Step 2. Basic Java Web application

1. Explain what happens in the browser when user presses OK button.

   Submit form sends the name value to the action destination 'response.jsp'.

2. Explain what happens in the backend side when user presses OK button.

   Response.jsp process the sent information by using java bean which uses java Class NameHandler. It first sets the property name to the sent value, and use it in the h1 tag, then send the response to the client.

3. Explain how the result of backend processing is delivered back to the browser.

   By http response.

Step 3: Ajax Java Web application

1. Explain the steps that take place when a user types one character in search box.

   onkeyup="doCompletion(); attributes in the input element triggers doCompletion JavaScript function. Whenever the 'readystate is changed 'callback' function is called(req.onreadystatechange = callback;). In the callback, if the request status is ok(200) 'parseMessages' function parses the returned responseXML and present it in the web browser.

2. In what way(s) are the client-server communication steps different compared to an application like the one in the first tutorial?

   Ajax function is run in the background and the web browser doesn't have to wait until the function is executed unlike the previous tutorial. Plus, it only updates parts of the page HTML and doesn't require full page refresh.

Step 4: jQuery Ajax Java Web application

1. How would you deliver the data shown in the browser dynamically from the server?

   With jquery and its UI library you can present the data nicely. To change dynamically generated DOM element you we need to use 'Event Delegation' technic.

Step 5: Reflection

1. What is a .jsp file? Could you replace .jsp with .html file? Why?

   Html is static and can't contain serverside logic. So can't, if html contains some java.

   http://www.oracle.com/technetwork/java/javaee/jsp/index.html

   JavaServer Pages (JSP) technology enables Web developers and designers to rapidly develop and easily maintain, information-rich, dynamic Web pages that leverage existing business systems.

   JSP technology uses XML-like tags that encapsulate the logic that generates the content for the page. The application logic can reside in server-based resources (such as JavaBeans component architecture) that the page accesses with these tags. Any and all formatting (HTML or XML) tags are passed directly back to the response page. By separating the page logic from its design and display and supporting a reusable component-based design, JSP technology makes it faster and easier than ever to build Web-based applications.

   JavaServer Pages technology is an extension of the Java Servlet technology. Servlets are platform-independent, server-side modules that fit seamlessly into a Web server framework and can be used to extend the capabilities of a Web server with minimal overhead,

maintenance, and support. Unlike other scripting languages, servlets involve no platform-specific consideration or modifications; they are application components that are downloaded, on demand, to the part of the system that needs them. Together, JSP technology and servlets provide an attractive alternative to other types of dynamic Web scripting/programming by offering: platform independence; enhanced performance; separation of logic from display; ease of administration; extensibility into the enterprise; and, most importantly, ease of use.

2. What is a bean?

Class with default constructor, standard accessor, mutator, and serializable interface

3. What is the role of WEB_INF folder in your project?

A special directory exists within the application hierarchy named `WEB-INF`. This directory contains all things related to the application that aren't in the document root of the application. **The `WEB-INF` node is not part of the public document tree of the application**. No file contained in the `WEB-INF` directory may be served directly to a client by the container. However, the contents of the `WEB-INF` directory are visible to servlet code using the `getResource` and `getResourceAsStream` method calls on the `ServletContext`, and may be exposed using the `RequestDispatcher` calls.

4. What does Ajax stand for? Explain each part of the acronym.

Ajax (Asynchronous JavaScript and XML) is a client-side script that communicates to and from a server/database without the need for a postback or a complete page refresh. The best definition I've read for Ajax is "the method of exchanging data with a server, and updating parts of a web page - without reloading the entire page.". It is asynchronous because it doesn't wait and use JavaScript to fetch XML data from the server, hence the name.

5. What is a servlet?

A servlet is a Java programming language class that is used to extend the capabilities of servers that host applications accessed by means of a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by web servers. For such applications, Java Servlet technology defines HTTP-specific servlet classes.

The javax.servlet and javax.servlet.http packages provide interfaces and classes for writing servlets. All servlets must implement the Servlet interface, which defines life-cycle methods. When implementing a generic service, you can use or extend the GenericServlet class provided with the Java Servlet API. The HttpServlet class provides methods, such as doGet and doPost, for handling HTTP-specific services.

6. Why is jQuery often used in web applications?

jQuery promotes simplicity. Developers find jQuery intuitive and easy to learn -- this library is built on shorter, simpler code, after all. jQuery makes animated applications. jQuery pages load faster.

7. In what ways is developing Java Web application different from developing PHP?(https://blog.udemy.com/php-vs-java/)

One of the primary structural differences between PHP and Java is the difference between strongly- and weakly-typed languages. Java is a strongly-typed language, meaning it requires explicit statements of intent to function and that it is backed by a compiler. At the highest level, you can think of this as meaning it has strict expectations on how you express inputs and outputs. If these exact expectations are not met, the compiler will fail and the program will not work until errors are resolved.

PHP, in contrast, is weakly typed, essentially meaning it is more flexible and reliant on "common sense programming" in how a task is accomplished. While this may sound more attractive because it requires less formal knowledge, some contend that it complicates certain tasks, particularly in object oriented programming, with its lack of standards.

Ultimately, these characteristics imply that java can be a good choice when development is complicated and relies on teams of professional developers. Rather than making work more challenging, the precisely defined standards allow for easier understanding and greater efficiency among teams. On the other hand, PHP is more accessible to inexperienced programmers. If a web product intends to allow users to work with the code (such as web and blog creation software or small business sites) PHP can fit the bill nicely. It can also be a nice economical and nimble choice for small teams using agreed coding standards and when dealing with simple functions.

PHP: Just put that anywhere

First, an advantage of PHP is that, aside from being free in terms of coding standards, it is free of cost. The open-source language is available on virtually every Web-hosting platform for anyone to use. It is the programming language of the people!

A complex blend for better sleep at night?

While PHP is often cited as being a more economical language, the opposite can be true when complicated object oriented elements are involved. Java is structured to handle this type of programming in a straightforward fashion, whereas PHP can lead to additional scripting, workarounds, and headaches.

Wrap-up

If you take nothing else from this discussion on Java vs. PHP, hopefully you take from it that there is no head-on war between Java and PHP, and there is no need for one. Perhaps you will find that you want to master both in becoming an outstanding, versatile web developer.