Generating dynamic pages

How can we generate a dynamic Web page? Hello, John! example

1. Programmatic approach: Write a program

Example: Java Servlet for “Hello, John!”

```java
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException
{
    PrintWriter out = response.getWriter();
    out.println("<html>\n    <head><title>Hello</title></head>\n    <body>Hello, " + request.getParameter("first_name") + "!\n    </body>\n    </html>\n    out.close();
}
```

2. Template approach: Write a Web page with “variables”

Example: Java ServerPages (JSP)

```html
<html>
<head><title>Hello</title></head>
<body>
    Hello, <%= request.getParameter("first_name") %>!
</body>
</html>
```

• Q: What are the potential problems with either of the two approaches?

• Comments
  o Even for template approach, once complex code gets embedded inside, the page gets ugly and becomes difficult to maintain quickly
  o Separation of data (model) and presentation (view)
    ▪ One data may be presented in many different ways depending on device and/or user
Code “ownership”
- Page design is often done by Web designers, while coding is often done by application developers.
- Who “owns” the above page(s)?
- Easy to introduce errors

Model-View-Controller (MVC) design pattern
- Application is developed using three modular components
  - Model: manages domain data and its related logic
    - Often implemented as a Java class or a Java Bean
  - View: deals with the output representation of data
    - Often implemented as a JSP page
    - Owned and maintained by “Web designers”
  - Controller: deals with user requests, retrieving and modifying the relevant models and forwarding them to the appropriate views
    - Often implemented as a Java servlet

Example

In Java servlet:

```java
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    ... retrieve data, update it, etc. here ...
    request.setAttribute("data1", XXX);  // prepare data for view
    request.getRequestDispatcher("/index.jsp").forward(request, response);
}
```

In index.jsp:

```html
<html>
<head><title>Demo</title></head>
<body>
Your data: <%= request.getAttribute("data1") %>
</body>
</html>
```

- Specialized “tags” to make it easy to add simple logic to the view
  - Java Standard Tag Libraries (JSTL)

```html
<%@ taglib uri="http://java.sun.com/jstl/core" prefix="c" %>
<html>
<body>
<table>
```
Cookies and Sessions

- HTTP is a stateless protocol. The server's response is purely based on the single request, not anything else

- Q: How does a web site like Amazon can “remember” what you were doing and implement things like shopping cart? How does it know that multiple HTTP requests are coming from you?

- Cookie: a short text that browser always sends to server for every request
  - helps the server identify all requests from a particular user
  - Protocol:
    * From server:
      Set-Cookie: name=value; path=/; domain=.ucla.edu; expires=date
      - browser stores the information locally and sends the (name,value) pair when the domain and the path match
      - With no expiration date: Cookie is deleted when browser closes
      * session cookie vs. persistent cookie
      - With "secure;" attribute: cookie is sent only over https
    * From client
      Cookie: name=value
      - (name, value) pair can be anything. userid, password, etc.
      - Same-origin policy: The browser sends a cookie only to the domain where it came from. No cross-domain cookie exchange
        - Q: Why same origin policy?

• Cookie theft and cookie poisoning

  - **Note** Be very careful about what you store in cookie

• Third-party cookie. Can we use cookie to identify a user across multiple domains?

• Q: How can we let users log in once, without asking for authentication for every request? How can we implement something like shopping cart?

  - Q: Any way to minimize the risk of cookie theft?

• Q: If we have two domains, cs143.com and cs144.com, is it possible to ask the user to log in just once for both domains?