

GROUP

2

PRESENTATION ABOUT SESSION

Session roles

- **Sessions** are used in web applications to maintain state across multiple HTTP requests. Since HTTP is stateless, **sessions** allow the server to remember the user's state and data between requests.

The main roles of sessions includes:

- **User Identification:** Storing user-specific information like login status, preferences, and data across multiple requests.
- **Data Persistence:** Keeping track of user interactions on the site such as shopping cart contents or filled-in forms.
- **Security Management:** Providing secure user authentication and authorization as per user roles.

HttpSession getSession() VS HttpSession getSession(boolean create)

The getSession() method can be called in two main ways

- **HttpSession getSession():**
 - This method retrieves the current session associated with the request, creating a new session if it does not exist.
- **HttpSession getSession(boolean create):**
 - If create is **true**, it behaves the same way as the no-argument method, returning the current session or creating a new one.
 - If create is **false**, it returns the current session if it exists but returns null if no session is associated with the request. This allows the application to check for session existence without creating one.

the methods used with HttpSession interface

The common methods that can be used with the HttpSession interface include:

- **String getId():** Returns the unique identifier for this session.
- **long getCreationTime():** Returns the time when this session was created.
- **long getLastAccessedTime():** Returns the last time that this session was accessed.
- **int getMaxInactiveInterval():** Returns the maximum time interval between client requests before this session will be invalidated.

the methods used with HttpSession interface (cont.)

- **void setMaxInactiveInterval(int interval):** Sets the maximum time interval before the session is invalidated.
- **Object getAttribute(String name):** Returns the value to which this session associates the specified name.
- **void setAttribute(String name, Object value):** Binds an object to this session, using the specified name.
- **void removeAttribute(String name):** Removes the object bound with the specified name.
- **void invalidate():** Invalidates this session and unbinds any objects bound to it.

Q4. PRACTICE