Week 14 - Wednesday HTTP Worksheet

**Problem 1**

* The 5 categories of response codes are informational(100s), successful(200s), redirection(300s), client side errors(400s), and server side errors(500s).

**Problem 2**

|  |  |  |
| --- | --- | --- |
| **HTTP Response Status Code** | **HTTP Response Status Message** | **Description** |
| **100** | Continue | Indicates that the client should continue the request or ignore the response if the request is already finished |
| **102** | Processing | Indicates the server has received request, is processing it, but no response is ready yet. |
| **103** | Early Hints | Allows user to preload a page while waiting on response from the server. |
| **200** | OK | Indicates a request has been successfully carried out and resulted in appropriate response depending on http method used. |
| **201** | Created | Indicates a request with either post or put method is successfully carried out creating a new resource or data. |
| **204** | No Content | Indicates there is no content available based on the response of the sent request. |
| **307** | Temporary Redirect | Indicates the user is being redirected to a temporary URI, while the original is perhaps being worked on, and the original is still valid. |
| **308** | Permanent Redirect | Indicates the user is being redirected to a new permanent URI which the user can directly access for resources. |
| **400** | Bad Request | Indicates the client’s request to server is not valid and does not meet requirement such as syntax. |
| **404** | Not Found | Indicates the URL in the client’s request cannot be found, does not exist, and/or is invalid for some other reason. |
| **418** | I’m a Teapot | This is a funny April fool’s joke, and the result of this joke became a permanent http status code. |
| **500** | Internal Server Error | This indicates the server has an internal error that it does not know how to handle. |
| **501** | Not Implemented | This indicates the server does not support the http method being requested. The only methods required by all servers are get and head. |

**Problem 3**

Regarding HTTP methods, idempotent means that the method’s result does not yield in a different outcome or result each time the method is called. Idempotent methods should provide consistent results, and the state of the server is consistent.

**Problem 4**

|  |  |  |
| --- | --- | --- |
| **HTTP Request Method** | **Safe** | **Idempotent** |
| **GET** | yes | yes |
| **PUT** | no | yes |
| **POST** | no | no |
| **DELETE** | no | yes |
| **OPTIONS** | yes | yes |
| **TRACE** | yes | yes |
| **PATCH** | no | no |
| **HEAD** | yes | yes |