

## Servlets

Java classes that extend the capabilities of a server by handling HTTP requests and responses.

- 1.) Client sends an HTTP request (think about everything we do on the internet; click, submit forms, enter URLs, etc).
- 2.) Server receives request and generates `HttpServletRequest` and `HttpServletResponse` java objects.
- 3.) Servlet container (aka, web container – in our case, Apache Tomcat) uses the deployment descriptor (`web.xml`) to map the request to the appropriate servlet.
- 4.) Servlet container instantiates the servlet.
- 5.) Container calls servlet's `init()` method (either upon 1<sup>st</sup> request to servlet, or on startup of the application based on the `<load-on-startup>` tag) to set any parameters specified in the D.D.
- 6.) Container calls servlet's `service()` method to process the request, which invokes a request handler method
- 7.) Http response is returned by servlet to server, then to the client.
- 8.) Each successive request to the server mapping to this servlet invokes `service()`.
- 9.) Container calls `destroy()` method (releasing instance to save memory; active period for servlet has elapsed, or container is shutting down).

```
public void service(ServletRequest req, ServletResponse res) { ... }  
    -- Called by web container, dispatches req/res pair  
    -- Converts servlet req/res to HttpServletRequest/Response  
protected void service(HttpServletRequest req, HttpServletResponse res) { ... }  
    -- Dispatches req/resp to appropriate request handler method, based on the type of  
    incoming request.
```

### Handler Methods

```
protected void doXXX(HttpServletRequest req, HttpServletResponse resp) throws  
ServletException, IOException { ... }  
    -- Handles various HTTP methods.
```

### PrintWriter

- PrintWriter pw = response.getWriter();
- Used to send character data in your response body

### Servlet Context

- Provides configuration details for all servlets in container.

### Servlet Config

- Provides configuration for one servlet.

Moving between resources: Redirect vs. Forward

## **Session Management**

What is a session? How can we manage them with servlets?

- Cookies, URL rewriting, hidden form fields, session objects

Cookies are small pieces of text stored in memory by their browser.