

## 2) HttpSession:

HttpSession is used to maintain a user's state and data across multiple requests in a web application. It helps manage session information on the server side. A session is created when getSession() is called on an HttpServletRequest object. If a session already exists for the client, getSession() returns that session; otherwise, it creates a new one.

Method	Description
void setAttribute(String name, Object value)	Sets an attribute in the session
Object getAttribute(String name)	Retrieves the attribute with the specified name
void removeAttribute(String name)	Removes the attribute with the specified name.
Enumeration<String> getAttributeNames()	Returns an enumeration of the attribute names in the session.
void putValue(String name, Object value)	Adds an attribute to the session.

## Contd...

Method	Description
Object getValue(String name)	Retrieves the value of the specified attribute
void removeValue(String name)	Removes the specified attribute
String[] getValueNames()	Gets all attribute names in the session.
void invalidate()	Removes the Session (Session expires)
ServletContext getServletContext()	Returns the ServletContext to which this session belongs.
HttpSessionContext getSessionContext()	Returns the HttpSessionContext for the session

### Creating a Session:

we can use getSession() method from HttpServletRequest to create an HttpSession instance.

**Syntax**    public abstract javax.servlet.http.HttpSession getSession();  
              public abstract javax.servlet.http.HttpSession getSession(boolean);

### 3) URL Rewriting:

URL rewriting is the process of adding parameters to a servlet URL pattern.

#### Syntax:

servlet-url-pattern?parameter=value&parameter=value&...

#### Special Symbols:

"?": Separates the URL pattern from the parameters.

"&": Separates multiple parameters.

#### Example:

<https://www.google.com/search?q=gmail&...>

## 4) Hidden Form Field:

Hidden form fields are created using `<input type="hidden">` in the `<form>` tag in HTML. They store data that is not displayed to the end user but can be sent with form submissions.

### Syntax:

```
<form action="url" method="POST/GET">  
  <input type="hidden" name="name" value="value">  
  ...  
</form>
```

**Note:** **URL Rewriting** and **Hidden Form Fields** are sub-techniques in the session tracking process. They are used not only for session management but also for passing data between servlets in a web application.

# Cookies Vs HttpSession Vs URL rewriting Vs hidden form fields

Feature	Cookie	HttpSession	URL Rewriting	Hidden Form Fields
Storage Location	Client-side (stored in browser)	Server-side	Client-side (appended to URL)	Client-side (hidden fields in forms)
Persistence	Until expiry or deletion by the user	Until session expires or is invalidated	Until the user navigates away or session expires	Only persists during form submission
Security	Limited security; accessible by client and server	More secure; stored only on the server	Less secure; session ID visible in URL	Not very secure; visible in HTML source
Usage	Store session data across pages; supports persistent data	Suitable for sensitive info (e.g., user IDs)	Track session if cookies are disabled; only for GET requests	Maintain data across form pages during submission