The HTTP relies on the "client-server" architecture that is built on the basis of messaging. HTTP messages are a way to exchange data between clients and servers on the Web. There are two types of messages: **requests** and **responses**.

A **request** is an operation that a client wants to perform on the server, and a **response** is an answer from the server to an incoming request. Usually, programmers do not need to worry about creating HTTP messages since they are produced by browsers, applications, and web servers.

**The format of messages**

In the HTTP protocol, all messages consist of text strings. Both requests and responses have roughly the same standardized format:

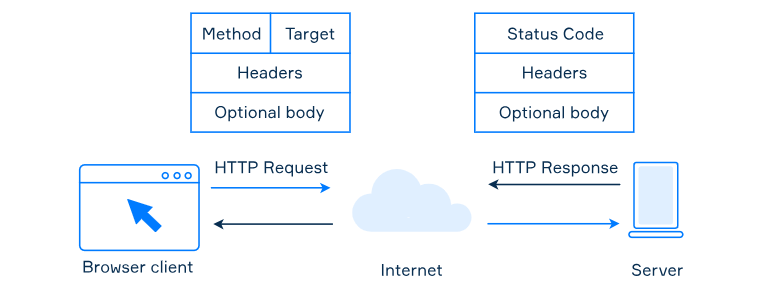
1. **Start line** which may vary:
   * for requests, it indicates the type of request (**method**) and the URL where to send it (**target**);
   * for responses, it contains a status code to determine the success of the operation.
2. **Headers** that describe the message and convey various parameters.
3. **Body** in which the message data is located.

The **start line** and the **header** are required attributes, so the other parts may be empty.

The full format can be quite complicated for beginners, so we have considered only its part which is the most important for understanding the general principles.

**The simplified HTTP interaction**

Here is a simplified HTTP interaction between a browser client and a server. The client and the server interact through requests and responses which follow the studied format:



Note that there are other possible types of client programs, not just a browser. You can even write your own HTTP client and interact with servers. The only requirement is that such a program should always follow the message format.

**Methods**

HTTP defines a set of request methods that specify what the desired action will be for a given resource. Despite the fact that their names can be nouns, these query methods are sometimes referred to as *HTTP verbs*.

Let's look at the most commonly used query methods:

* GET method is only used to retrieve data from the server;
* POST method is used to send data to the server;
* HEAD requests data from the server in the same way as the GET method, but without a response body.

Every time you click on a link, you basically communicate to your browser that you want to GET this page. When you want to log in to your favorite site, you POST your login and password to receive access to it.

There are [more available verbs to learn](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods). You don't need to memorize them all right now.

**Status codes**

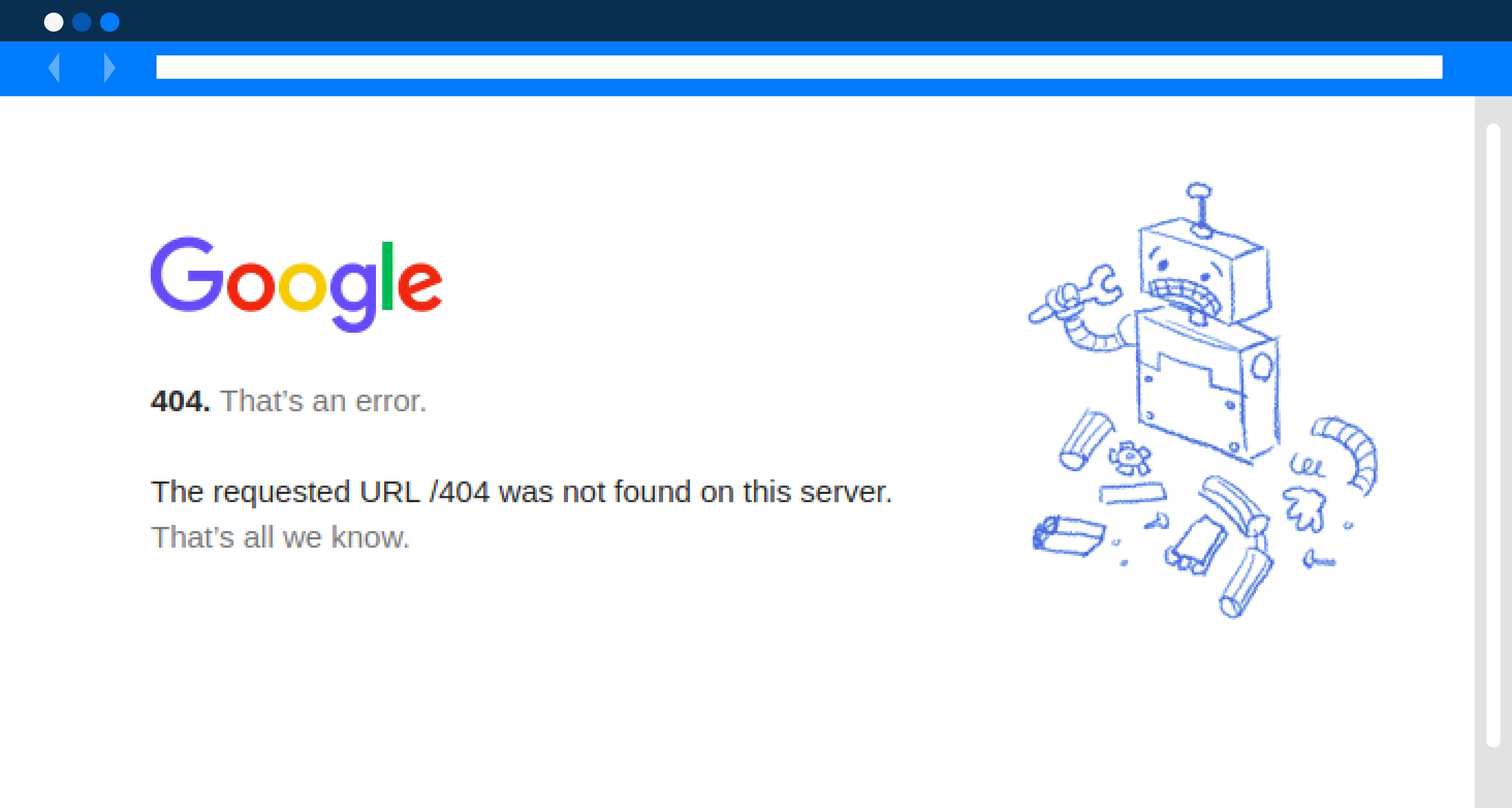
For normal operation of computer programs and web pages that work via HTTP, besides the content of the page, the server returns a three-digit code, which specifies the response for the request. With the help of this code, it is possible to redirect the client to another site or to indicate the change of the page, as well as detect an error in the data processing.

Currently, the standards define five classes of status codes:

|  |  |
| --- | --- |
| **1xx: Informational** | Codes beginning with "1" are called information codes. They report on how client requests are processed. |
| **2xx: Success** | Messages of this class inform that the action requested by the client has been successfully accepted for processing. |
| **3xx: Redirection** | It means further action must be taken in order to complete the request. |
| **4xx: Client Error** | It reports errors on the client's side. |
| **5xx: Server Error** | The code indicates that the operation was unsuccessful due to the fault of the server. |

As an example, if you have successfully loaded a website, the response you received has code 200. You can check this by opening the developer tools of your browser, and clicking on the Networking tab. Then try reloading the web page and you will see the status codes. The combination of keys to open the developer tools can vary. To give you an example, this might be Ctrl + Shift + I or F12 on Windows and Linux, or Cmd + Opt + I on macOS.

You have also probably been in a situation where your browser displays the **"404 Not Found"** message when you input the address of a page that does not exist. This is how these failure messages usually look:



Browsers display error messages so that users can understand that something has gone wrong, rather than continuing to look at the blank page while waiting for the content to be downloaded.

**Conclusion**

Let's highlight the main points we've just discussed here:

* HTTP messages can be of two types: requests and responses.
* They are composed of the start line, headers, and body. The start line in requests includes method and target, while in responses it includes status code.
* The commonly used methods in request messages are GET, POST, and HEAD.
* Status code indicates the response from the server as a three-digit number. It can be one of 5 classes: Informational, Success, Redirection, Client Error, and Server Error.

Now, when you've finished reading the topic, you can visit various sites in a browser and try to guess what your actions look like from a technical point of view.

What does a message body contain? Data

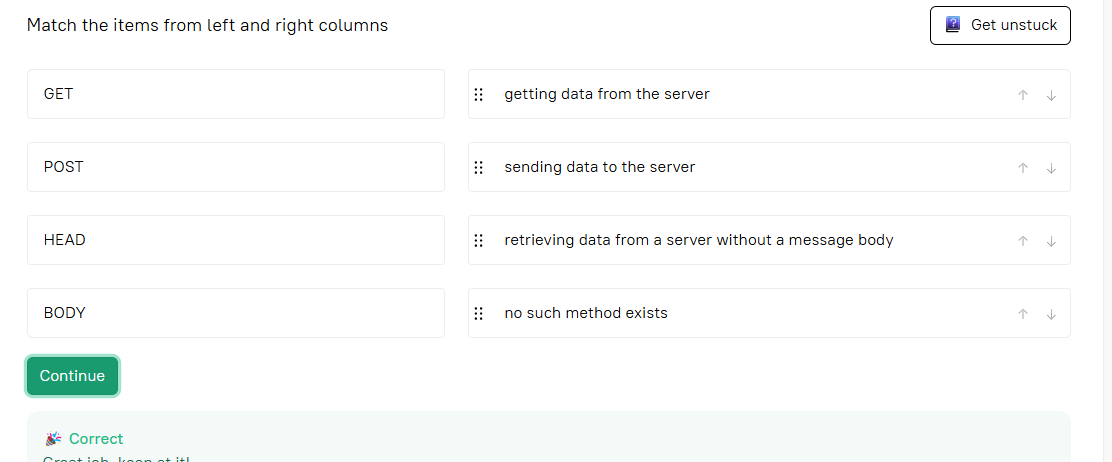
What status code do you think will be returned when you go to <https://www.google.com/page_not_found?404>

2xx What is the response code you received from the server by opening the page with this task?

Hint   
Press "F12" on your browser and click on network. That's all

200

Match the methods and their functionality.



Choose the two existing HTTP methods.

GET

POST

What class will the status code come in if the server crashes while processing the client request? 5xx

Match classes of status codes with their values.

 See hint

| **Code Class** | **Meaning** |
| --- | --- |
| **1xx** | it means the request has been received and the process is continuing |
| **2xx** | messages of this class inform that the action requested by the client has been successfully accepted for processing |
| **3xx** | it means further action must be taken in order to complete the request |
| **4xx** | it reports errors on the client's side |
| **5xx** | it reports server side errors |

What are the required parts in HTTP message syntax?

The **required parts** in HTTP message syntax are:

✅ **Start line**  
✅ **Headers**

### Optional:

* **Message Body** – Present in some requests/responses (e.g. POST requests, response with data).
* **Footer** – ❌ Not part of standard HTTP message syntax.

### ✅ Correct answers:

* **Start line**
* **Headers**

What HTTP method should be used if you want to implement the ability for users to send messages to the forum? Post

What are the types of HTTP messages?