Servlet

Objectives

- Explain HTTP Basics
- Understand: What is Servlet, Why Servlet
- Exploring Servlet API
- Understand Life Cycle of Servlet
- Understand ServletConfig, ServletContext
- Explain: HTML Form Processing
- Differentiate between GET and POST Request
- Understand Collaboration
- Explain Session Management

HTTP

- HTTP stands for Hyper Text Transfer Protocol.
- HTTP is a stateless protocol or request-response protocol.
- Does not maintain any conversational state between the 2 requests.
- Cannot recognize the client.
- The most commonly used protocol in Web Application.

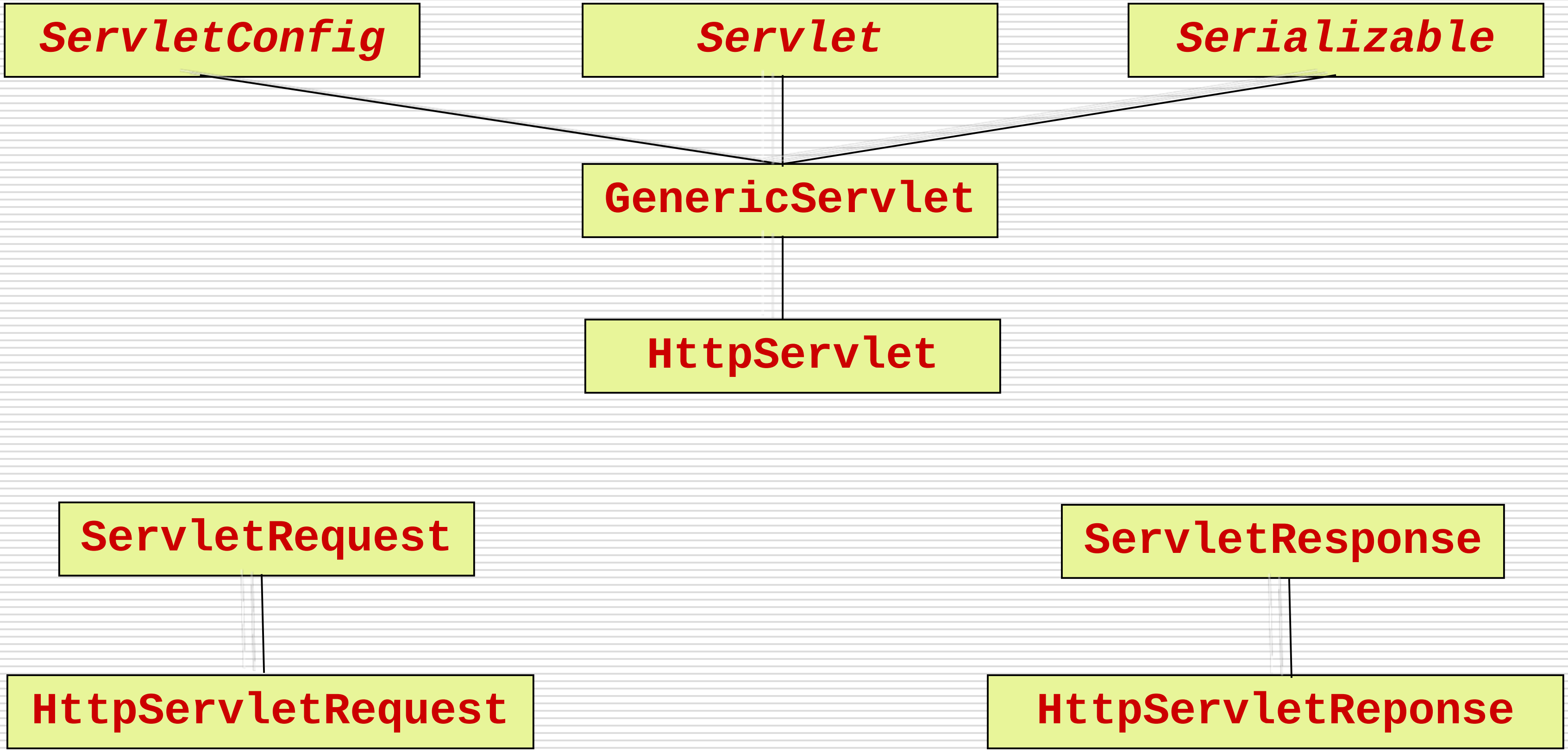
What is Servlet

- Servlet is a component that is used to extend the functionality of web server.
- A component that resides on server side and performs server side processing.
- Used to generate dynamic web contents.

Why Servlet

- Servlets are written in Java language, thus inherit all the features of Java.
- Portable
- Secured
- Platform Independent

Servlet API



ServletConfig

Servlet

Serializable

GenericServlet

HttpServlet

ServletRequest

ServletResponse

HttpServletRequest

HttpServletReponse

Servlet Life Cycle

- Life Cycle of Servlet consists of 3 stages:
  - Instantiation and Initialization
  - Service
  - Destroy
- There are 3 life cycle methods:
  - init()
  - service()
  - destroy()

ServletConfig

- An object of `ServletConfig` is associated with a servlet.
- Stores configuration specific information related to the servlet.
- Can be used to retrieve initial parameters.
- E.g.

```
String name;
name = getInitParameter(“name”);
```

ServletContext

- An object of `ServletContext` is created per application.
- Thus, useful to handle the application level information.
- Useful Methods:
  - `public void setAttribute(String, Object);`
  - `public Object getAttribute(String);`

HTML Form Processing

- In a web application, end user enters data using some HTML form.
- Once, SUBMIT is clicked, request is made to the server and it is to be processed by some server side component.
- E.g. User validation using Login page, User registration using registration page.
- This is done using `action` attribute of the HTML `<form>` element.

Difference between GET and POST

GET	POST
Request parameters are appended to URL.	Request parameters are sent with the page body.
Limitation on data transfer. Generally 8kb.	There is no limitation on data transfer.
Limitation on length of the URL: 255 characters	There is no limitation on URL length.

Collaboration

- When 2 components of same web application are interacting with each other, that process is known as collaboration.
- Benefits
  - Modularity
  - Reusability

RequestDispatcher

- Used to achieve collaboration between the components running within the same web application.
- Methods:
  - `public void forward(ServletRequest, ServletResponse);`
  - `public void include(ServletRequest, ServletResponse);`

HttpServletResponse: sendRedirect()

- It’s an alternative by which a control can be transferred from one component to another.
- E.g.

```
String url = http://www.google.com
response.sendRedirect(url);
```

Difference between forward() and sendRedirect()

forward()	sendRedirect()
Allows to pass the control from one web component to another running in same web application.	Allows to pass the control from one web component to another, even running in different web application.
Original request parameters are also propagated.	Original request parameters are not propagated.
Takes less time as control is just forwarded from one component to another.	Takes more time as it makes a round trip and generates a new fresh request.

Session Management

- HTTP is a stateless protocol.
- In a web application, an end user can make some transaction through one or multiple requests.
- During this, server needs to maintain a conversational state along with the client.
- This technique is known as session tracking.
- Different methods used for Session Tracking:
  - URL Rewriting
  - Hidden Fields
  - Cookies
  - Servlet API - HttpSession

URL Rewriting

- Incoming URL is rewritten by appending some additional information.
- E.g.

  ```
  <a href="/MyApp/myServlet?param=1">
     Click Here
  </a>
  ```

- Not suitable for large scale applications as URL's are always to be modified dynamically.

Hidden Fields

- Similar to URL Rewriting but data is sent to the server through hidden form fields.
- E.g.

```
<input type=‘hidden’ name=‘param’
value=‘1’/>
```

- Not suitable for large scale applications as hidden fields are always to be generated dynamically.

Cookies

- Cookie is a small text file that stores information in name-value pairs.
- Cookies are created on Server and stored on Client.
- Cookies are divided into 2 categories:
  - Transient Cookies
    - Reside in the browser's memory as long as browser window is opened.
  - Persistent Cookies
    - Permanently stored on the client machine until deleted explicitly.

Using Cookies

- `javax.servlet.http.Cookie`
- Attaching a cookie to the HTTP Response

  ```
  Cookie c1 = new Cookie(“name”,”Jack”);
  response.addCookie(c1);
  ```

- Retrieving cookies through HTTP Request

  ```
  Cookie ck[] = request.getCookies();
  ```

- Important Methods:
  - `getName()`
  - `getValue()`
  - `setMaxAge()`

HttpSession

- A Servlet API that is used to handle Session Tracking.
- `HttpServletRequest` is used to obtain the object of `HttpSession`.
  - `getSession()`
  - `getSession(boolean)`
- Important Methods:
  - `setAttribute()`
  - `getAttribute()`
  - `isNew()`
  - `setMaxInactiveInterval()`
  - `invalidate()`