

**Name: Ibrahim Adamu**

**Course: Web Technology.**

**Course code: CSC 105.**

**Date: 12th of October, 2023.**

## **1. Research on HTTP and HTTPS, stating what each status code represents and when we can possibly have each.**

**HTTP** (Hypertext Transfer Protocol) is a protocol that defines how messages are formatted and transmitted between a web browser and a web server. It is the foundation of the World Wide Web and is used to deliver web pages, images, videos, and other resources to users.

**HTTPS** (Hypertext Transfer Protocol Secure) is a secure version of HTTP that uses Transport Layer Security (TLS) to encrypt all traffic between the client and server.

**HTTP status codes** are three-digit numbers that are returned by a web server to a client in response to a request. They indicate whether the request was successful or not, and may provide additional information about the error that occurred.

Here are some examples of common HTTP status codes and when they might be used:

- **200 OK:** This is the most common HTTP status code and indicates that the request was successful and the server has returned the requested content.
- **301 Moved Permanently:** This code indicates that the requested resource has been moved to a new location. The client should automatically redirect to the new location.
- **403 Forbidden:** This code indicates that the client does not have permission to access the requested resource.
- **404 Not Found:** This code indicates that the requested resource does not exist on the server.

- **500 Internal Server Error:** This code indicates that an unknown error occurred on the server while processing the request.

#### **When we can possibly have each HTTP status code:**

- **100-199 Informational:** These codes are typically used when the server is processing a large request and needs to send the client updates on the progress.
- **200-299 Successful:** These codes are returned whenever the server is able to successfully process the client's request.
- **300-399 Redirections:** These codes are returned when the server needs the client to take further action to complete the request. For example, a server might return a 301 Moved Permanently code if the requested resource has been moved to a new location.
- **400-499 Client Errors:** These codes are returned when the client makes a bad request. For example, a server might return a 404 Not Found code if the client requests a resource that does not exist.
- **500-599 Server Errors:** These codes are returned when the server encounters an unexpected error while processing the request. For example, a server might return a 500 Internal Server Error code if the database is unavailable.

## **2. Differentiate after a good explanation on Response and Request.**

### **Request**

A request is a message sent by a client to a server requesting specific information or action. The request contains information about the requested resource, such as its URL, and any other necessary data, such as authentication credentials or request parameters.

### **Response**

A response is the message sent by the server in response to a request from a client. The response contains the requested information or the result of the requested action. The

response also contains status information about the request, such as whether it was successful or not.

Differences between requests and responses:

Requests:

- Are initiated by the client.
- Can contain a variety of HTTP methods, such as GET, POST, PUT, and DELETE.
- Can contain headers with additional information about the request, such as the requested resource, the client's capabilities, and authentication credentials.
- Can contain a body with data to be sent to the server, such as a form submission or a file upload.

Responses:

- Are sent by the server in response to a request from the client.
- Always contain a status code indicating the success or failure of the request.
- Can contain headers with additional information about the response, such as the content type, length, and caching instructions.
- Can contain a body with the requested data, the results of an action, or an error message.

### **3. Write short notes on HTML and Tags.**

#### **HTML and Tags**

HTML, or HyperText Markup Language, is the standard markup language for creating web pages. It uses tags to define the structure and content of a web page. Tags are enclosed in angle brackets (< >) and come in pairs, with an opening tag and a closing tag. For example, the opening tag for a paragraph is <p> and the closing tag is </p>.

HTML tags can also have attributes, which provide additional information about the tag. For example, the <img> tag has a src attribute, which specifies the URL of the image . When a web browser encounters an HTML document, it interprets the tags and displays the content of the document accordingly. For example, if the browser encounters the tag <p>This is a paragraph of text.</p>, it will display the text "This is a paragraph of text." on the screen.

HTML tags are essential for creating web pages. By using tags to define the structure and content of a web page, you can create pages that are informative, engaging, and easy to navigate.

#### **4. Write short notes on client and server and types of server.**

##### **Client and Server**

Clients and servers are two types of computer programs that work together to provide services to users. A **client** is a program that runs on a user's computer and sends requests to a server. A **server** is a program that runs on a remote computer and processes requests from clients. Client-server architecture is a common way to design computer networks. It is used in a wide variety of applications, including web browsing, email, file sharing, and database access.

##### **Types of Servers**

There are many different types of servers, each with its own specific purpose. Some of the most common types of servers include:

- Web servers: Web servers deliver web pages to users' web browsers.
- Email servers: Email servers send and receive email messages.
- File servers: File servers store and share files with users.
- Database servers: Database servers store and manage databases.
- Application servers: Application servers host and run web applications.
- Proxy servers: Proxy servers act as intermediaries between clients and other servers.
- DNS servers: DNS servers translate domain names into IP addresses.