



Professional Skills
Agile Fundamentals
Jira
Git
Databases Introduction
Java Beginner
Maven
Testing (Foundation)
Java Intermediate
HTML
CSS
Javascript
Spring Boot
Selenium
Sonarqube
Advanced Testing (Theory)
Cucumber
MongoDB
Express
NodeJS
React
Express-Testing
Networking
<div>○ Networking Basics</div>
<div>○ Networking Models</div>
<div>○ Routing and IP Addresses</div>
<div>○ Network Address Translation</div>
<div>○ HTTP Requests</div>
Security
Cloud Fundamentals

HTTP Requests

Contents

- [Overview](#)
- [What is HTTP?](#)
- [Request and Response](#)
- [Different kinds of requests](#)
- [Response Codes](#)
- [Tutorial](#)
 - [Prerequisites](#)
 - [Install](#)
 - [GET Request](#)
 - [POST Request](#)
- [Exercises](#)

Overview

In this module we will look at what HTTP is and different HTTP requests.

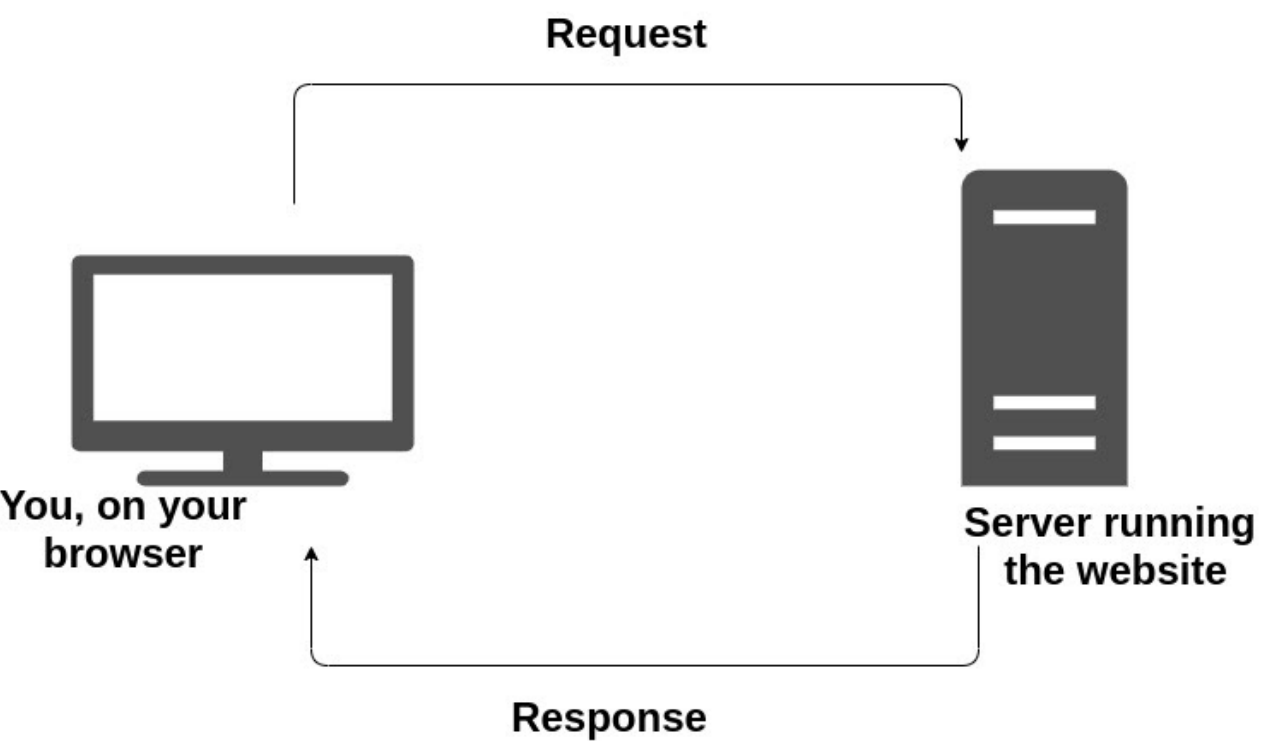
What is HTTP?

HTTP stands for HyperText Transfer Protocol.

This is the protocol used on the application layer. It is used over the internet and defines how messages are formatted and transferred. It defines how a webpage should act on a certain request.

Request and Response

HTTP is a request and response protocol. A request is made and the response is what you see on your webpage. Most of the time the request is clicking a button for a webpage to load and the response to that is the webpage you asked for.



A request is made up of:

- The HTTP Method
- A URL
- Parameters (For when you want to send data)

A response is made up of:

AWS Foundations
AWS Intermediate
Linux
DevOps
Jenkins Introduction
Jenkins Pipeline
Markdown
IDE Cheatsheet

- The status code
- The type of content (HTML, Image, Text)
- The content itself

Different kinds of requests

There are many types of requests. The most commonly used of which are **GET** and **POST**. We are going to look at these a couple more.

- GET
 - This request is the most common. A request is made and you get something in return. It's as simple as that. Whenever you load up any webpage, you are making a GET request.
- POST
 - This request is used to create new resources. You send a request with the data attached as parameters and then a response will tell us whether the addition of the data was successful
- PUT
 - This request is used to update existing resources.
- DELETE
 - This is used to delete existing data.

Response Codes

All responses come with a status code, which will tell us what happened to our request. Whether it was successful, it was redirected or failed.

The most common ones are 200 and 404.
A 200 status code means the request was successful. 404 means 'not found'.

Status codes follow this structure.

- 1XX - Information
- 2XX - Success
- 3XX - Redirected
- 4XX - Client Error (e.g. User attempting to visit a page that doesn't exist)
- 5XX - Server Error (e.g. A bug in the code for the webpage.)

Tutorial

In this tutorial we are going to use Postman which is a software development tool that allows you to make requests to websites and specify the type of request.
It will then show you the response.

Prerequisites

Postman can be installed as a Chrome extension. So you will need Google Chrome to complete this tutorial, or you can install Postman onto your computer.

Install

To install Postman on Chrome just go [here](#) and click on **Add To Chrome**.

Then click on the Postman app and make an account.

Close any messages that come up to see the main page.

GET Request

You should see the drop down to change the HTTP request method. For now leave this as GET.

Type in any website url and see what the response is e.g. google.com, and click **Send**.

You should see a **Status: 200 OK**, this is the status code from the response. You should also notice the big chunk of code that is the HTML file for the website your entered, this is the content of that website.

POST Request

Now change the HTTP method to POST.

We are going to use an API call httpbin to send POST requests to.

In the url section type in **httpbin.org/post**.

Then click the **Params** button to add some parameters. The parameters are formatted in keys and values. So set a key **message** and the value of that key to **helloworld**.

	Key	Value
<input checked="" type="checkbox"/>	message	helloworld

Notice that the url you entered has now changed to include your parameters.

Click **Send**.

You should see your response that includes your parameters and has a status code of **200 OK**.

Exercises

If I was to send a request to the same url but this time as a GET request, what status code should I get? Why?