

Dear IT/CS 312 Prelim Teams,

To prepare for your upcoming prelim practical examinations for IT/CS 312L, please carefully read and practice the following key HTTP concepts and features.

- As part of the practical activities, you are required to create and configure either an `.htaccess` or `httpd-vhosts.conf` file in your local server environment. The directives you include in this file will be used to apply and modify HTTP header settings such as authentication, cache control, compression, content negotiation, conditional requests, and range requests. Your instructor will ask you to demonstrate your understanding and ability to modify these directives during the lab examination.
- Read on the Apache HTTP server directives related to the abovementioned features and practice editing the `.htaccess` / `httpd-vhosts.conf` file and testing your changes using tools like curl, Postman, or browser developer tools.
- Prepare the web resources that you will utilize such as dummy HTML files, images, pdfs, xml files and your corresponding `.htaccess` / `httpd-vhosts.conf`. These features will be demonstrated via GET requests only and will not require any scripts.

1. Authentication

- Understand how HTTP authentication works (Basic Auth).
- Practice sending the `Authorization` header with valid credentials.
- You should be able to demonstrate how to change these credentials.

2. Cache-Control

- Practice setting and modifying `Cache-Control` headers like:
 - `no-cache`
 - `no-store`
 - `public, max-age`
- Observe how these affect browser or client caching.
- Use browser dev tools to inspect caching behavior.

3. Compression

- Learn how to enable and test response compression (e.g. gzip).
- Practice sending `Accept-Encoding` headers from the client and verify that the server responds with compressed content using `Content-Encoding`.
- Verify that the response body was compressed and deflated accordingly. For those using browser dev tools, explore how this can be done.

4. Conditional Requests

- Understand `If-Modified-Since` and `If-None-Match` (ETag) headers.
- Practice returning `304 Not Modified` when content hasn't changed.
- Understand how `Last-Modified` and `ETag` headers get included in server responses.

5. Content Negotiation

- Practice handling `Accept` headers to return different content types (e.g., JSON, HTML).
- Be prepared to switch response formats based on the client's request.

6. Range Requests

- Learn how to enable byte range requests by setting the `Accept-Ranges` header.