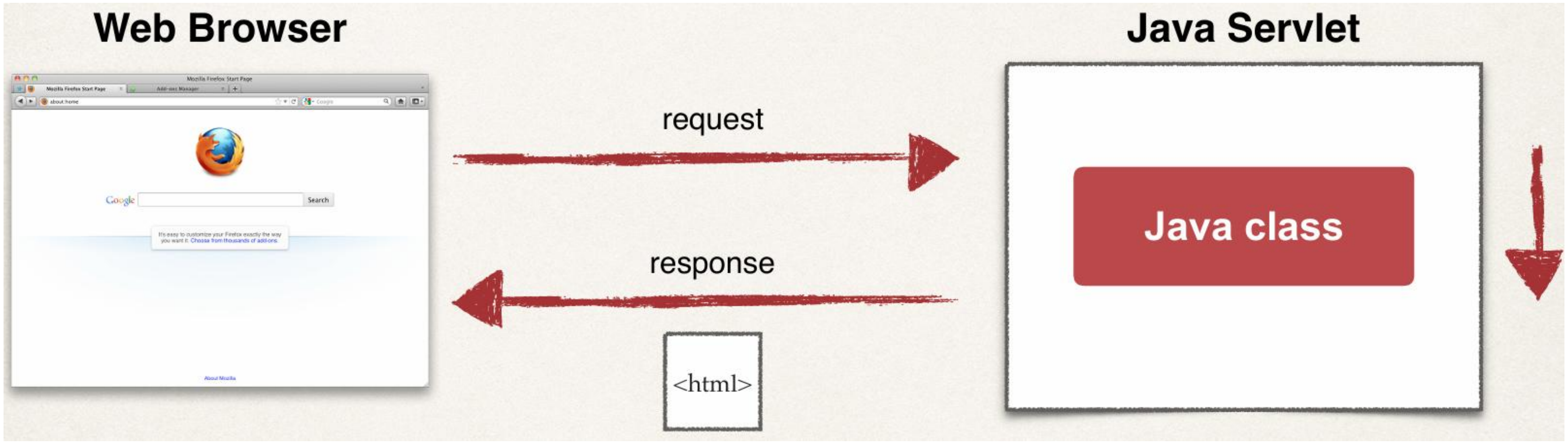
Servlets

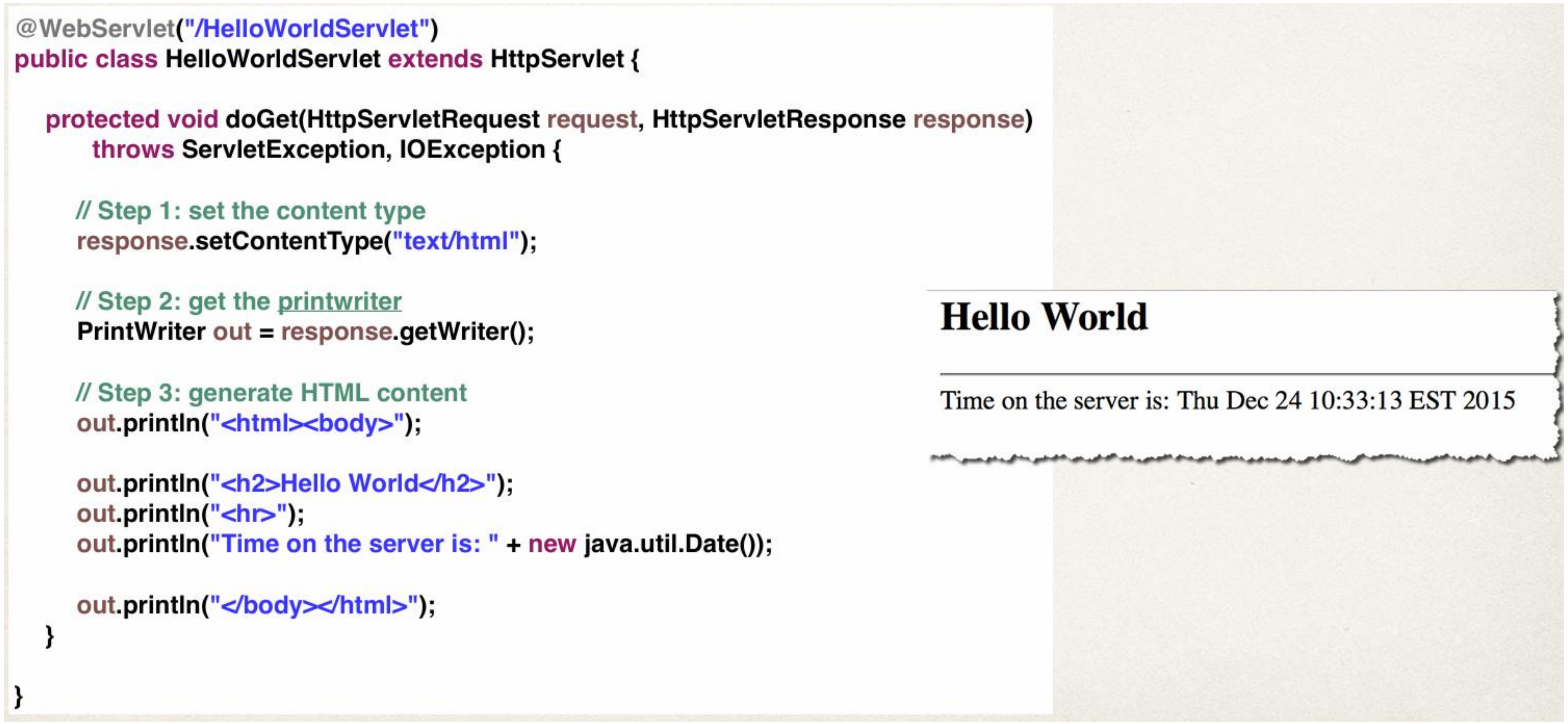
Hello World Servlet

- Java class that is processed on the server
- Java class generates HTML that is returned to browser
- Can read HTML form data, use cookies and sessions etc…
- At a high-level, similar functionality to JSPs

What Are Servlets?



Show me the code!



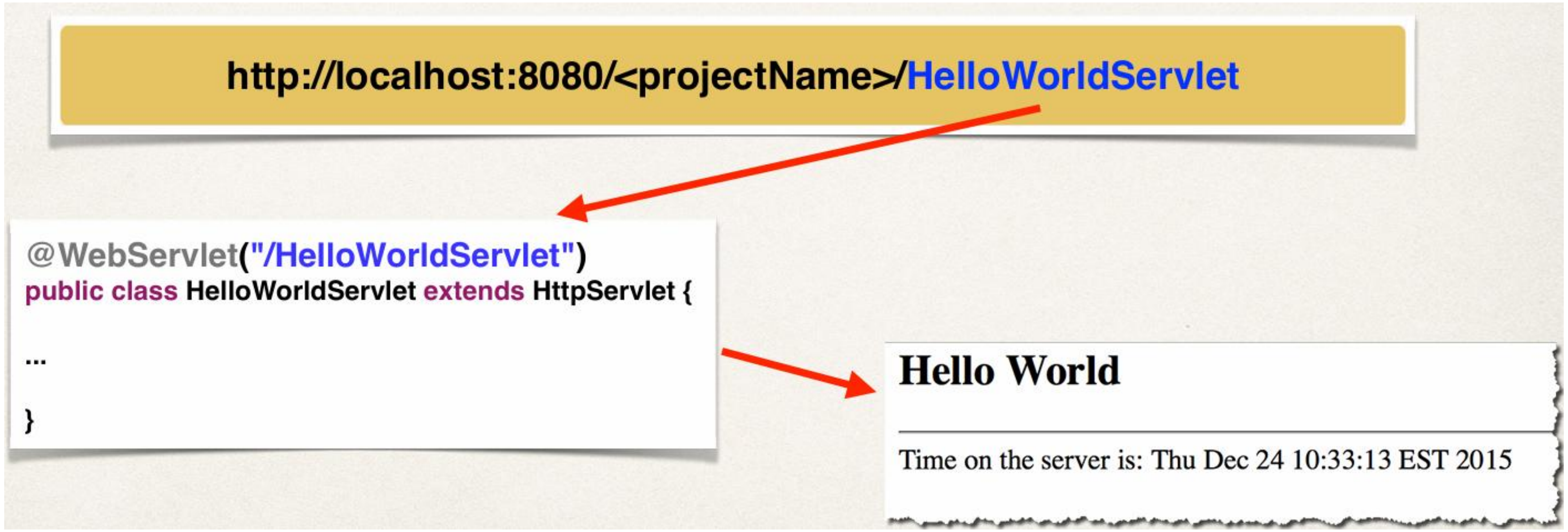
```java
@WebServlet("/HelloWorldServlet")
public class HelloWorldServlet extends HttpServlet {

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

      // Step 1: set the content type
      response.setContentType("text/html");

      // Step 2: get the printwriter
      PrintWriter out = response.getWriter();

      // Step 3: generate HTML content
      out.println("<html><body>");

      out.println("<h2>Hello World</h2>");
      out.println("<hr>");
      out.println("Time on the server is: " + new java.util.Date());

      out.println("</body></html>");
   }

}
```

Hello World

Time on the server is: Thu Dec 24 10:33:13 EST 2015

Access the Servlet

- Use the path specified in WebServlet annotation



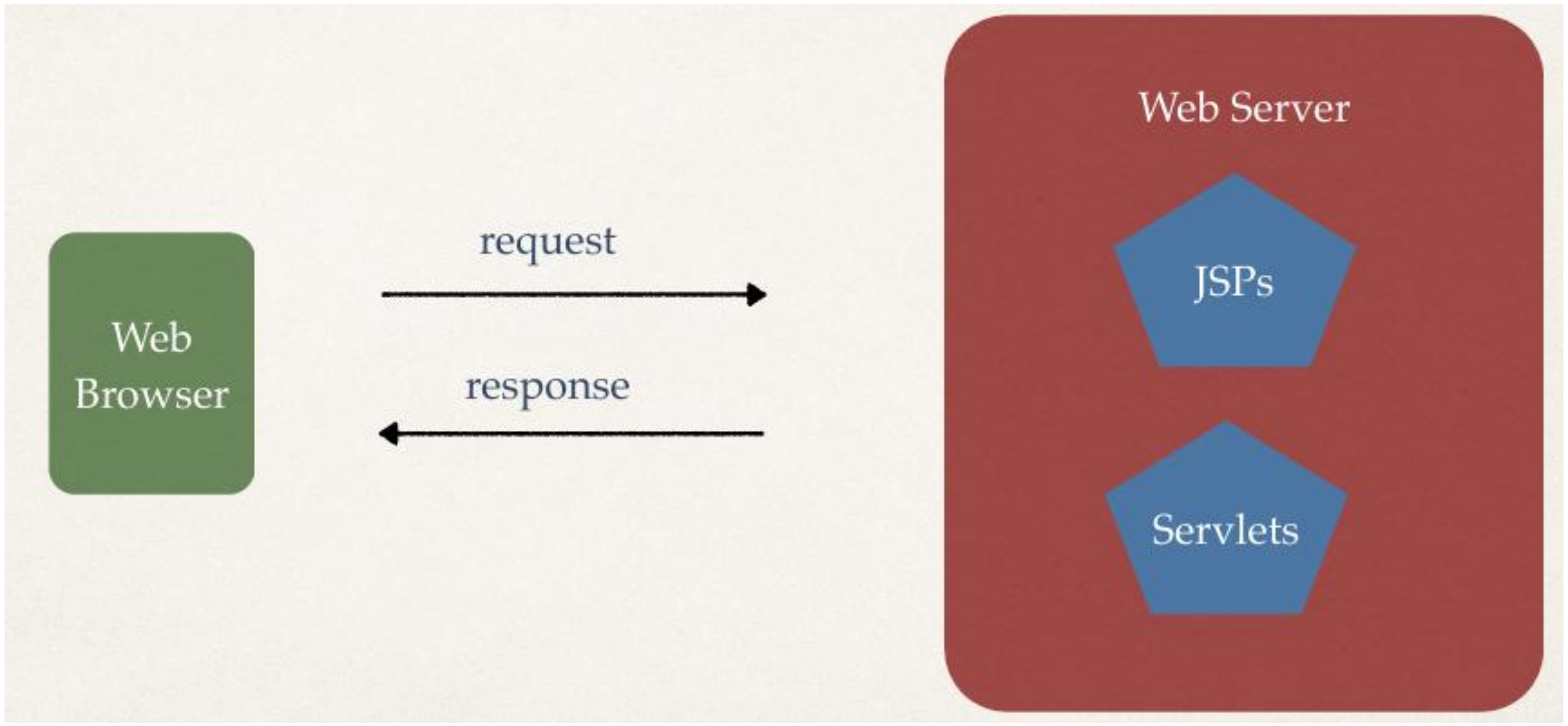
http://localhost:8080/<projectName>/HelloWorldServlet

```
@WebServlet("/HelloWorldServlet")
public class HelloWorldServlet extends HttpServlet {

...

}
```

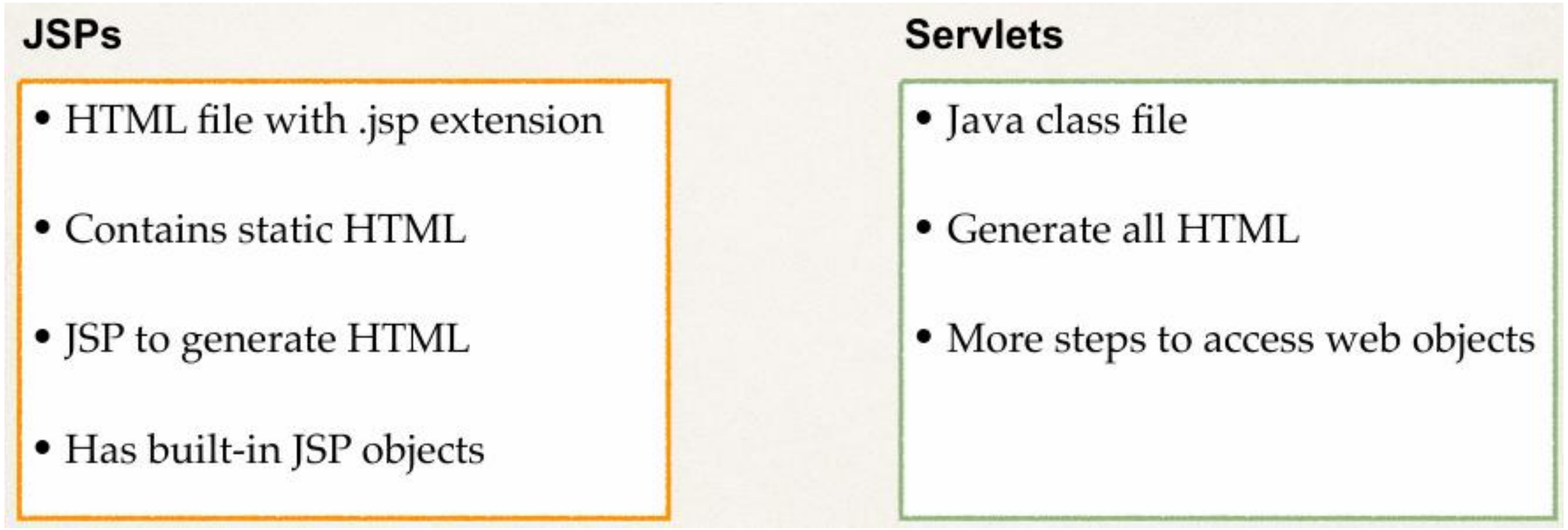
```
Hello World

Time on the server is: Thu Dec 24 10:33:13 EST 2015
```

Comparing Servlets and JSP

JSP and Servlets



Comparing JSP and Servlets



JSPs

- HTML file with .jsp extension
- Contains static HTML
- JSP to generate HTML
- Has built-in JSP objects

Servlets

- Java class file
- Generate all HTML
- More steps to access web objects

Which One?

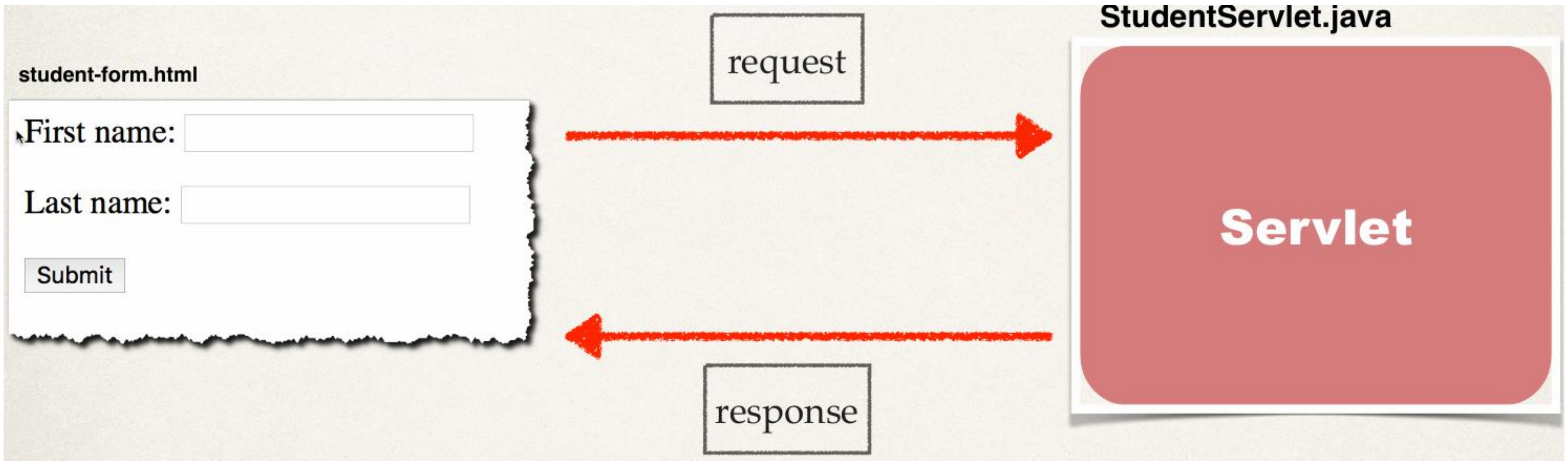
- Can use either one for building Java web apps …
- Build entire site using only Servlets … or
- Build entire site using only JSPs

Best Practice

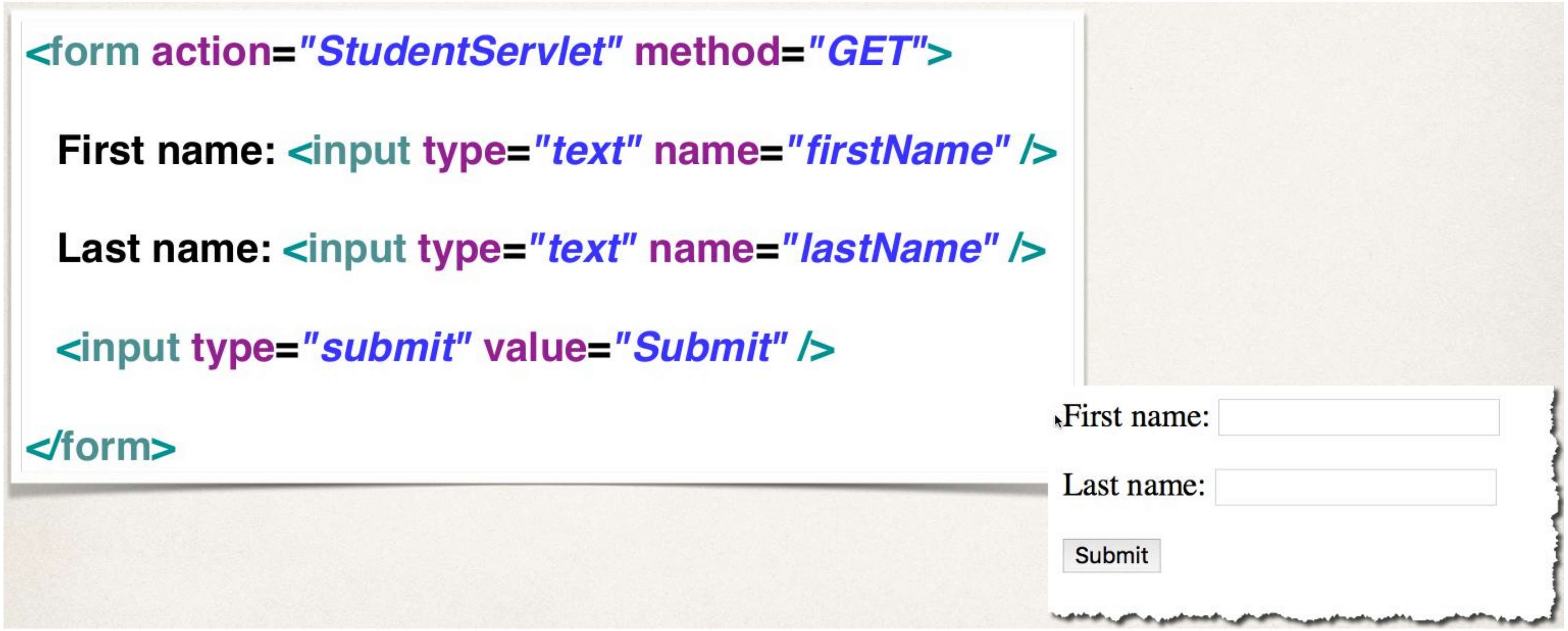
- Integrate them both together!
  - Servlet does the business logic
  - JSP handles the presentation view
- Model-View-Controller (MVC) Design Pattern

Reading HTML Form Data with Servlets

HTTP Request / Response



Step 1: Building HTML Form

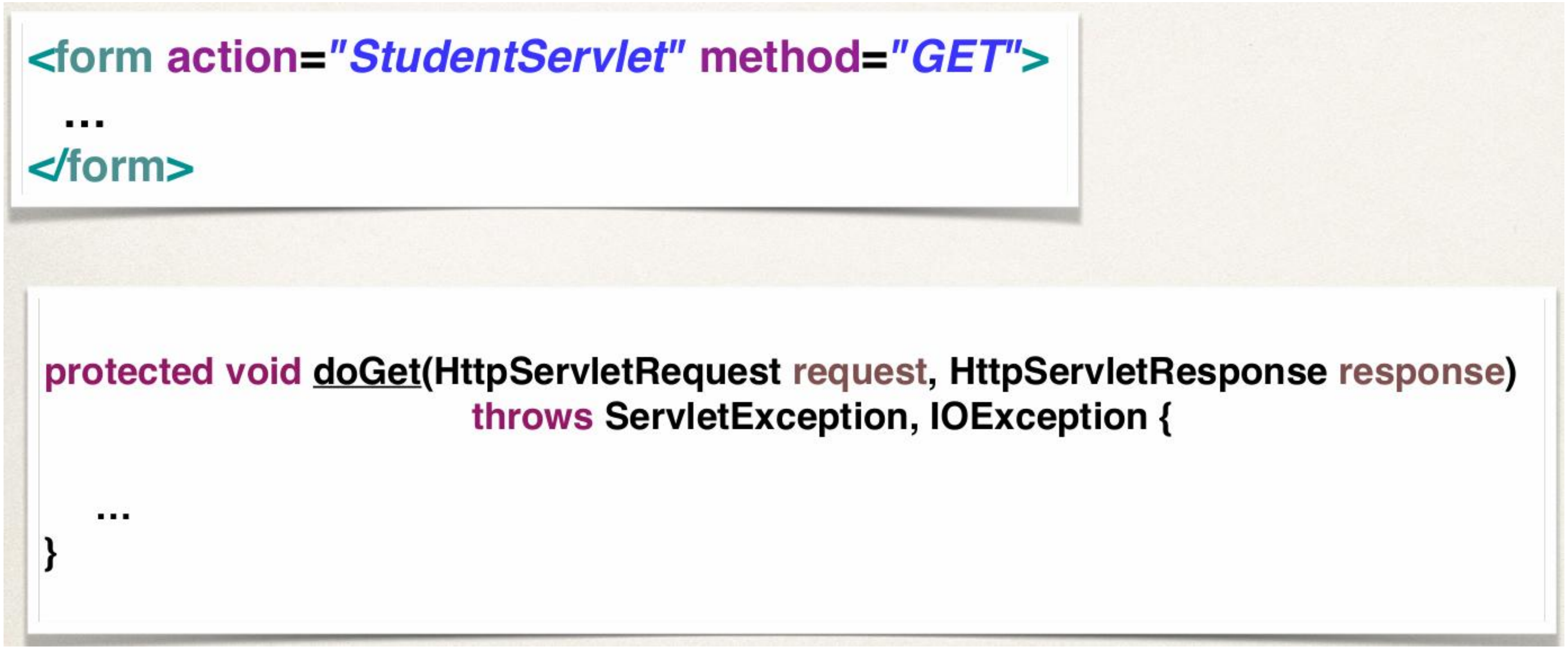


```
<form action="StudentServlet" method="GET">

  First name: <input type="text" name="firstName" />

  Last name: <input type="text" name="lastName" />

  <input type="submit" value="Submit" />

</form>
```

First name:

Last name:

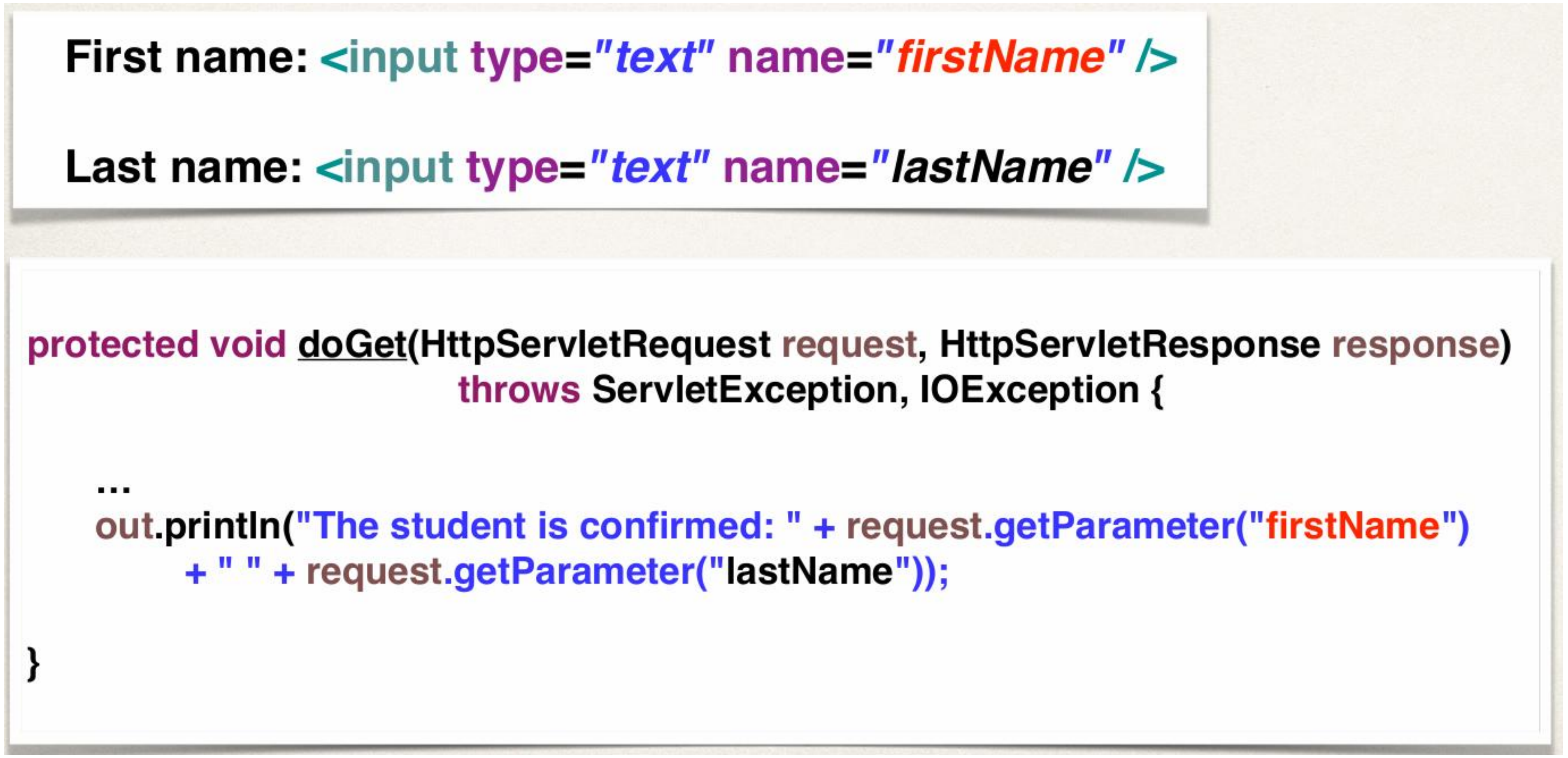
Form GET method calls Servlet doGet() method



```
<form action="StudentServlet" method="GET">
  ...
</form>
```

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

    ...
}
```

Step 2: Reading Form Data with Servlet

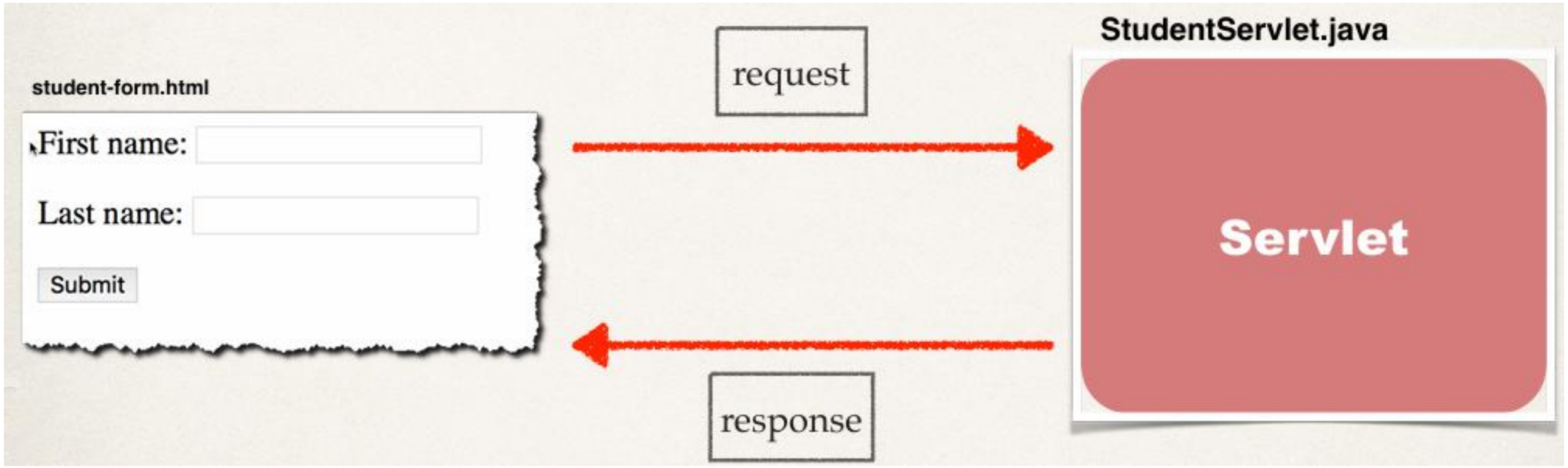


```
First name: <input type="text" name="firstName" />

Last name: <input type="text" name="lastName" />
```

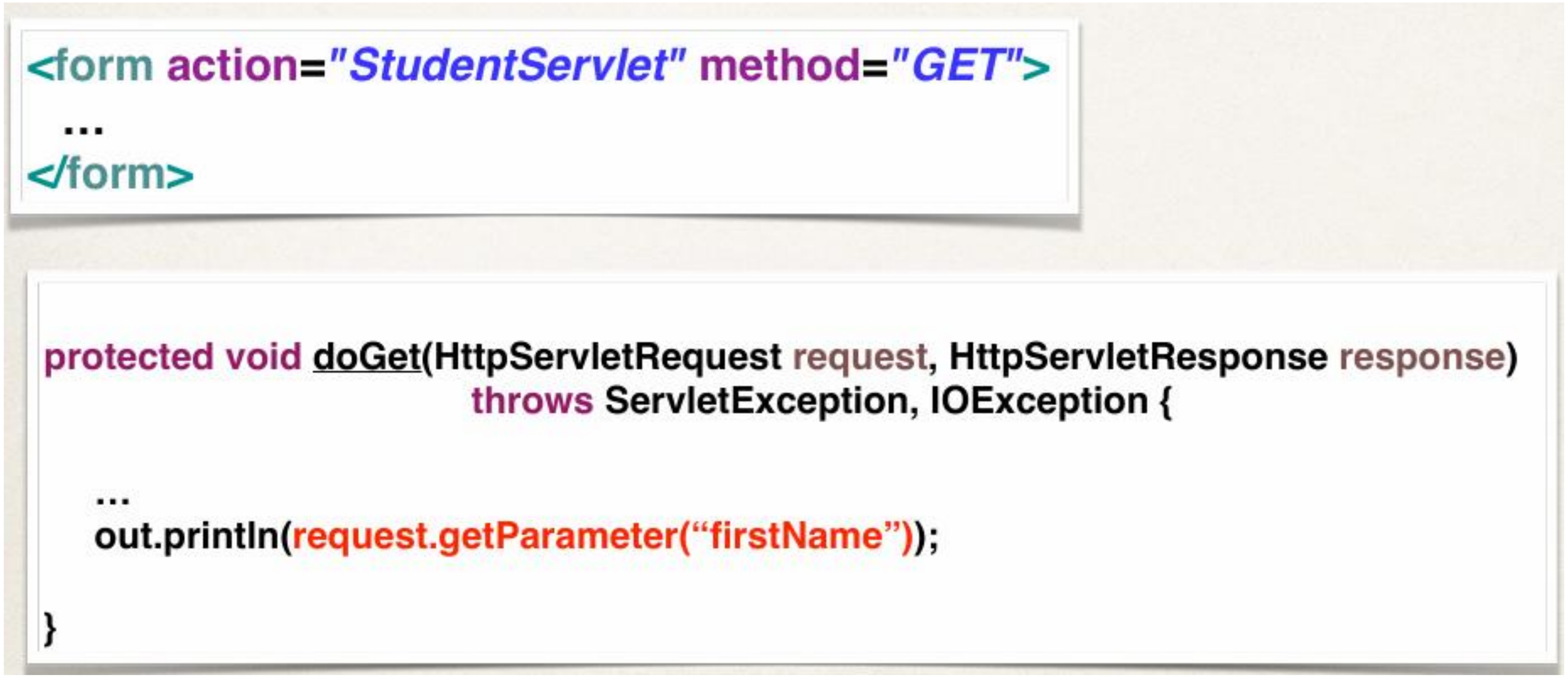
```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
                                throws ServletException, IOException {

    ...
    out.println("The student is confirmed: " + request.getParameter("firstName")
          + " " + request.getParameter("lastName"));

}
```

Difference between GET and POST

HTTP Request / Response



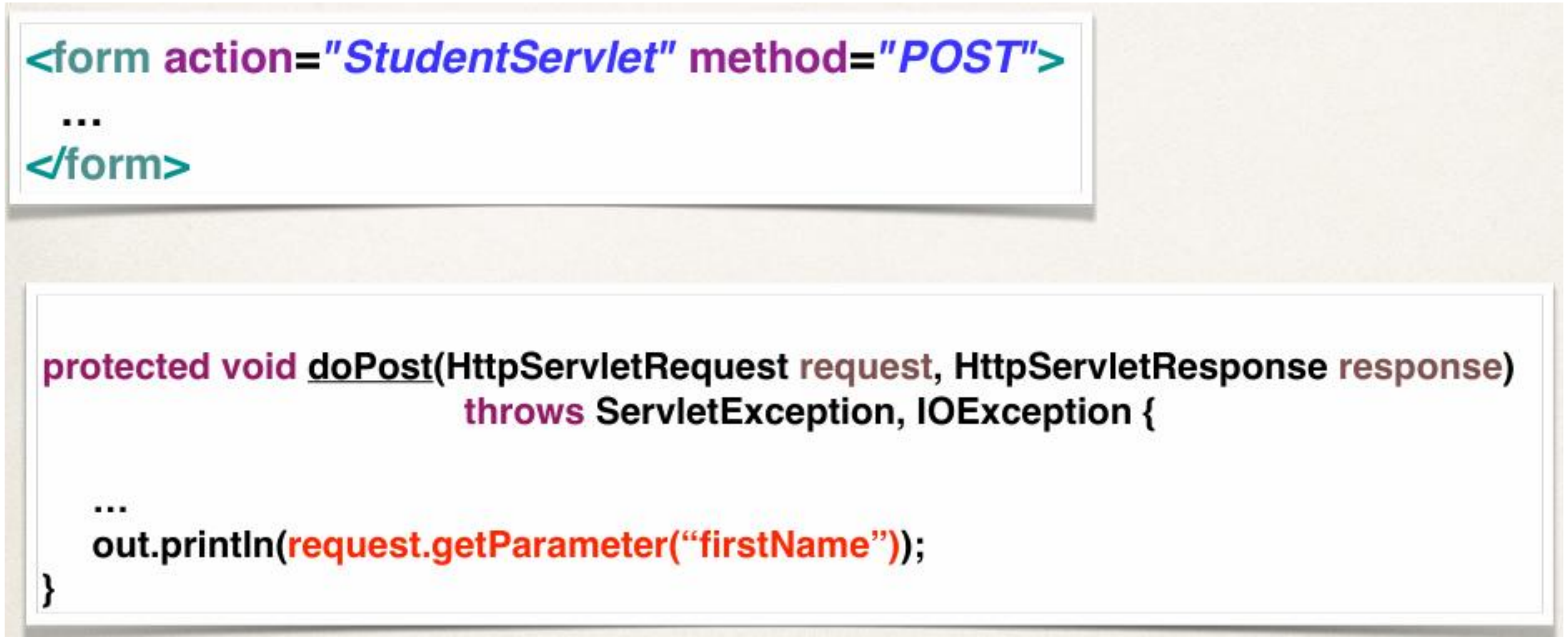
Form GET method calls Servlet doGet() method



```
<form action="StudentServlet" method="GET">
  ...
</form>
```

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

    ...
    out.println(request.getParameter("firstName"));

}
```

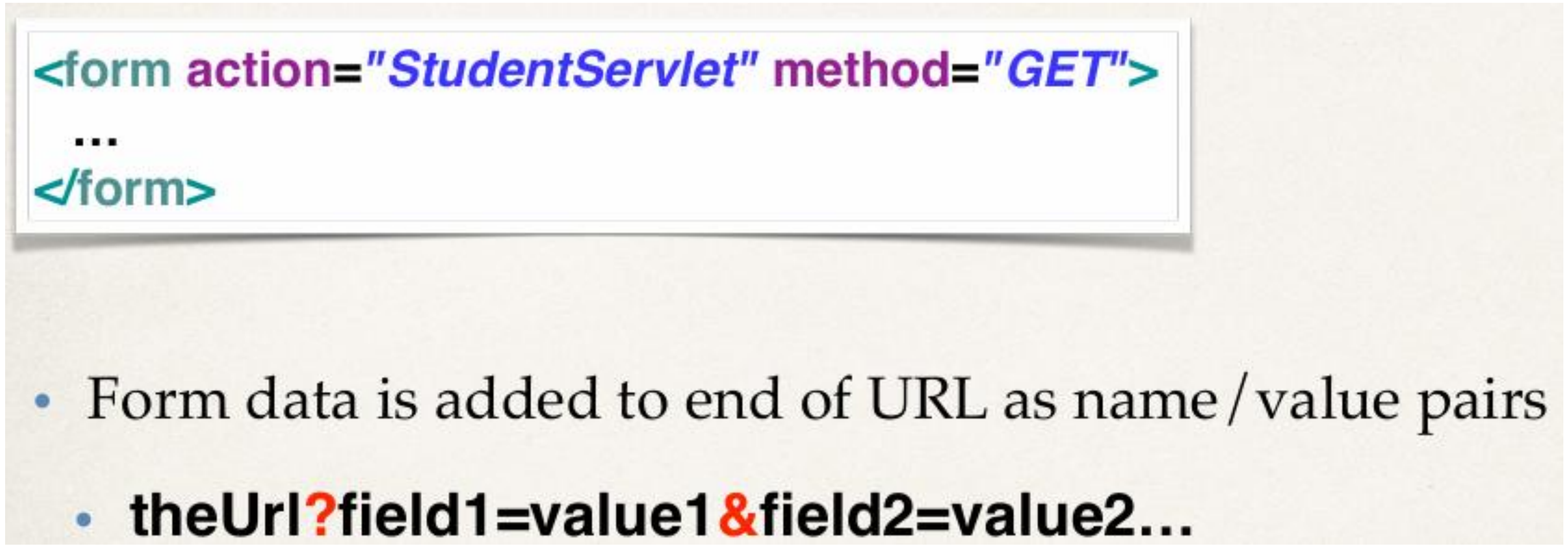
Form POST method calls Servlet doPost() method



```
<form action="StudentServlet" method="POST">
  ...
</form>
```

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)
                    throws ServletException, IOException {

    ...
    out.println(request.getParameter("firstName"));
}
```

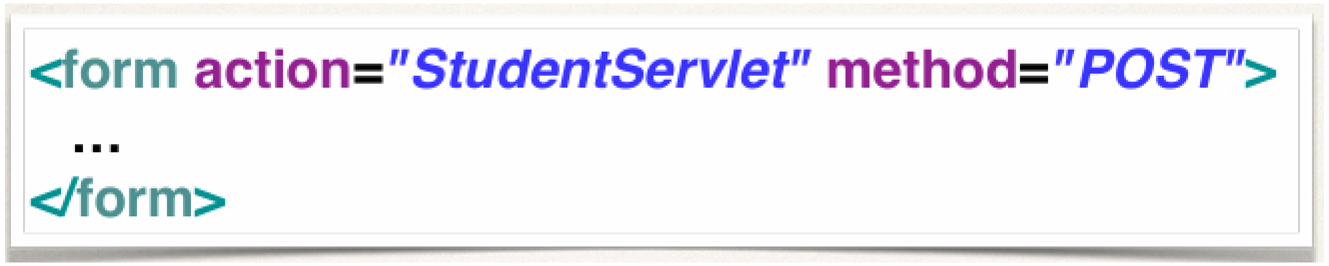
Sending Data with GET method



```
<form action="StudentServlet" method="GET">
  ...
</form>
```

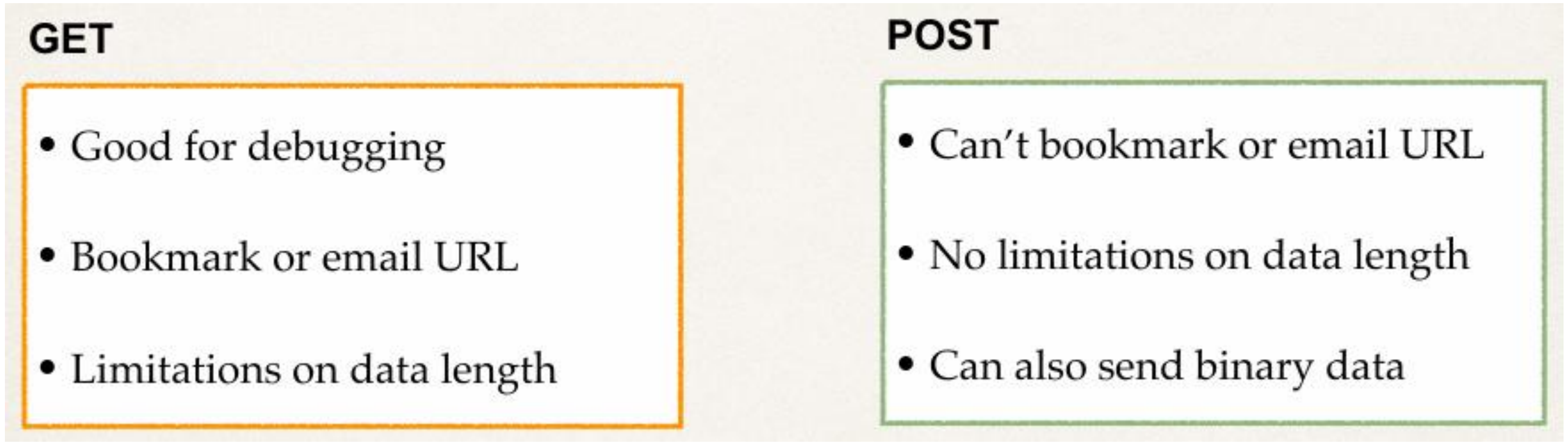
- Form data is added to end of URL as name/value pairs
  - **theUrl?field1=value1&field2=value2...**

Sending Data with POST method



```
<form action="StudentServlet" method="POST">
  ...
</form>
```

Well …. which one???

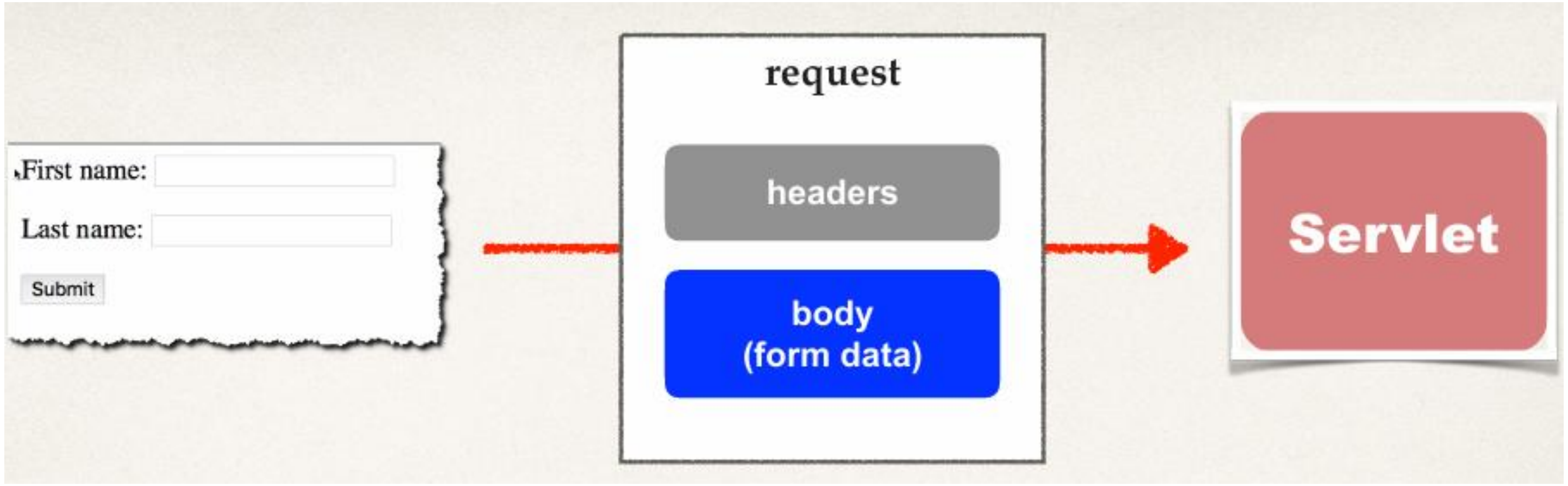


GET

- Good for debugging
- Bookmark or email URL
- Limitations on data length

POST

- Can’t bookmark or email URL
- No limitations on data length
- Can also send binary data


First name:
Last name:
Submit
request
headers
body
(form data)
Servlet