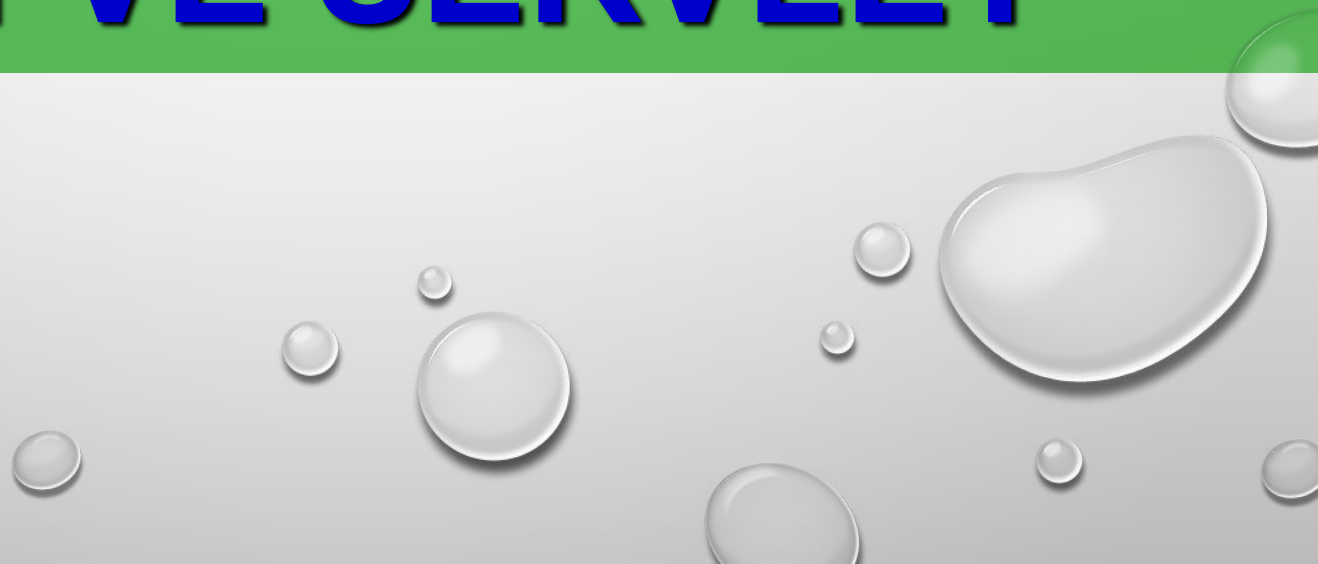


CHƯƠNG 13 – CƠ BẢN VỀ SERVLET



NỘI DUNG

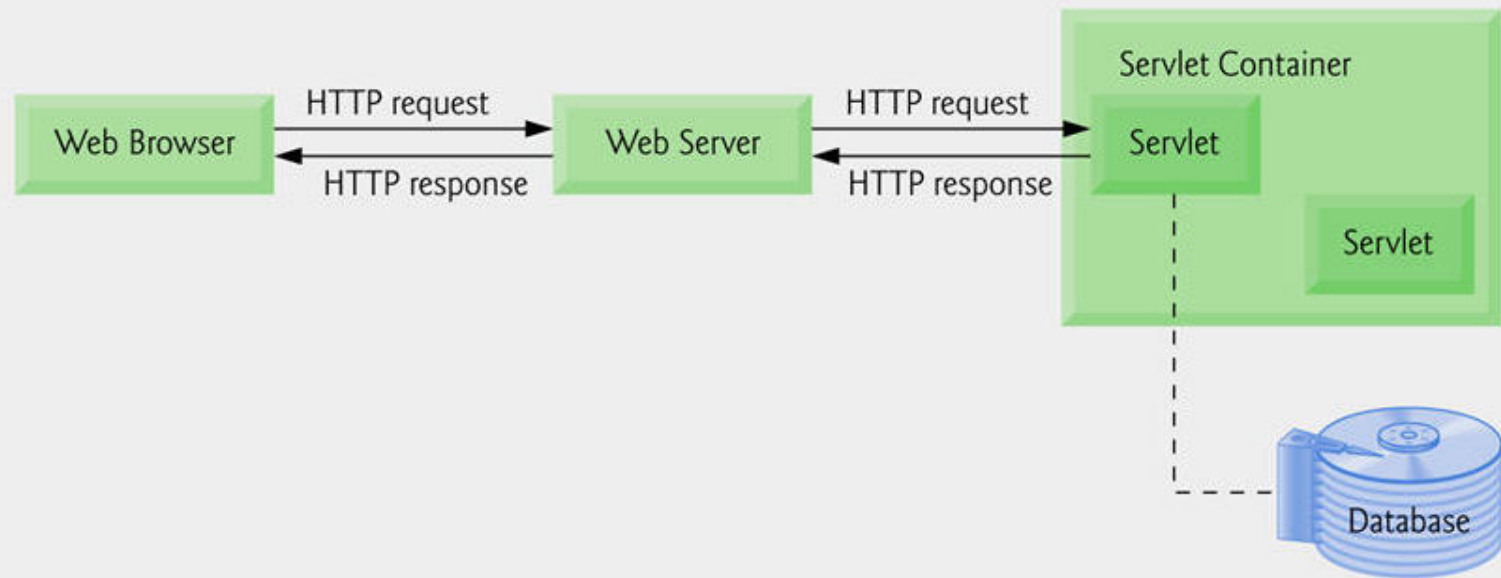
1. Giới thiệu về Servlet
2. Vòng đời của Servlet
3. Servlet Container
4. Servlet API
5. Cấu trúc và triển khai ứng dụng

1. GIỚI THIỆU VỀ SERVLET

GIỚI THIỆU VỀ SERVLET

- Servlet là gì ?
 - Servlet là một lớp Java ở phía server được thiết kế để xử lý những yêu cầu HTTP từ phía client và sinh ra nội dung động trả về cho client.
- Các đối tượng Java, mở rộng chức năng của 1 HTTP server.
- Được ánh xạ (mapped) với 1 URL và được quản lý bởi container tương ứng
- Chạy được trên tất cả các web servers và các app servers chuẩn

MÔ HÌNH SERVLET REQUEST RESPONSE

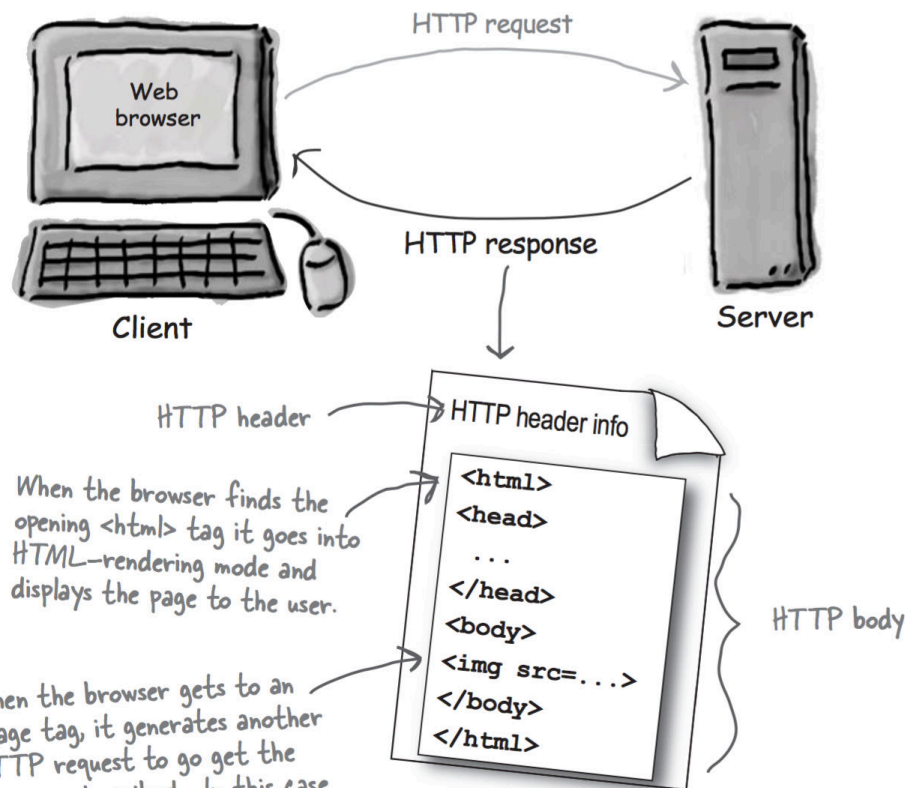


NHIỆM VỤ CỦA SERVLET

- Nhận client request (hầu hết ở dạng HTTP request)
- Trích xuất 1 số thông tin từ request
- Xử lý nghiệp vụ (truy cập DB, tính toán...) hoặc sinh động nội dung
- Tạo và gửi trả response cho client (hầu hết ở dạng HTTP response) hoặc forward request cho servlet khác/cho trang JSP

REQUEST VÀ RESPONSE

- Request là gì?
 - Thông tin được gửi từ client tới 1 server
 - Ai tạo ra request
 - Dữ liệu gì được user nhập vào và gửi đi
 - Các HTTP headers
- Response là gì?
 - Thông tin được gửi đến client từ 1 server
 - Dữ liệu Text (html, thuần text) hoặc dữ liệu binary (image)
 - HTTP headers, cookies, ...

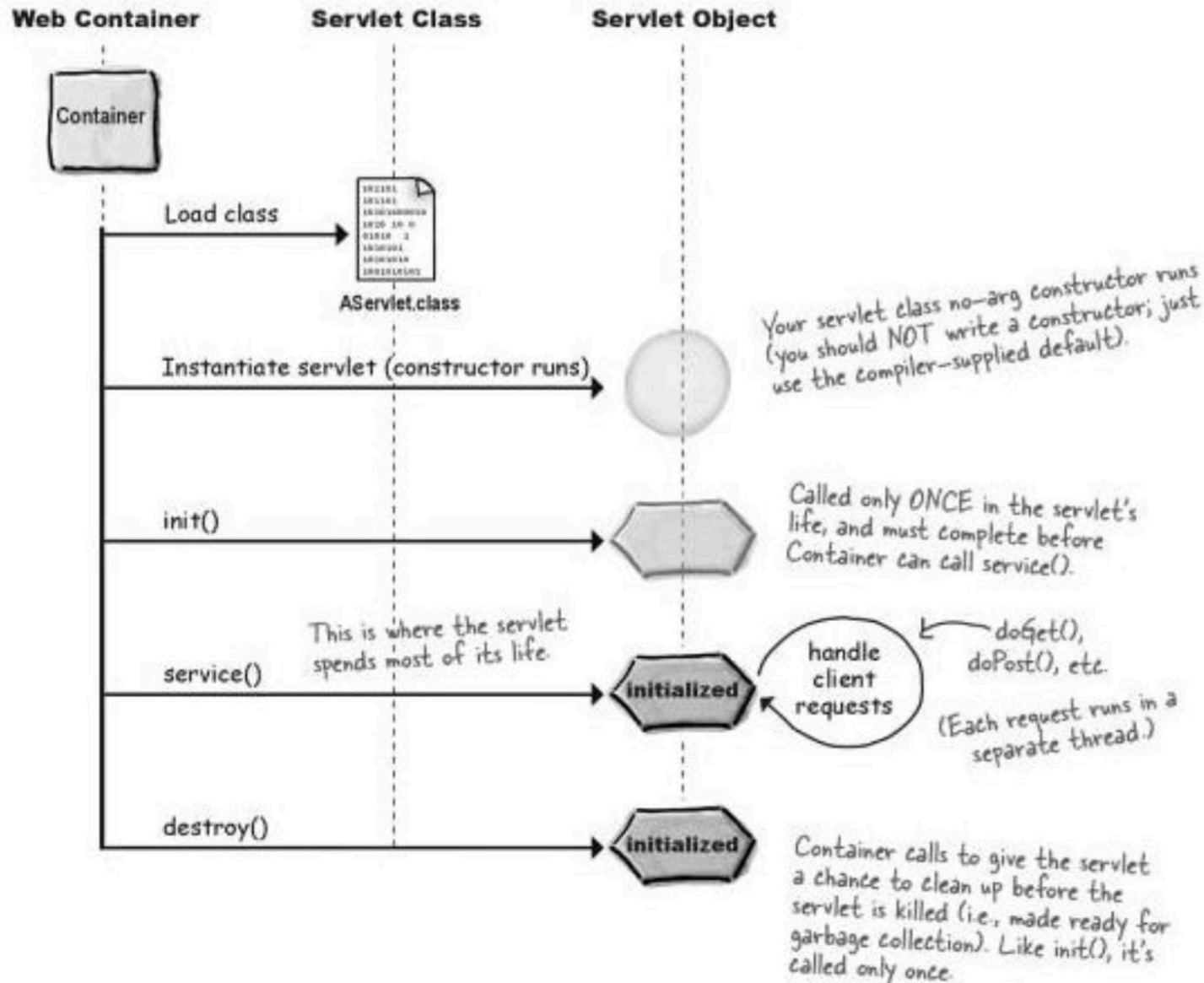


When the browser finds the opening `<html>` tag it goes into HTML-rendering mode and displays the page to the user.

When the browser gets to an image tag, it generates another HTTP request to go get the resource described. In this case the browser will make a second HTTP request to get the picture referenced in the `` tag.

2. VÒNG ĐỜI CỦA SERVLET

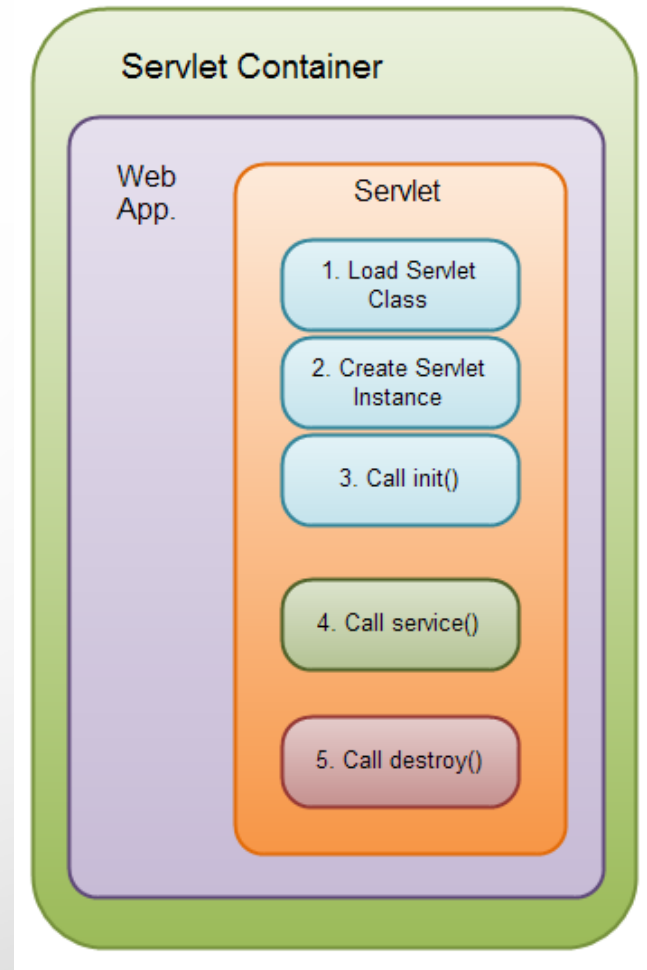
VÒNG ĐỜI CỦA SERVLET



CÁC PHƯƠNG THỨC TRONG VÒNG ĐỜI

Có 5 bước:

1. Tải Servlet Class vào bộ nhớ.
2. Tạo đối tượng Servlet.
3. Gọi method servlets init()
4. Gọi method servlets service().
5. Gọi method servlets destroy().



- Được gọi bởi container
 - Container điều khiển vòng đời của 1 servlet
- Định nghĩa trong:
 - Lớp `javax.servlet.GenericServlet`
 - `init()`
 - `destroy()`
 - `service()` - là phương thức abstract
 - Lớp `javax.servlet.http.HttpServlet`
 - `doGet()`, `doPost()`, `doXxx()`
 - `service()` – implementation

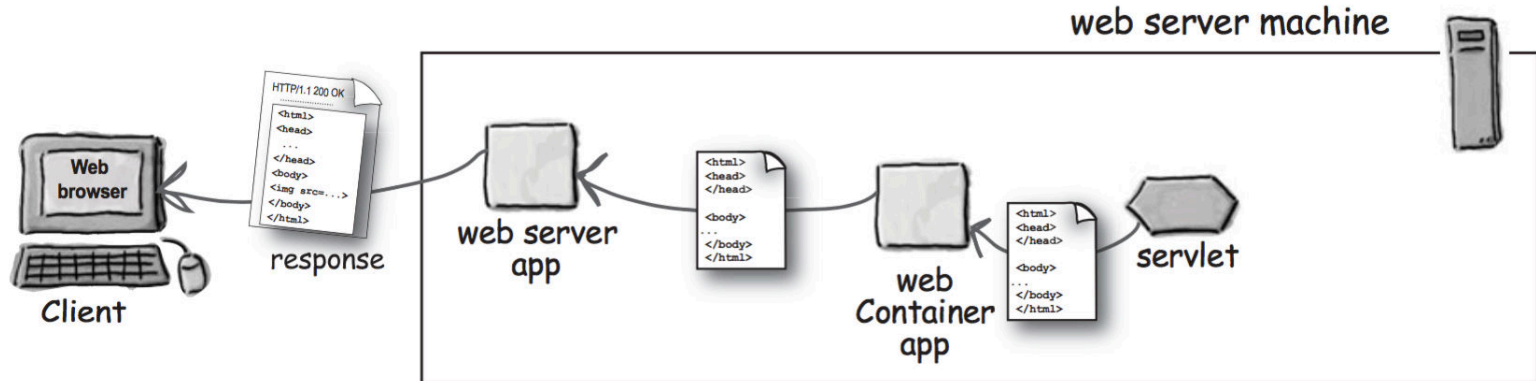
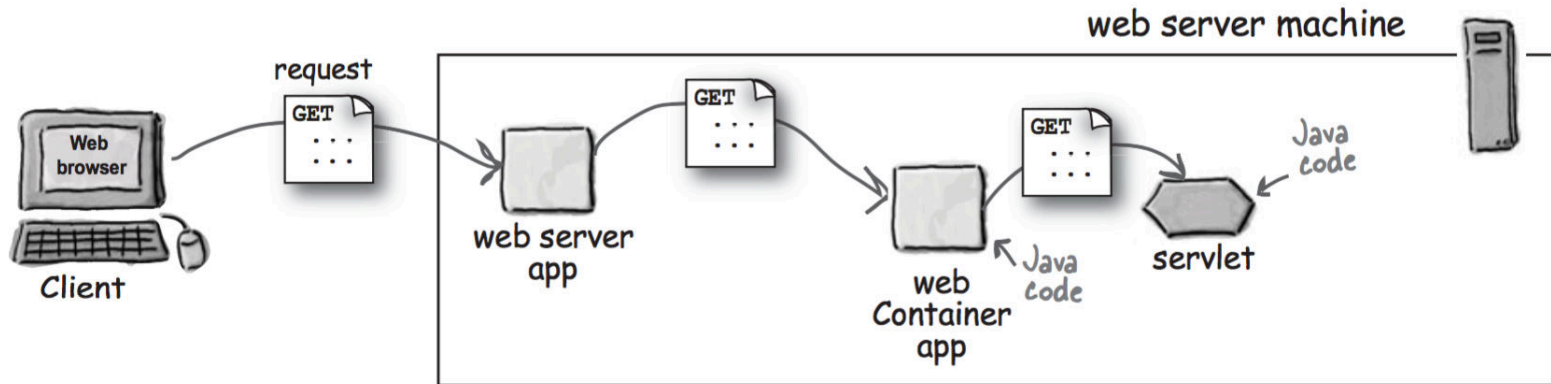
VÍ DỤ VỀ SERVLET

- Ví dụ: HelloWorld.java

```
public class HelloWorld extends HttpServlet {  
    public void doGet(HttpServletRequest request,  
                        HttpServletResponse response)  
                        throws ServletException,  
                        IOException  
    {  
        PrintWriter out = response.getWriter();  
        out.println("<html>\n" +  
                    "<head><title>Hello World</title></head>\n" +  
                    "<body>\n" +  
                    "<h1>Hello World</h1>\n" +  
                    "</body>  
                    </html>");  
    }  
}
```

3. SERVLET CONTAINER

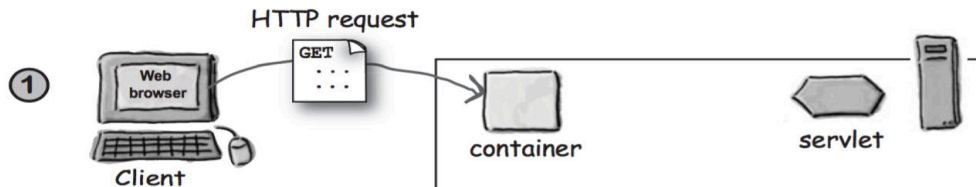
SERVLET CONTAINER



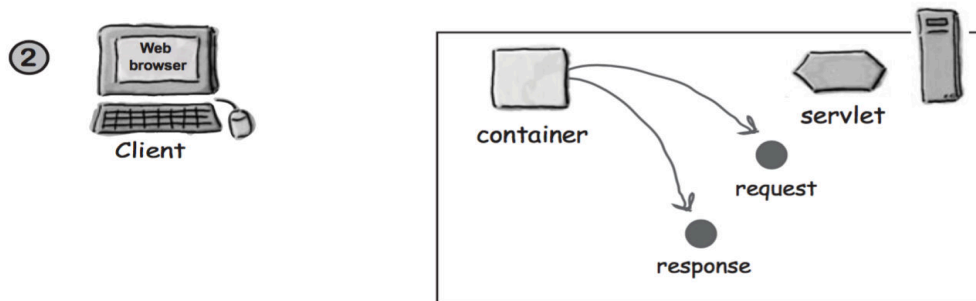
CONTAINER LÀM GÌ

- Hỗ trợ kết nối
- Quản lý vòng đời servlet
- Hỗ trợ đa luồng
- Bảo mật (Declarative Security)
- Hỗ trợ JSP

CONTAINER XỬ LÝ REQUEST?

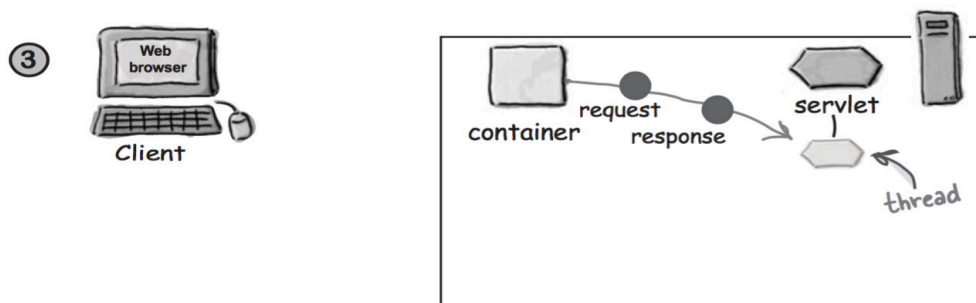


User clicks a link that has a URL to a servlet instead of a static page.

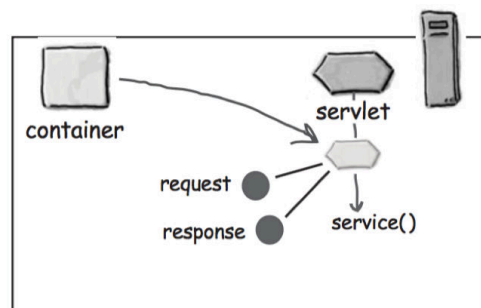
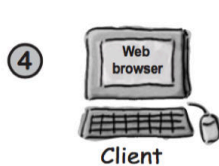


The container “sees” that the request is for a servlet, so the container creates two objects:

- 1) `HttpServletResponse`
- 2) `HttpServletRequest`

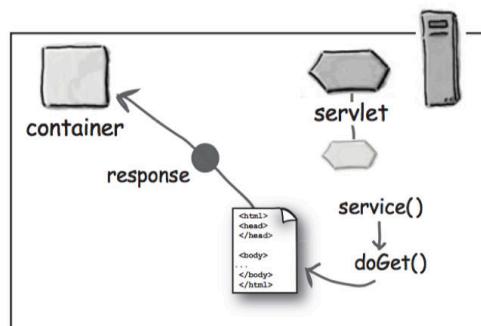
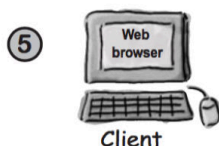


The container finds the correct servlet based on the URL in the request, creates or allocates a thread for that request, and passes the request and response objects to the servlet thread.

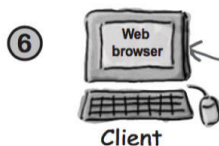


The container calls the servlet's `service()` method. Depending on the type of request, the `service()` method calls either the `doGet()` or `doPost()` method.

For this example, we'll assume the request was an HTTP GET.

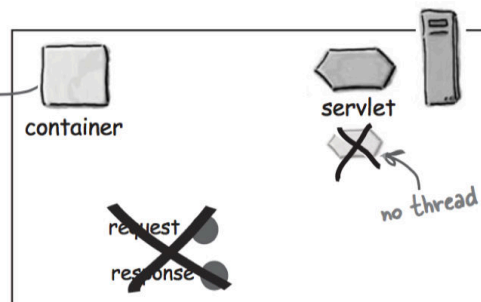


The `doGet()` method generates the dynamic page and stuffs the page into the response object. Remember, the container still has a reference to the response object!



HTTP response

```
<html>
<head>
</head>
<body>
...
</body>
</html>
```

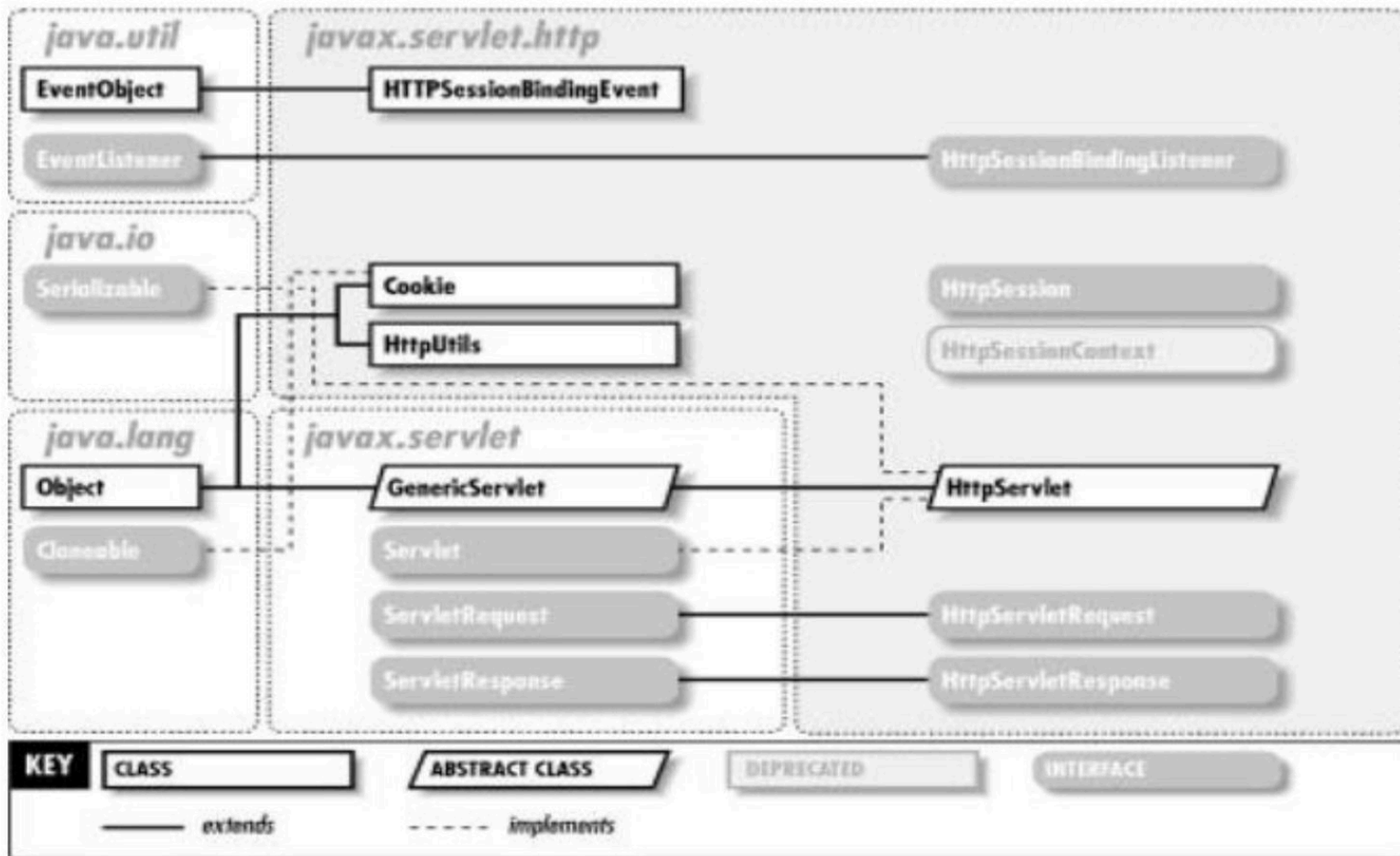


The thread completes, the container converts the response object into an HTTP response, sends it back to the client, then deletes the request and response objects.

4. SERVLET API

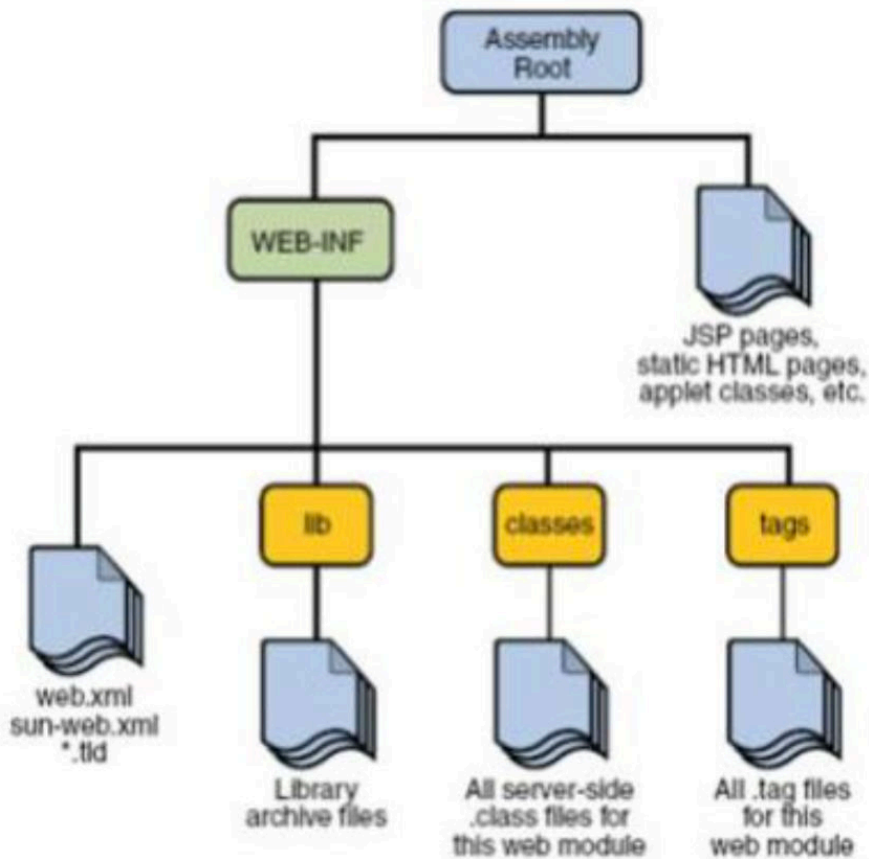
SERVLET API

- Packages:
 - javax.servlet
 - javax.servlet.html
- Gói javax.servlet chứa các giao diện servlet chung và các lớp độc lập của giao thức bất kỳ
- Gói javax.servlet.html để xây dựng hỗ trợ cho giao thức HTTP



5. CẤU TRÚC VÀ TRIỂN KHAI ỨNG DỤNG

CẤU TRÚC ỨNG DỤNG SERVLET



```
C:\tomcat
├ bin
├ classes
├ ...
├ ...
└ webapps
  └ helloapp
    ├── *.html, *.gif, *.js, *.css
    ├── *.jsp
    └ WEB-INF
      ├── classes
      │   └ com
      │       └ mycompany
      │           └ myclass.class
      ├── lib
      │   └ *.jar (jdbcdriver.jar, ...)
      └ web.xml
```

- Thư mục WEB-INF

- Nằm trong thư mục gốc của ứng dụng
- Chứa thư mục:
 - Class: các tập tin lớp servlet và các tập tin cần thiết hỗ trợ các servlet hoặc JSP
 - Lib: tất cả các tập tin JAR/ZIP được sử dụng bởi các ứng dụng web
 - Tập tin mô tả triển khai **web.xml**

TRIỂN KHAI ỨNG DỤNG LÊN SERVER

- Đóng gói tệp tin war và lưu trữ vào thư mục web của application server
 - Tomcat có thư mục webapps
- Tomcat tự động trích xuất nội dung tệp tin WAR vào một thư mục dưới ứng dụng web
- Chạy ứng dụng

HẾT CHƯƠNG 13