a cookie in the client.
  - `Cache-Control`: Cache the data for optimization

Useful links

- HTTP Request Methods (MDN Docs)
- HTTP Response Status Codes (MDN Docs)