SAPUI5/FIORI

INTERVIEW QUESTION AND ANSWERS

1. What is SAPUI5?

SAPUI5 is a technology framework used for building web applications with a user-friendly interface. It's based on JavaScript and provides pre-built controls and tools for creating responsive and interactive applications.

2. What is Fiori?

Fiori is a design concept by SAP that focuses on creating user-friendly, consistent, and visually appealing user interfaces for SAP applications. It aims to enhance user experience across different devices and roles.

3. What is the difference between UI5 and Fiori?

UI5 (SAPUI5) is the technology framework used to build applications, while Fiori is a design principle that focuses on how applications should look and behave. UI5 is the toolkit, and Fiori is the design guideline.

4. Bootstrapping? Explain each value of bootstrapping.

Bootstrapping in UI5 is the process of initializing the framework and application. There are three values for bootstrapping:

- Implicit: The framework initializes automatically when you include the library in the HTML file.

- Explicit: You manually call a script to initialize the framework.

- Combined: A mix of implicit and explicit bootstrapping, used for advanced scenarios.

5. Flow of SAPUI5 application execution?

Application flow includes:

- Index.html loads the UI5 library.

- Application's "Component.js" is executed.

- Component configuration, router, and models are initialized.

- Views and controllers are loaded based on the routing.

6. UI5 application project structure? Explain each one.

A UI5 project usually has:

- index.html: Entry point, includes UI5 library.

- Component.js: Initializes the application.

- manifest.json: Configuration for the app.

- controller: Handles view logic.

- view: Defines the UI structure.

- model: Manages data.

7. Types of Views?

There are four types of views in UI5:

- XMLView: Using XML for defining views.

- JSONView: Using JSON notation for views.

- JavaScript View: Using JavaScript code to define views.

- HTMLView: Uses HTML Markup to define the structure of the view.

8. Types of Models used in UI5?

Commonly used models in UI5 are:

- JSON Model: Stores data in JSON format.

- XML Model: Represents data in XML format.

- OData Model: Connects to OData services to fetch and manipulate data.

- Resource Model: used for internationalization (i18n) purposes to support multiple languages.

9. What is the Default Model?

The default model is automatically created when the application initializes. It's usually a JSON model and can be accessed globally within the app without explicit naming.

10. Types of Binding Modes? Default Binding for the Models?

Binding modes define how data is synchronized between the model and the UI. There are three binding modes:

- OneWay: Updates UI when model changes.

- TwoWay: Keeps UI and model synchronized.

- OneTime: Updates UI only once from the model.

For most models, the default binding mode is TwoWay, ensuring changes in UI reflect in the model and vice versa.

11. Types of Binding?

Property Binding: This binds a property of a control to a value in the model. Changes in the model will reflect in the control and vice versa.

Aggregation Binding: It binds an aggregation of a control (like items in a list or cells in a table) to a list in the model. It allows dynamic population of controls based on the model data.

Element Binding: This binds a single entity from the model to a control. It's useful when you want to bind the entire context of a control to a single entity in the model.

Expression Binding: It allows defining complex binding expressions using a scripting language that evaluates to a value. This can be used for calculated values or conditional formatting.

12. Combobox setting binding values?

You set binding values for a ComboBox using its "items" aggregation. You bind the aggregation to a model path containing the values you want to display in the ComboBox.

13. How to set the particular element of ComboBox as the default value?

You can set the default value of the ComboBox by binding its "selectedKey" property to the model path of the desired value.

14. How to add secondary values to ComboBox?

You can add secondary values to a ComboBox by using its "items" aggregation. Each item can have text for display and additional text for secondary information.

15. Multi ComboBox uses? Give a real-time scenario.

A MultiComboBox allows users to select multiple items from a dropdown list. A real-time scenario could be in an e-commerce website's filter options where users can select multiple categories to narrow down their search.

16. Different types of values for the input control?

Input control can have values like:

- Text: Plain text input.
- Number: Numeric input.
- Date: Date selection input.
- Password: Masked input for passwords.
- Email: Input for email addresses.

17. Difference between a table and a list?

Table:

- Displays data in rows and columns.
- Often used for structured data representation.
- Supports sorting, filtering, and grouping.

List:

- Displays data in a single column.
- Typically used for less structured or simple data.

- Commonly used for displaying lists of items.

18. Difference between sap.ui.table and sap.m.table?

sap.ui.table:

- Used for responsive tables with advanced features.

- Ideal for complex data presentation.

- Supports features like sorting, filtering, and row selection.

sap.m.table:

- Used for simple mobile-friendly tables.

- Designed for lightweight, straightforward data display.

- Suitable for mobile applications.

19. Formatters? Use of formatter?

Formatters are functions used to format values before they are displayed in UI elements. They are often used to convert data into a user-friendly format. For example, a formatter can convert a date object into a more readable date string.

20. Expression Binding? How to enable the expression Binding?

Expression Binding allows you to directly write expressions in the binding syntax. To enable expression binding, you need to use curly braces with a colon, like `{= expression}`. This enables more complex and dynamic bindings. For example, `{= ${price} > 100 ? 'Expensive' : 'Affordable'}` would show different text based on the value of `price`.

21. Fragments? Use of fragments?

Fragments are reusable UI components in SAPUI5 that define a part of the user interface. They can be reused across different views or controls. Fragments are useful for creating complex UI structures that need to be included in multiple places.

22. Dynamic object available at runtime of SAPUI5 application?

The JSONModel is often used to hold dynamic data in a SAPUI5 application. This model can be updated at runtime to reflect changes in the data, and the UI components bound to this model will automatically update accordingly.

23. What are the OData methods to call OData Service in a UI5 application?

In UI5, you can use the `create`, `read`, `update`, and `remove` methods to interact with OData services using the `ODataModel` class.

24. What are global variables? How to define them?

Global variables are variables that are accessible throughout the entire application. In JavaScript, you can define a global variable by omitting the `var`, `let`, or `const` keywords before declaring a variable. However, it's recommended to use namespaces or modules to avoid cluttering the global scope.

25. What is neo_app.json used for?

`neo_app.json` is used in SAP Cloud Platform applications. It's a configuration file that defines properties and settings for the application's deployment and runtime environment. It specifies routes, resources, and various application-specific settings for the SAP Cloud Platform environment.

26. Use of Component.js file?

The Component.js file is used to define and configure the application's main component. It initializes the application, defines metadata, and sets up routing and models.

27. Use of Component_preload.js file?

The Component_preload.js file is an optimized and compressed version of the application's resources. It's used to improve the loading performance by reducing the number of separate requests made to the server.

28. Dialogs use?

Dialogs are used to display temporary UI content over the main view. You can create a dialog using the Dialog control, like:

29. Pagination?

Pagination is the process of dividing content into smaller chunks, or pages, to improve readability and navigation. In UI5, it's often used to display large sets of data in tables or lists by showing a limited number of items per page.

30. Navigation?

Navigation in UI5 refers to moving between different views or pages within an application. It's usually managed using the `Router` class, which enables navigation based on defined routes and parameters.

31. Routing with parameter passing from one view to another?

You can pass parameters between views using the `Router`:

```
this.getOwnerComponent().getRouter().navTo("targetView", {
  param1: value1,
  param2: value2
});
```

32. Different Fiori Design principles?

Some Fiori design principles are:

- Role-based: Tailored for specific user roles.
- Delightful: User-centric and visually appealing.
- Simple: Minimize complexity and provide clear tasks.
- Coherent: Consistent design and behavior.
- Responsive: Works well on various devices.

33. Fiori Configuration?

Fiori configuration involves setting up the environment, applications, and user roles to ensure that SAP Fiori apps work effectively. It includes tasks such as defining app catalogs, configuring launchpad, roles, authorizations, and user-specific settings.

34. What is a semantic object? Benefits?

A semantic object is a concept in SAP Fiori that represents a business entity or process, like a customer, product, or sales order. It allows deep linking and navigation to specific views or apps related to that object. Benefits include consistent navigation, contextual information, and improved user experience.

35. What do tile, catalog, and group indicate?

In the Fiori launchpad:

- Tile: Represents an app or link on the launchpad's home screen.
- Catalog: Organizes related tiles for a specific role or user.
- Group: Groups tiles within a catalog for further organization.

36. Types of tiles in Fiori application? Explain?

There are several types of tiles:

- Static Tile: Displays static content like an icon and title.
- Dynamic Tile: Displays dynamic content from a data source.
- Tile Catalog: Organizes tiles for specific user roles.
- KPI Tile: Displays key performance indicators.
- App Launcher Tile: Launches an app or web link.

37. Types of Fiori apps? Explain?

Fiori apps can be categorized into three types:

- Transactional Apps: Perform specific tasks like creating orders.
- Analytical Apps: Provide insights through data analysis.
- Fact Sheets: Display detailed information about a specific object, like a customer or product.

38. Cross-app navigation? Syntax?

Cross-app navigation enables navigation from one Fiori app to another. You can use the `CrossApplicationNavigation` service:

```
var oCrossAppNav =
sap.ushell.Container.getService("CrossApplicationNavigation");

oCrossAppNav.toExternal({ target: { semanticObject: "Object", action: "display" }
});
```

39. What are extension points in a standard Fiori app? View extension point and controller extension point?

Extension points are predefined hooks in standard Fiori apps that allow you to add custom code, views, or controller logic without modifying the app's original code. View extension points allow you to add UI elements, while controller extension points allow you to modify app behavior.

40. Configuring Analytical app? Prerequisites for configuring an analytical app?

Configuring an analytical app involves setting up KPIs, charts, and data sources. Prerequisites include defining the data model, having relevant OData services, ensuring appropriate authorizations, and having the necessary backend configurations.

41. What is a CDS view?

A CDS (Core Data Services) view is a definition in SAP that represents a virtual database table enriched with business logic and data definitions. It's used to model data and provide a standardized way of defining and consuming data structures in various SAP applications.

42. How to check the version of a SAPUI5 application?

You can check the version of a SAPUI5 application by accessing the browser's developer console (usually by pressing F12), and then typing `sap.ui.version` in the console and pressing Enter.

43. Smart controls?

Smart controls in SAPUI5 are pre-configured UI controls that are designed to automatically adjust their behavior based on the data model and annotations from the backend. They help simplify application development by providing automatic sorting, filtering, and other functionalities.

44. Custom Controls?

Custom controls in SAPUI5 are controls that developers create to extend or customize the standard set of UI5 controls. They allow developers to build unique UI elements tailored to their application's specific requirements.

45. Deploying SAPUI5 application on-premise?

Deploying a SAPUI5 application on-premise involves configuring your server or SAP system to host the application's files and resources. You typically upload your application to the server, set up routing, and ensure that the necessary backend services are accessible.

46. Deploying application to the cloud?

To deploy a SAPUI5 application to the cloud, you package the application's resources, including HTML, JavaScript, and other assets, and then upload them to a cloud platform like SAP Cloud Platform or other cloud providers. You may need to configure routing, authentication, and permissions.

47. What is a destination in WebIDE?

In SAP Web Integrated Development Environment (WebIDE), a destination is a configuration that defines the connection details to a remote system or service. It's used to connect to external services, such as OData services or Git repositories, from within the WebIDE.

48. What does SAP version indicate?

The SAP version indicates the specific version and release of SAP software that you are using. It helps in understanding the features, improvements, and compatibility of the software.

49. Filtering, Sorting, and Grouping?

Filtering is the process of selecting specific data based on certain criteria. Sorting is arranging data in a particular order, usually ascending or descending. Grouping involves categorizing data based on common attributes, often used to create summaries or hierarchies.

50. Difference between OData V2 and V4?

OData V2 and V4 are versions of the OData protocol:

- V2: More established, widely used, and supported. It has a richer ecosystem of client libraries and tools. It supports two-way data binding and is suited for real-time applications.

- V4: More advanced, focused on simplicity, performance, and scalability. It offers enhanced query options and better support for mobile devices. However, the adoption of V4 might require updating client libraries and tooling.

Certainly, here are simple and easy-to-understand answers to the questions you've listed:

51. How to identify the application running device?

You can use the sap.ui.Device API to identify the device on which the application is running. It provides properties like system.phone, system.tablet, and system.desktop to detect the type of device.

52. How to declare global variables?

In JavaScript, global variables can be declared without the \`var\`, \`let\`, or \`const\` keyword. However, it's recommended to limit the use of global variables and to encapsulate them in a namespace to prevent naming conflicts.

53. Difference between sap.ui.define and sap.ui.require?

sap.ui.define is used to define modules (classes, controls, etc.), while sap.ui.require is used to load modules that have been defined previously. define also supports asynchronous loading and dependency management.

54. Can one view have multiple controllers?

No, a view in SAPUI5 is bound to a single controller. However, a controller can handle multiple views by managing their logic and interactions.

55. Can one controller have multiple views?

Yes, a controller can be associated with multiple views. This is useful when you have related views that share similar logic.

56. How to increase the performance of an application?

To increase application performance:

- Optimize code for efficiency.
- Reduce unnecessary DOM manipulations.
- Use data binding for dynamic content.
- Minimize network requests and optimize loading.
- Utilize caching for repetitive data.
- Implement responsive design for efficient rendering.

57. Different charts in SAPUI5?

SAPUI5 provides various chart types like:

- Line Chart
- Bar Chart
- Column Chart
- Pie Chart
- Donut Chart
- Scatter Plot Chart
- Bubble Chart
- Radar Chart
- Stacked Chart

58. Expression binding with syntax?

Expression binding allows you to write expressions directly in your binding statements. Syntax example:

```xml
<Text text="{= ${quantity} > 10 ? 'High' : 'Low'}" />
```

59. Master-detail routing concept?

Master-detail routing involves navigating from a list (master) to a detailed view (detail) while maintaining context. It's often used to show details of an item selected from a list. The detail view is populated based on the selected item's data.

LETS CONNECT!

DISCOVER THE ENDLESS POSSIBILITIES OF SAP TOGETHER.

RONAK SHUKLA
SAP Technical Consultant