HTTP
AJAX Requests
JSON

Asynchrony

Asynchronous Programming vs Parallel Programming

Asynchronous Code

In <u>synchronous</u> programs, if you have two lines of code (L1 followed by L2), then L2 cannot begin running until L1 has finished executing.

In <u>asynchronous</u> programs, you can have two lines of code (L1 followed by L2), where L1 schedules some task to be run in the future, but L2 runs before that task completes.
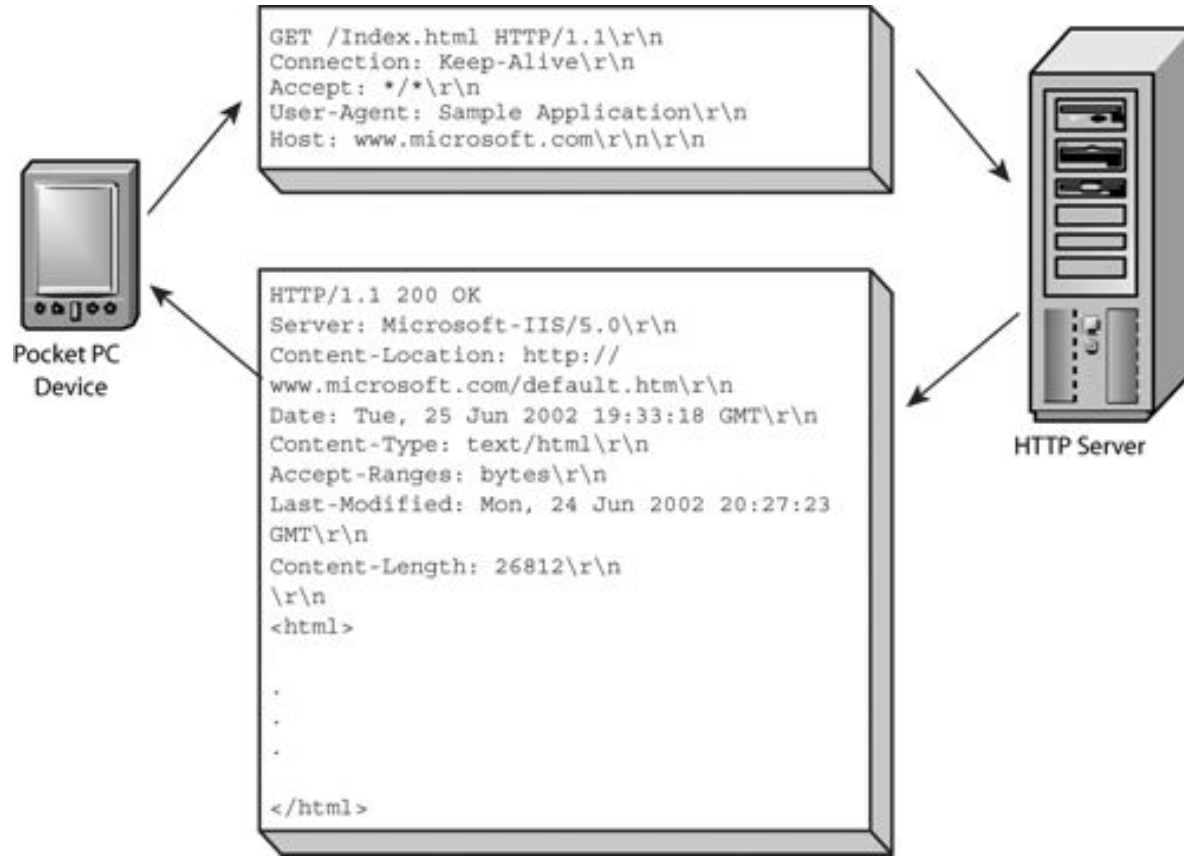
Callbacks

"Each function gets an argument which is another function that is called with a parameter that is the response of the previous action."

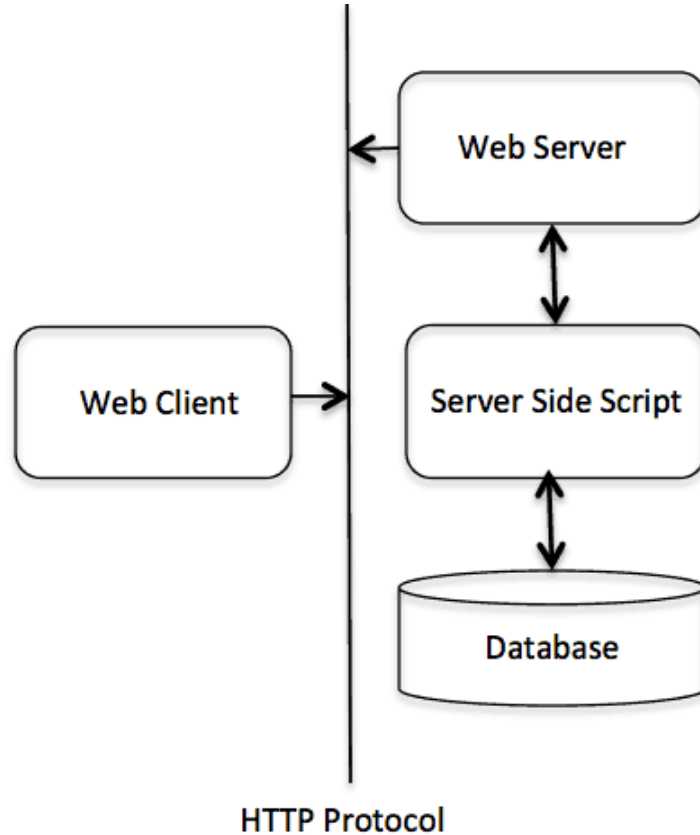# HTTP - Hypertext Transfer Protocol

HTTP is the foundation of data communication for the World Wide Web.

HTTP is the protocol that allows for sending documents back and forth on the web.

HTTP – Hypertext Transfer Protocol

```
GET /Index.html HTTP/1.1\r\n
Connection: Keep-Alive\r\n
Accept: */*\r\n
User-Agent: Sample Application\r\n
Host: www.microsoft.com\r\n\r\n
```

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0\r\n
Content-Location: http://
www.microsoft.com/default.htm\r\n
Date: Tue, 25 Jun 2002 19:33:18 GMT\r\n
Content-Type: text/html\r\n
Accept-Ranges: bytes\r\n
Last-Modified: Mon, 24 Jun 2002 20:27:23
GMT\r\n
Content-Length: 26812\r\n
\r\n
<html>

.
.
.

</html>
```

Pocket PC
Device

HTTP Server

HTTP – Hypertext Transfer Protocol

HTTP Protocol

HTTP – Hypertext Transfer Protocol

HTTP is based on the client-server architecture model and a stateless request/response protocol that operates by exchanging messages across a reliable TCP/IP connection.

HTTP Request

An HTTP client sends an HTTP request to a server in the form of a request message.

The request method indicates the method to be performed on the resource identified by the given Request-URI

HTTP Request - Methods

GET

The GET method is used to retrieve information from the given server using a given URI. Requests using GET should only retrieve data and should have no other effect on the data.

POST

A POST request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms.

HTTP Response

After receiving and interpreting a request message, a server responds with an HTTP response message.

HTTP Response - Status

1	**1xx: Informational** It means the request was received and the process is continuing.
2	**2xx: Success** It means the action was successfully received, understood, and accepted.
3	**3xx: Redirection** It means further action must be taken in order to complete the request.
4	**4xx: Client Error** It means the request contains incorrect syntax or cannot be fulfilled.
5	**5xx: Server Error** It means the server failed to fulfill an apparently valid request.

JSON - JavaScript Object Notation

A *language-agnostic* lightweight data-interchange *format*

JSON – JavaScript Object Notation

JSON is built on two structures:

- A collection of name/value pairs. In various languages, this is realized as an object, record, struct, dictionary, hash table, keyed list, or associative array.

- An ordered list of values. In most languages, this is realized as an array, vector, list, or sequence.

XHTTP Request - Get Example

```javascript
var xmlHttp = new XMLHttpRequest();
    xmlHttp.onreadystatechange = function() {
        if (xmlHttp.readyState == 4 && xmlHttp.status == 200)
            console.log(xmlHttp.responseText);
    }
    xmlHttp.open("GET", 'https://ghibliapi.herokuapp.com/films/58611129-2dbc-4a81-a72f-77ddfc1b1b49', true); // true for asynchronous
    xmlHttp.send(null);
```

Fetch API - GET Example

```
fetch('https://ghibliapi.herokuapp.com/films/58611129-2dbc-4a81-a72f-77ddfc1b1b49', {
    method: 'get'
})
    .then(function (response) {
        return response.json()
            .then(
                function (data) {
                    console.log(data);
                }
            )
}).catch(function (err) {
    // Error :(
    console.log('bebebe');
});
```

Public APIs

https://github.com/toddmotto/public-apis