

SERVLETS III

CS 434

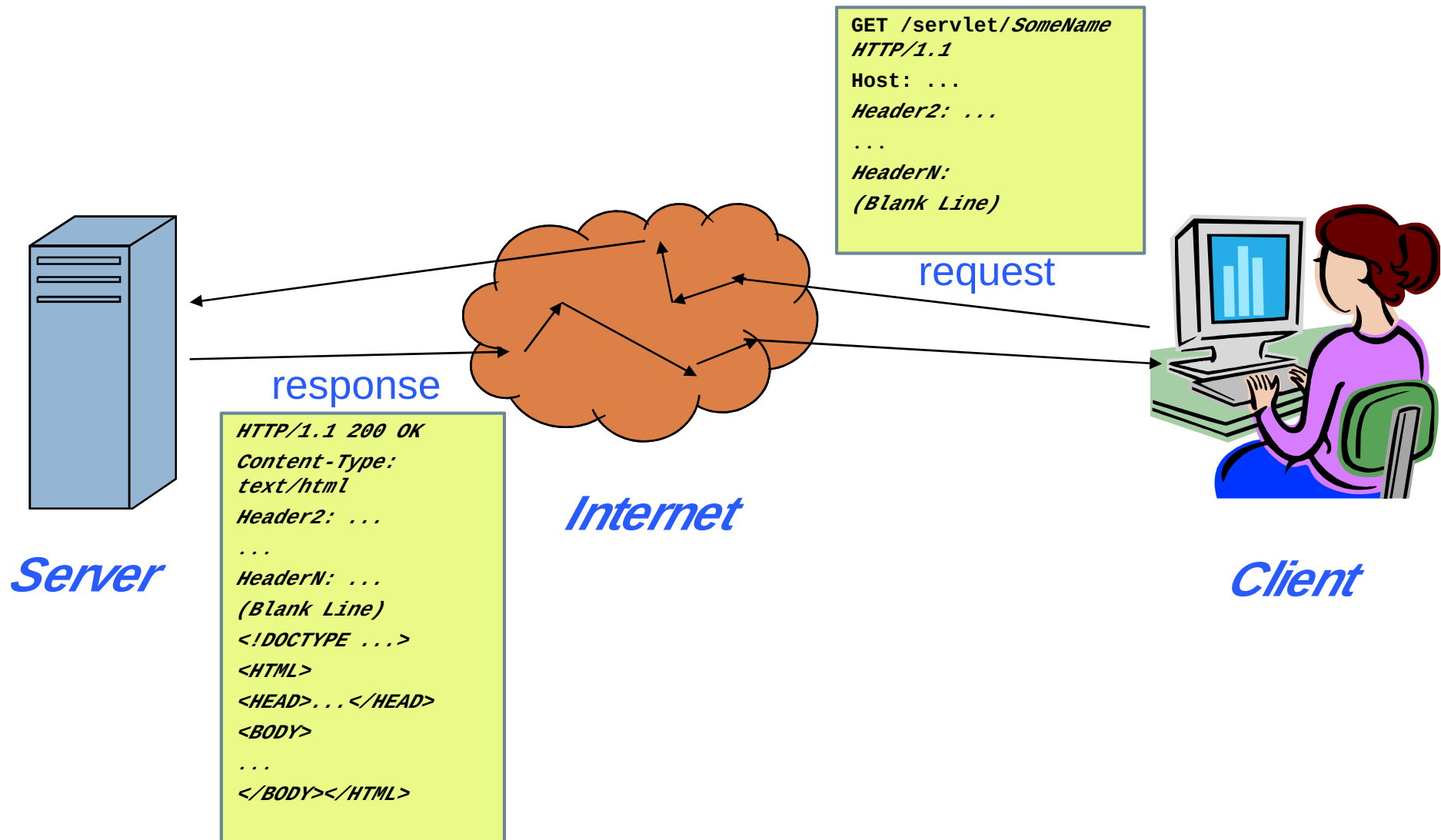
T. Asma Alsaleh and T. Lamees Alhazzaa

Outlines

2

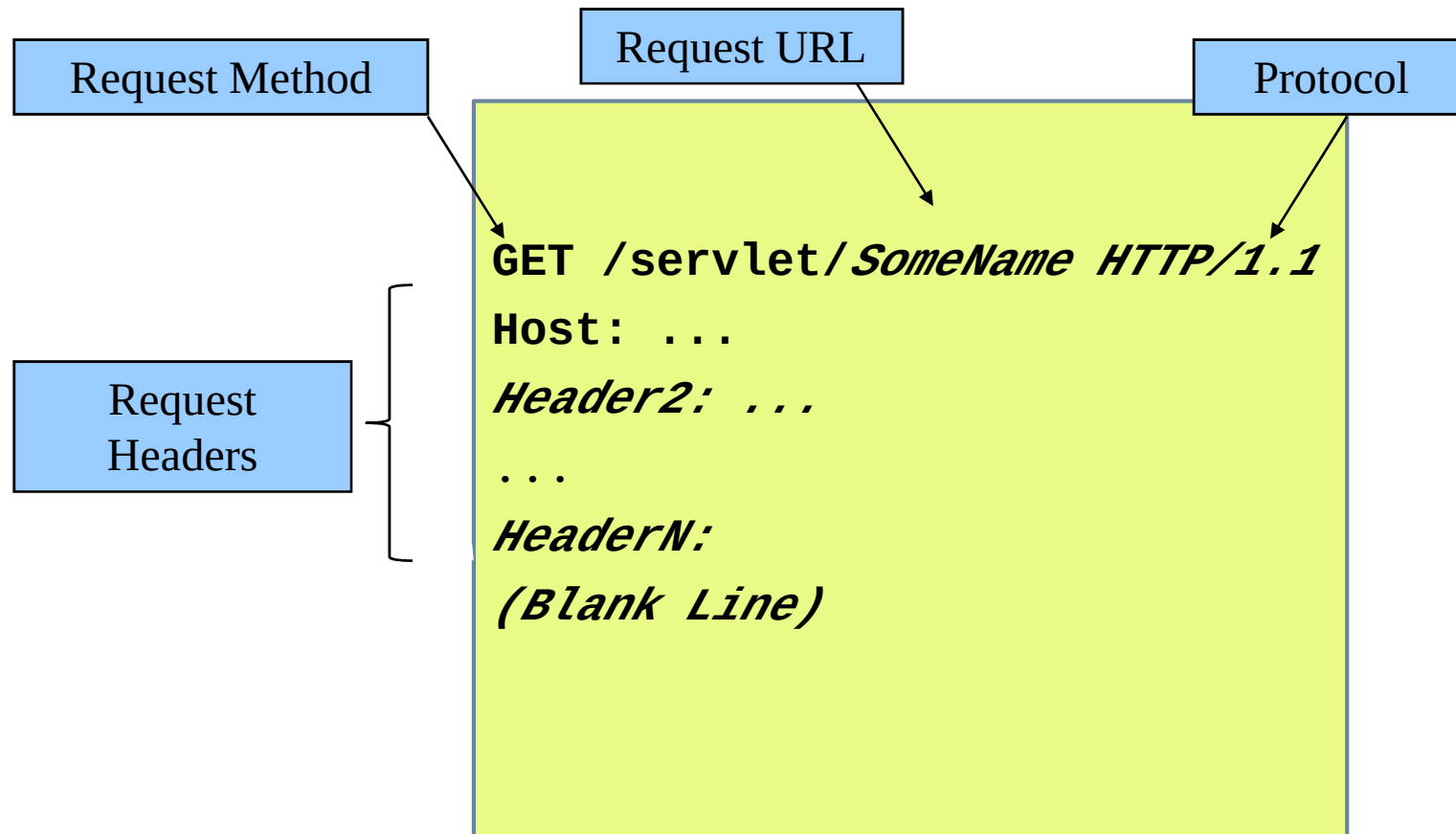
- Overview of servlets.
- Servlets life cycle.
- Handling the client request:
 - ▣ Form data.
 - ▣ HTTP request headers.
- Generating the server response:
 - ▣ HTTP status codes.
 - ▣ HTTP response headers.
- Handling cookies.
- Session tracking.

Client-Server Model



A Typical HTTP 1.1 Request

4



A Typical HTTP 1.1 Response

e

Status Code

Status Line

Request
Headers

Document

```
HTTP/1.1 200 OK  
Content-Type: text/html  
Header2: ...  
...  
HeaderN: ...  
(Blank Line)  
<!DOCTYPE ...>  
<HTML>  
<HEAD>...</HEAD>  
<BODY>  
...  
</BODY></HTML>
```

HTTP Status Codes

Status Code

▼

- The status line consists of
 - ▣ The HTTP version.
 - ▣ An integer that is interpreted as a status code.
 - ▣ A very short message corresponding to the status code.
- Servlets don't usually need to specify a status code at all.
 - ▣ A code of 200 (OK) is set automatically.

Setting Status Codes



- Methods (HttpServletResponse):
 1. response.setStatus
 2. response.sendError
 3. response.sendRedirect

- Set status codes before sending any document content to the client.

Setting Status Codes: setStatus

9

- `response.setStatus(int statusCode)`
 - ▣ Use a constant for the code, not an explicit `int`.
 - ▣ Constants are in `HttpServletResponse`.
 - ▣ For example, you would use:

```
response.setStatus(response. SC_NOT_FOUND);
```

rather than:

```
response.setStatus(404);
```

Setting Status Codes: `sendError`

- Shortcut method in `HttpServletResponse`:

1. **`sendError`** :

- ▣ `response.sendError(int code, String message);`
- ▣ The `sendError` method sends a status code (usually 404) along with a short message inside an HTML document and sent to the client.

Setting Status Codes: `sendRedirect`

11

2. `sendRedirect`:

- ▣ `response.sendRedirect(String url)`
- ▣ Generates:
 1. 302 response.
 2. Location header giving the URL of the new document.
- ▣ Use it instead of `setStatus`.

A Servlet That Redirects Users to Browser-Specific Pages

۱۲

```
public class WrongDestination extends HttpServlet {

    public void doGet(HttpServletRequest request,
        HttpServletResponse response) throws ServletException,
        IOException
    {

        String userAgent = request.getHeader("User-Agent");

        if ((userAgent != null) && (userAgent.indexOf("MSIE") != -1))
        { //Internet Explorer users
            response.sendRedirect("http://home.netscape.com");
        }
        else
        { //Other users
            response.sendRedirect("http://www.microsoft.com");
        }
    }
}
```

Some of Status Codes

۱۳

- 200–299 Values in the 200s signify that the request was *successful*
- 300–399 Values in the 300s are used for files that have *moved* and usually include a Location header indicating the new address
- 400–499 Values in the 400s indicate an *error* by the *client*

Some of Status Codes

14

- 200 (OK)
 - ▣ SC_OK.
 - ▣ Everything is fine; document follows.
 - ▣ Default for servlets,; if you don't use `setStatus`, you'll get 200.

- 204 (No Content)
 - ▣ SC_NO_CONTENT.
 - ▣ Browser should keep displaying previous document because no new document is available.
 - ▣ This behavior is useful if the user periodically reloads a page by pressing the Reload button.
 - ▣ `response.setStatus(response.SC_NO_CONTENT);`

Some of Status Codes

10

- 301 (Moved Permanently)
 - ▣ SC_MOVED_PERMANETLY.
 - ▣ Requested document permanently moved elsewhere (indicated in Location header).
 - ▣ Browsers go to new location automatically.

```
response.setStatus(response.SC_MOVED_PERMANTLY);  
response.setHeader("Location", "http://...");
```

OR

```
response.sendRedirect("http://...");
```

Some of Status Codes

۱۶

- 401 (Unauthorized)
 - ▣ SC_UNAUTHORIZED
 - ▣ Browser tried to access protected page without proper Authorization header.

- 404 (Not Found)
 - ▣ SC_NOT_FOUND
 - ▣ No such page.
 - ▣ `response.setStatus(response.SC_NOT_FOUND);`
 - ▣ Servlets should use `sendError` to set this header.

HTTP Response Headers

HTTP Response Headers

18

1. Setting the HTTP response headers often (not always) goes hand in hand with setting the status codes in the status line.
 - ▣ E.g. all the “document moved” status codes (300 through 307) have an accompanying Location header.
2. Specifying headers can also play a useful role even when no unusual status code is set.
 - ▣ E.g. Response headers can be used to:
 - Specify cookies.
 - Instruct the browser to reload the page after a designated interval.
 - Designate the type of document being generated.

Setting Arbitrary Response Headers

19

- **`response.setHeader(String headerName, String headerValue)`**
 - ▣ Sets an arbitrary header.
- **`response.setIntHeader(String name, int headerValue)`**
 - ▣ Prevents need to convert **`int`** to **`String`** before calling **`setHeader`**.
- **`addHeader, addIntHeader`**
 - ▣ *Adds* new occurrence of header instead of *replacing* any existing header with the same name.
 - ▣ Different supported MIME types and different cookies.

Common HTTP 1.1 Response Headers

□ **addCookie(Cookie c)**

- ▣ Inserts a cookie into the **Set-Cookie** header.
- ▣ Add (insert), not set (replace).

□ **Location**

- ▣ The URL to which browser should reconnect.
- ▣ Use **sendRedirect** instead of setting this directly.

□ **Content-Type**

- ▣ The MIME type of the document being returned.
- ▣ Servlets *almost always* use this.
- ▣ **response.setContentType("text/html");**
- ▣ See table 7.1 in chapter 7 for a list of common MIME type.

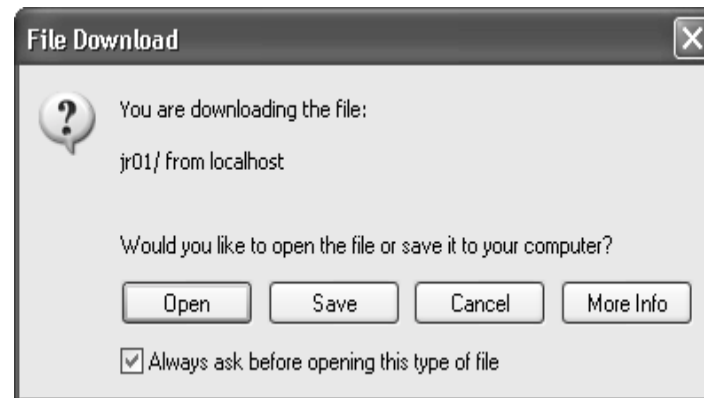
Common HTTP 1.1 Response Headers

□ Content-Disposition

- ▣ Lets you request that the browser ask the user to save the response to disk in a file of the given name:

Content-Disposition: attachment; filename=*file-name*

- ▣ Non-HTML contents such as Excel spreadsheet.



□ Content-Encoding

- ▣ The way document is encoded during transmission.
- ▣ E.g. gzip.

Common HTTP 1.1 Response Headers

۲۲

□ Last-Modified

- ▣ The time document was last changed.

□ Refresh

- ▣ The number of seconds until browser should reload page:
- `response.setIntHeader("Refresh", 30 )//30 seconds`
 - ▣ Can also include URL to connect to
- `response.setHeader("Refresh", "5; URL=http://host/path/");`

Building Excel Spreadsheets

۲۳

http://localhost/servlet/coreservlets.ApplesAndOranges - Microsoft Internet Explorer

File Edit View Insert Format Tools Data Go To Favorites Help

Address <http://localhost/servlet/coreservlets.ApplesAndOranges> Go

	A	B	C	D	E	F	G	H
1		Q1	Q2	Q3	Q4	Total		
2	Apples	78	87	92	29	286		
3	Oranges	77	86	93	30	286		
4								
5								
6								

coreservlets.ApplesAndOranges

Unknown Zone

Building Excel Spreadsheets

۲۴

```
public class ApplesAndOranges extends HttpServlet {

    public void doGet(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException
    {
        response.setContentType("application/vnd.ms-excel");

        PrintWriter out = response.getWriter();

        out.println("\tQ1\tQ2\tQ3\tQ4\tTotal");
        out.println("Apples\t78\t87\t92\t29\t=SUM(B2:E2)");
        out.println("Oranges\t77\t86\t93\t30\t=SUM(B3:E3)");
    }
}
```

References

۲۵

- Material for this lecture was obtained from:
 1. The core web programming, by Marty Hall and Larry Brown, A Sun Microsystems Press/Prentice Hall PTR Book.