1. Explain MVC architecture

The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. It is a design pattern that separates the business logic, presentation logic and data. Controller acts as an interface between View and Model. Controller intercepts all the incoming requests. MVC is one of the most frequently used industry-standard web development framework to create scalable and extensible projects.

Model

The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data. For example, a Customer object will retrieve the customer information from the database, manipulate it and update it data back to the database or use it to render data.

View

The View component is used for all the UI logic of the application. For example, the Customer view will include all the UI components such as text boxes, dropdowns, etc. that the final user interacts with.

Controller

Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output. For example, the Customer controller will handle all the interactions and inputs from the Customer View and update the database using the Customer Model. The same controller will be used to view the Customer data.

2. What is HTTP session?

HTTP sessions is an industry standard feature that allows Web servers to maintain user identity and to store user-specific data during multiple request/response interactions between a client application and a Web application.

On client's first request, the Web Container generates a unique session ID and gives it back to the client with response. This is a temporary session created by web container. ... The Web Container uses this ID, finds the matching session with the ID and associates the session with the request.

3. What is Expression language?

 Expression Language, which is used for accessing the data and makes it possible to easily access the application data stored in the JavaBean's components and other objects like request, session, and application etc. It was a scripting language which allowed access to Java components (JavaBeans) through JSP.

4. Scopes of Expression language-

* PageScope constrains access to the current JSP.
* RequestScope constrains access to the current request. The variable is not accessible to other requests, even if they originate from the same user session.
* SessionScope enables access to the user session. The variable persists for the duration of that session.
* ApplicationScope enables access to all application resources and users. The state of the variable is the same for all users of the application.

The default scope is pageScope, which is used when the variable declaration omits scope specification. You should specify the narrowest scope possible, in order to minimize the chance that two variables with the same name have overlapping scopes.

5. What is page directive?

It provides attributes that get applied to entire JSP page. It defines page dependent attributes, such as scripting language, error page, and buffering requirements. It is used to provide instructions to a container that pertains to current JSP page.

Syntax- <%@page..%>