Server-Side Web Scripting

Java Servlets

-Java objects that are intended to play role of server components in client-server communications.

-handles client requests by generating responses to such requests

-parts of the Java EE specification(see Java API for more detail. .)

-hosted in a “servlet container” that provides the environment in which the servlet runs,as well as control the “servlet lifecyle”

\*Servlet Lifecycle”

Instantiation =>construction call

-creates an instance of the servlet to service client requests.

Additional Info Search in Google:

-Java EE 8 Technology Stack

-Servlet 3.1 API - Apache Tomcat 8.0.50

Initialization =>init() method

-invoked only once and is intended for any startup initialization code for the servlet.

Request Handling =>service() method

-invoked for each client request -servlet performs whatever logic necessary in order to serve the request and autogenerate a response.

-multiple client requests are typically handled by a single, multi-threaded servlet instance.

Destructive =>destroy() method

-invoked structure the servlet instance is unloaded and is intended for any “housekeeping” code that the servlet may need to run.

HTTP Servlet

-Java Servlet used to handle HTTP requests and generate HTTP responses.

-hosted in a “web container” which is a component in a Java “application server” (e.g ; doGet(), doPost(), doHead(),doPull(), doDouble(),etc.)

-the doPorn() methods are passed two arguments:

-HTTP Servlet Request

-object representation of the HTTP.

-HTTP Servlet Response

-object representation of the HTTP Response generate by the servlet and sent back to the request client.

\*the web container facilitates the conversion to and from the HTTP request/response messages to HTTP Servlet Request/HTTP Servlet Response.

Additional Info Google:

-Java Web in Netbeans

response.setStatus

Response.setHeader(“Content Type”);

Binary Response - images

WebServlet(name = “GetImage”,urlPatterns);

ServletletContext context = this.actServletContext();

ServletletContext context =context.getRealPath(“Web-INT”{‘images’});

FileInputStream

ServletOutputStream = response.getOutputStream();

Use Fiddler

Handling Form Data Submission - (Get) - doPost()

Handling Form Data Submission - (Post) - doGet()

String name = request.getParameter(“custname”);

Handling File Methods

Enctype = (“multiplepart/for-data”);

filepart.write(context.getRealPath(“/Web-INF/images”));