Jai Swaminarayan

Web Technology

Date / /

Page 1

Q1 Ajinkya Rathod

Q1 Javascript Cookies

Ans Cookies let you store user information in web pages.

Cookies are data, stored in small text files, on your computer.

When a web server has sent a web page to a browser, the connection is shut down and the server forgets everything about the user.

Cookies are invented to solved the problem about how to remember information about user.

They are stored in key-value pairs.

> username = John Doe.

* The cookie string looks like Document.Cookie.

To set a new cookie use:-

document.cookie = xy2 + expire time + path;

* To get Cookie use,

let a = getCookie("a");

Q1

Q3 Push

- The push() method adds new items to end of array.
- push() changes the length of array and returns new length

Example

```
const cars = ["Volvo", "BMW"]
cars.push("Mercedes");
```

Now cars will have ->
[Volvo, BMW, Mercedes]

Unshift

The unshift method adds new items to the beginning of an array and returns the new length

Note: Unshift overwrites the original array

Example

```
const array1 = [1, 2, 3];
console.log(array1.unshift(6));
console.log(array1);
```

O/P :-

[6, 1, 2, 3]

Difference

The only difference is that unshift() adds the element at 0 index and all values get shifted by returning length of array.

Q2

Q1(b) Request Dispatcher.

The request dispatcher interface provides the facility of dispatching the request to another resource it may be html, servlet or jsp.

This interface can also be used to include the content of another resource also.

It is one way of servlet collaboration

Methods

-> public void forward(req, res) {
..
}

-> public void include(req, res) {
..
}

It defines an object that receives request from the client and sends them to any resource such as servlet, HTML or JSP on server.

It creates "Request Dispatcher" object which is used as a wrapper around given resource located at particular path by a map.

Date / /
Page 1

Q2

Q The most critical difference between @include and <jsp:include> is that the include directive is processed at translation time but include action that is <jsp:include> is processed at request time

2. Include uses relative and absolute path but include action always uses only relative path

3. include can have contents of resource like HTML, CSS. but cannot do dynamic resources.

<jsp:include> can do dynamic results.

4. In include, we cannot ~~include~~ pass request or response to call jsp and it is possible in <jsp:include>

5. In include, parameter can not be passed.

In include action, we can pass params like request, response etc.. using <jsp: params> tags

Q3 Scopes of Bean

(1) Singleton

The scopes of bean definition to single instance per IOC container

(2) Prototype

The scope of single bean definition to have any number of object instances.

2. Request

This scopes a bean definition to an HTTP request.

4. Session

Bean of HTTP Session

5. Global Session

Bean definition to global HTTP Session.

Exampl

<beans xml=" http: ------
xmlns:xsi =" ------

xsi:schema locations = ----

">

<bean id = " hello world " class =
" com.ajinkya.Hello "
scope = " prototype ">

</bean>

</beans>

Q3 (2)

	Send Redirect	Forward
1.	This works on client side	This works on server side.
2.	It sends new request	It sends same request and response objects to another servlet.
3.	It can be used within and outside the server.	It can work within server only.

To simply explain the difference

→ response.send Redirect("login.jsp");
does'nt prepend the context path but

→ request.getRequestDispatcher("login.jsp"). forward(...);
will prepend the context path

Moreover, Redirect request is used to redirect to resources to different severs or domains.

But, Forward Request is used to forward to resources available within server from where the call is made.

This transfer of control is done by container internaly and client (i.e browser) is not involved.