Servlets and JSP

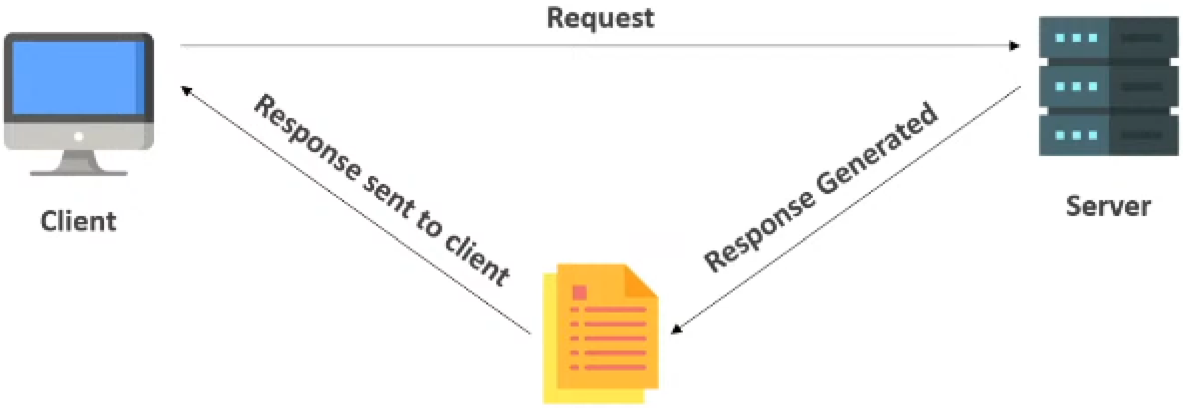
Outline

- Overview of Web Application
- HTTP Methods and Responses
- Lifecycle of Web Servlets
- Writing Servlet programs with Servlet APIs
- Reading and Processing Forms
- Handling GET/POST Requests
- Database connectivity through servlets
- Cookies and Sessions

Web Application

- Application is a large program or we can also say a collection of functions.
- A web application is a computer program that utilizes web browsers and web technology to perform tasks over the internet.
- Web applications are usually coded in browser-supported language such as JavaScript and HTML as these languages rely on the browser to render the program executable.
- The web application requires a web server to manage requests from the client, an application server to perform the tasks requested, and, sometimes a database to server the information.

Websites generally operate on a ‘**request-response**’ basis



Benefits of Web Application

- Web applications run on multiple platforms regardless of OS or device as long as the browser is compatible.
- All users access the same version, eliminating any compatibility issues.
- They are not installed on the hard drive, thus eliminating space limitations.
- They reduce software piracy in subscription-based web applications(i.e. SaaS)
- They reduce costs for both the business and end user as there is less support and maintenance required by the business and lower requirements for the end user’s computer.

The HTTP(Hypertext Transfer Protocol) Protocol

- The HTTP is an application layer protocol that allows web-based applications to communicate and exchange data using request/response cycle.
- The HTTP is the messenger of the web.
- It is a TCP/IP based protocol.

HTTP Methods

- The primary or most commonly-used HTTP methods are POST, GET, PUT and DELETE.
- These methods correspond to create, read, update and delete(or CRUD) operations, respectively.
- There are a number of other methods too, but they are utilized less frequently.

GET

- The HTTP GET method is used to read(or retrieve) a representation of a resource.
- GET requests are used only to read data and not change it. So, they are considered safe.
- Example: jsonplaceholder.typicode.com/comments?postId=1
- Browsers in default make GET request.
- You can easily test GET request using browser.

Post

- The POST method submits an entity to the specified resource, often causing a change in start or side effects on the server.

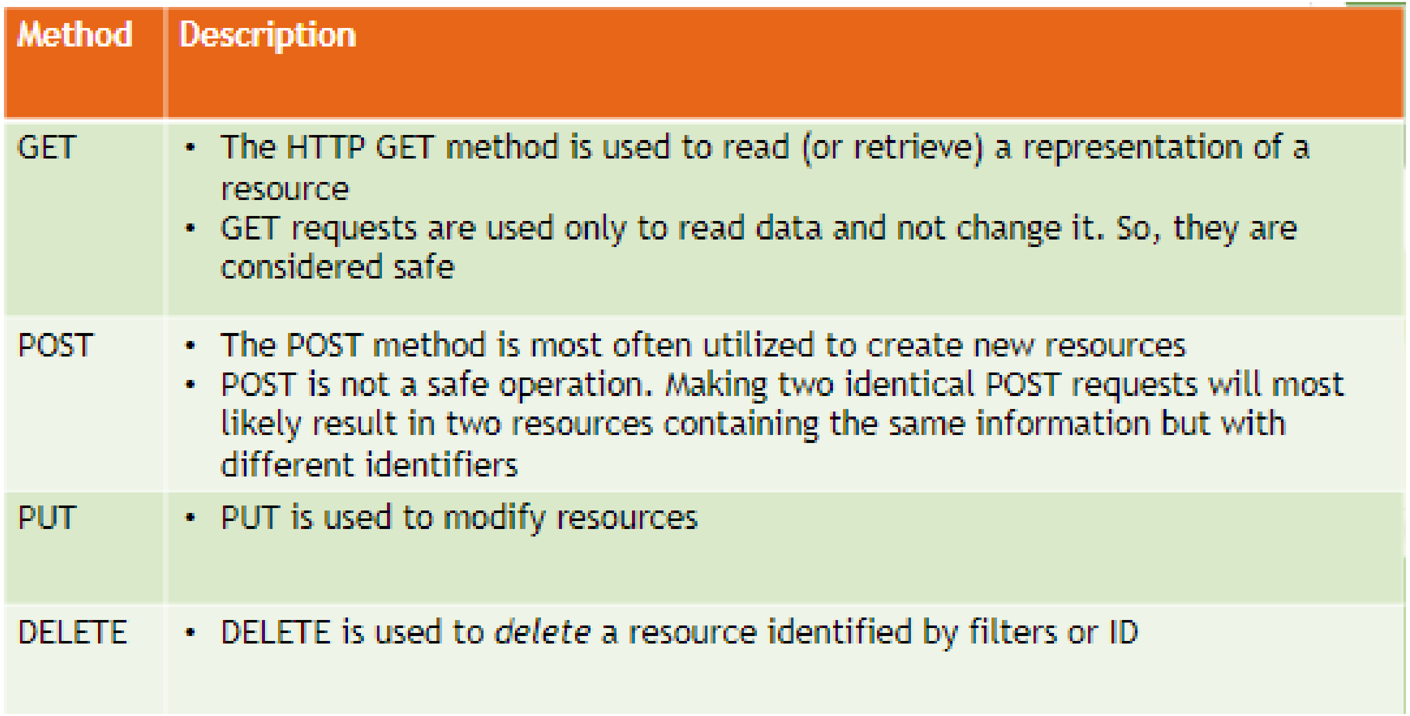
DELETE

- If a DELETE method is successfully applied, there are several response status codes possible:

PUT

- The HTTP PUT request method creates a new resource or replaces a representation of the target resource with the request payload.
- The difference between PUT and POST is that PUT is idempotent: calling it once or several times successively has the same effect(that is no side effect), whereas successive identical POST requests may have additional effects, akin to placing an order several times.

HTTP Methods



Method	Description
GET	• The HTTP GET method is used to read (or retrieve) a representation of a resource • GET requests are used only to read data and not change it. So, they are considered safe
POST	• The POST method is most often utilized to create new resources • POST is not a safe operation. Making two identical POST requests will most likely result in two resources containing the same information but with different identifiers
PUT	• PUT is used to modify resources
DELETE	• DELETE is used to *delete* a resource identified by filters or ID