🍀

HTTP Headers

HTTP headers are critical for communication between clients and servers, and developers should be familiar with a wide range of them to build and debug web applications effectively. Here's a categorized list of important headers that developers should know:

General Headers

Used in both requests and responses:

1. **`Cache-Control`**:
    - Controls caching behavior.
    - Examples:
        - `Cache-Control: no-cache` (forces validation with the server).
        - `Cache-Control: max-age=3600` (cache for 3600 seconds).
    - Common in performance optimization.
2. **`Content-Type`**:
    - Specifies the media type of the request/response body.
    - Examples:
        - `Content-Type: application/json`
        - `Content-Type: text/html; charset=UTF-8`
3. **`Content-Length`**:
    - Indicates the size of the request/response body in bytes.
    - Helps the client know when the body ends.
4. **`Content-Encoding`**:

- Specifies compression methods applied to the body.
- Example: `Content-Encoding: gzip`

5. `Accept`:
    - Informs the server about acceptable response media types.
    - Example: `Accept: application/json`
6. `Accept-Encoding`:
    - Indicates acceptable compression methods for the response.
    - Example: `Accept-Encoding: gzip, deflate, br`
7. `User-Agent`:
    - Contains information about the client application (e.g., browser, version).
    - Example: `User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)`

Request Headers

Used in client-to-server communication:

1. `Authorization`:
    - Used for authentication.
    - Examples:
        - `Authorization: Bearer <token>` (OAuth token).
        - `Authorization: Basic <base64-encoded-credentials>`
2. `Host`:
    - Specifies the domain name of the server (required in HTTP/1.1).
    - Example: `Host: www.example.com`
3. `Referer` (or `Referrer`):
    - Identifies the URL of the page that referred the request.
    - Example: `Referer: https://www.example.com`
4. `Origin`:

   - Specifies the origin (scheme, host, and port) of the request.
   - Important for Cross-Origin Resource Sharing (CORS).

5. `Cookie`:
   - Sends cookies from the client to the server.
   - Example: `Cookie: session_id=abc123`

6. `X-Requested-With`:
   - Often used in AJAX requests to identify the request as originating from JavaScript.
   - Example: `X-Requested-With: XMLHttpRequest`

Response Headers

Used in server-to-client communication:

1. `Set-Cookie`:
   - Sets a cookie on the client.
   - Example:

     ```
     vbnet
     Copy code
     Set-Cookie: session_id=abc123; HttpOnly; Secure; SameSite=Strict
     ```

2. `Access-Control-Allow-Origin`:
   - Specifies allowed origins for CORS.
   - Example: `Access-Control-Allow-Origin: *`

3. `ETag`:
   - Provides a unique identifier for the response content, used for caching validation.
   - Example: `ETag: "abc123"`

4. `Location`:
    - Indicates the URL for redirection.
    - Example: `Location: https://www.example.com/login`
5. `Content-Disposition`:
    - Suggests how the content should be handled (e.g., as an attachment).
    - Example:
        - `Content-Disposition: inline`
        - `Content-Disposition: attachment; filename="file.pdf"`
6. `Retry-After`:
    - Suggests a time to retry the request, typically after a `503 Service Unavailable`.
    - Example: `Retry-After: 120` (retry after 120 seconds).

Security Headers

These headers are essential for protecting web applications:

1. `Strict-Transport-Security` **(HSTS)**:
    - Enforces HTTPS connections.
    - Example: `Strict-Transport-Security: max-age=31536000; includeSubDomains`
2. `Content-Security-Policy` **(CSP)**:
    - Defines allowed sources for scripts, styles, etc., to mitigate XSS attacks.
    - Example:

      ```
      arduino
      Copy code
      Content-Security-Policy: default-src 'self'; script-src 'self' https://apis.example.com
      ```
3. `X-Content-Type-Options`:

  - Prevents browsers from guessing MIME types.
  - Example: `X-Content-Type-Options: nosniff`

4. **`X-Frame-Options`**:
   - Controls whether a page can be displayed in a frame to prevent clickjacking.
   - Example: `X-Frame-Options: DENY`
5. **`X-XSS-Protection`**:
   - Enables cross-site scripting filters in older browsers.
   - Example: `X-XSS-Protection: 1; mode=block`

Caching Headers

1. **`Expires`**:
   - Specifies when the content expires (absolute date/time).
   - Example: `Expires: Tue, 10 Jan 2025 15:00:00 GMT`
2. **`Last-Modified`**:
   - Indicates the last modification date of the resource.
   - Example: `Last-Modified: Mon, 04 Jan 2025 12:00:00 GMT`
3. **`Vary`**:
   - Specifies which request headers affect the cached response.
   - Example: `Vary: Accept-Encoding`

Debugging Headers

1. **`X-Debug-Token`** / **`X-Debug-Token-Link`**:
   - Used for debugging and profiling in development environments.
2. **`X-Powered-By`**:
   - Indicates the technology used by the server.
   - Example: `X-Powered-By: Express`