**Query String:**

* Query String is a name & value string pair which passes information ONLY to Dynamic Resources such as Servlets & JSPs
* In URL, It’s an optional information and if present it starts with question mark followed by one or more name-value pair which are separated by an ampersnd(&)

Examples:

[www.google.com/search?q=Knight](http://www.google.com/search?q=Knight)

<http://localhost:8080/studentsApp/currentDate?fname=xyz&lname=sfd>

servlet Code to get Query String Information:-

String f = request.getParameter(“Fname”);

String l = request.getParameter(“lname”);

Where,

request = Object reference of HttpServletRequests

**Fragment ID**

* A fragment ID or Fragment Identifier, as the name implies, it refers to a particular section within a web page
* In URL, It’s an optional information & if present, it begins with a hash (#) character followed by an identifier.

Example:

<http://tomcat.apache.org/tomcat-6.0-doc/manager-howto.html#SessionStatistics>

Key elements of HTTP Requests are:

1. URL
2. Form Data (if any)
3. HTTP Method
4. Cookies (if any)

Key elements of HTTP Response are:

1. Status code
2. Content Type
3. Actual Content
4. Cookies (if any)

**Status Code:**

* Status code represents the status of HTTP Request For example,

Status Code: Description

200 Server successfully handled the request

404 Requested Resource (static/dynamic) is not found at server side

500 Server encountered an unexpected condition which prevented it from fulfilling the request.

* It’s a Mandatory information & it will be present in Header of HTTP Response
* Generally Webserver provides “Status Code” info in Http Response

**Content Type OR Multipurpose Internet Mail Extensions (MIME)**

* Content Type OR Multipurpose Internet Mail Extensions (MIME) Type, tells the browser that what type of content it’s going to receive so that it can prepare itself to handle the response data
* For example,
* Open an Adobe Reader to handle PDF content
* Open Media Player to handle media content etc.
* It’s a mandatory information & it will be present in Header of HTTP Response
* The default content type is “text/html”
* Few Examples:

text/html

appplication/pdf

video/quicktime

Many more…

**Actual Content**:

* It’s a Mandatory information & it will be present in Body of HTTP Response
* In case of static resource content of the resource becomes the “Actual Content”
* In case of dynamic resource, content present in servlets/ JSP becomes the “Actual Content”
* In case of Error Scenarios webserver generates error information & it becomes the “Actual Content”.

**Request Components:**

**Web Url**

* Web application is a “Collection of Web Resources” & every web resource (static/dynamic) should have its unique address in the form of web URL
* Hence every request should consist of Web URL (Mandatory information) & it will be present in Header HTTP Request.

**Form Data:**

* Data collected using HTML form is called as Form Data
* i.e Whenever we make request by Submitting Form, Then ONLY HTTP Request will have Form Data
* Hence in HTTP Request it’s an Optional Information
* **If Present, it may be present in either Header or Body of HTTP Request which depends on HTTP Method Present in the Request.**

**HTTP Method**

* It’s a mandatory information present in the header of the HTTP Request
* HTTP Method is the first component in the HTTP Request header
* HTTP 1.0 had 3 mehods & in HTTP 1.1, Five new Methods got introduced so in total HTTP 1.1 has 8 different methods & every Http Request should consist of “ONE of the 8 HTTP Methods”

(Present in HTTP 1.1)

1. HEAD
2. TRACE
3. PUT
4. DELETE
5. OPTIONS
6. GET
7. POST
8. CONNECT

Code Word: htpp dog c

* Servlet API has “Default Implementation” for these methods (excluding CONNECT method)
* All these default implementations are present in the Servlet API Class by name “javax.servlet.http.HttpServlet”
* So whenever we create a Servlet by extending HttpServlet
* We can inherit all these default implementations

OR

* We can override all of them

OR

* We can override couple of them
* **Depending on the HTTP method present in the request, Webserver invokes the corresponding doXXX(HSR, HSR) method**
* In other words, HTTP method indicate the desired action to be performed on the Dynamic Web Resource.

i.e Servlet

|  |  |
| --- | --- |
| **Modifier and Type** | **Method and Description** |
| protected void | [**doDelete**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#doDelete%28javax.servlet.http.HttpServletRequest,%20javax.servlet.http.HttpServletResponse%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req, [HttpServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletResponse.html) resp)  Called by the server (via the service method) to allow a servlet to handle a DELETE request. |
| protected void | [**doGet**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#doGet%28javax.servlet.http.HttpServletRequest,%20javax.servlet.http.HttpServletResponse%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req, [HttpServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletResponse.html) resp)  Called by the server (via the service method) to allow a servlet to handle a GET request. |
| protected void | [**doHead**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#doHead%28javax.servlet.http.HttpServletRequest,%20javax.servlet.http.HttpServletResponse%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req, [HttpServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletResponse.html) resp)  Receives an HTTP HEAD request from the protected service method and handles the request. |
| protected void | [**doOptions**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#doOptions%28javax.servlet.http.HttpServletRequest,%20javax.servlet.http.HttpServletResponse%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req, [HttpServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletResponse.html) resp)  Called by the server (via the service method) to allow a servlet to handle a OPTIONS request. |
| protected void | [**doPost**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#doPost%28javax.servlet.http.HttpServletRequest,%20javax.servlet.http.HttpServletResponse%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req, [HttpServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletResponse.html) resp)  Called by the server (via the service method) to allow a servlet to handle a POST request. |
| protected void | [**doPut**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#doPut%28javax.servlet.http.HttpServletRequest,%20javax.servlet.http.HttpServletResponse%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req, [HttpServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletResponse.html) resp)  Called by the server (via the service method) to allow a servlet to handle a PUT request. |
| protected void | [**doTrace**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#doTrace%28javax.servlet.http.HttpServletRequest,%20javax.servlet.http.HttpServletResponse%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req, [HttpServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletResponse.html) resp)  Called by the server (via the service method) to allow a servlet to handle a TRACE request. |
| protected long | [**getLastModified**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#getLastModified%28javax.servlet.http.HttpServletRequest%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req)  Returns the time the HttpServletRequest object was last modified, in milliseconds since midnight January 1, 1970 GMT. |
| protected void | [**service**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#service%28javax.servlet.http.HttpServletRequest,%20javax.servlet.http.HttpServletResponse%29)([HttpServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletRequest.html) req, [HttpServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServletResponse.html) resp)  Receives standard HTTP requests from the public service method and dispatches them to the do*XXX* methods defined in this class. |
| Void | [**service**](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\http\HttpServlet.html#service%28javax.servlet.ServletRequest,%20javax.servlet.ServletResponse%29)([ServletRequest](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\ServletRequest.html) req, [ServletResponse](file:///E:\Ebooks\Programming\Java\Java%20EE%20docs\glassfish4\docs\api\javax\servlet\ServletResponse.html) res)  Dispatches client requests to the protected service method |

**POST Method**

* If this method present in the Request then Webserver invokes doPost(HSR, HSR) Method in that Servlet
* This method allows user to Post the data of unlimited size to the server using HTML FORM (i.e Form Data)
* Post requests have a body
* Hence data sent using the POST is present in the body of the HTTP Request.

**GET Method**

* **If this method present in the Request then Webserver invokes doGet(HSR, HSR) Method in that Servlet.**
* **This method allows us to get the data from server**
* **It’s a default method**
* **Get requests “Using the GET Method the form-data is present in the Header part of HTTP Request “in the form of Query String”**

**What determines whether Browser sends GET or POST requests:**

1. **Typing a URL in Browser makes request to contain GET method**
2. **Clicking on a Hyper link in Browser makes request to contain GET Method**
3. **Submitting the form with method= “get” form attribute in Browser makes request to contain Get method.**
4. **Submitting the form with method = “post” form attribute in Browser makes request to contain “POST” method.**
5. **Submitting the form with “No method form attribute declaration “ in Browser makes request to contain “GET” method.**