| **Topic** | **Elementary** | **Basic** | **Advanced** | **Expert** |
| --- | --- | --- | --- | --- |
| **Fiori Elements** | | | | |
| **Extension capabilities** | 1. Understand why extensibility in SAP’s business applications matters. Ability to give some real use cases where it matters. 2. Be aware of UI Adaptation editor (Layout editor) capabilities | 1. Understand the **stable ids** concept. 2. Know the main aspects of the **app extension via component / manifest** configuration:    1. Controller extension. Hooks in the standard controller.    2. Controller replacement.    3. View extension.    4. View modification.    5. View replacement.    6. Extension points in view. 3. Know how to implement **localized texts** for extended apps, i18n enhancement. 4. Know **Limitations, caveats, supportability**. 5. Be aware of [Extensibility Information in Documentation](https://wiki.wdf.sap.corp/wiki/display/fiorisuite/Extensibility+Information+in+Documentation) 6. Be aware of **SAPUI Flexibility features** (run-time adaptation). | \_empty\_ | \_empty\_ |
| **Build & deployment** | 1. Understand the **difference** between the app in **local environment** and app in **deployed one**. Understand the features that the app should possess in order to successfully operate in a deployed environment. 2. Be aware of the web app **build approaches** (grunt, gulp, **SAP IBSO SAPUI5 Tools**). | 1. Understand what a **Component-preload** file is. 2. Be aware of the main features of **UI5 tooling**:    1. Main tasks.    2. Ability to implement custom middlewares.    3. Building types & aspects. 3. Be aware of **deployment to SCP Neo** procedure. 4. Be aware of **deployment to SCP CF** procedure. 5. Be aware how to work with SAP Project Portal and Gerrit | 1. Be aware of the web app **build approaches** (grunt, gulp, UI5 tooling). 2. Be aware of deployment to **ABAP repository via transport request (Web IDE)** procedure. 3. Be aware of deployment to **ABAP repository via transaction** in SAP LogOn. 4. Know the features of **SAP CI pipelines**. 5. Understand what the **Nexus** is. | 1. Be aware how to create JaaS instance and configure pipeline to perform xMake jobs, ABAP deploy, Checkmarx |
| **SAPUI5 ABAP repository & the ABAP Back-End infrastructure** | 1. Understand what the **SAP Gateway** is. Its purpose, main use cases. | 1. Understand the basic key terms of the **big picture of the landscape (**SAP Systems, mandant/client, Gateway, ABAP backend, DB, SADL, CDS, BOPF)**.** 2. Know what is a **package, BSP**. 3. Be aware of **Downloading / Uploading reports (SE38)**. 4. Know the way to do the **cache clearage**. 5. Understand what is a **transport**? **Workbench** & customizing transport request types. | 1. Understand the **Cache Buster** mechanism. 2. Know what is an **App Index**. 3. Be aware of **SAPUI5 Text repository**. 4. Be aware of the **main transactions**:    1. SEGW    2. SE38    3. SE09    4. SE80    5. SU01    6. SICM    7. STFK    8. SEMOBJ    9. LPD\_CUST    10. SICF | 1. Understand the whole architecture of the SAP business application with UI5 as a client. |
| **Misc** | 1. UI5 Debugging techniques:    1. Know the **UI5 inspector** features.    2. **Debugging** on runtime through the **global variables**.    3. Know how to **get control globally** on the runtime.    4. Be aware of the possibility to debug UI5 sources. Know the steps of the procedure.    5. Know how to enter the **debug mode**, make use of **assistant diagnostics**.    6. Know the **logging** possibilities. 2. Be aware of **content density**. 3. Be aware of **launchpad concept** basics:    1. **Shell bar** concept.    2. Understand **tiles**, their **types**. | 1. Be aware of **VizFrames**, know the **basic chart types** and their purpose. Know the basics of data binding for visualisation. 2. Be aware of apps troubleshooting techniques (from docu). 3. Be aware of the **reuse library** approach. 4. Know the ways to transfer data between multiple pages in the app. 5. Be aware of **device adaptation** aspects. 6. Know the **messages handling** concept:    1. Message Model.    2. Message Manager.    3. Validation messages:       1. From BE.       2. From UI (data types, custom).    4. Message button in footer, message popover. 7. Know what is a **target mapping**. 8. Be aware of an event bus and its use cases. | 1. Be aware of **draft concept**, **locking mechanism**, **etags.** 2. Understand the **theming mechanism**. Be aware of how the **custom theme** can be created. Theme parameters reusage. 3. Know the **main rules** of adding **custom CSS** or **overriding** the standard CSS (in an emergency case). 4. Understand the **SpreadSheet export** mechanism. 5. Be aware of **variant management** technology:    1. SmartTable, SmartFilterBar, SmartChart.    2. Custom implementation. | 1. Be aware of **native app** implementation & **Cordova** support. 2. Know the apps optimization capabilities:    1. Performance improvement.    2. Modularization approach.    3. Cache buster usage. 3. Understand **Drag & Drop** mechanism. 4. Be aware of Rich Text Editor and TinyMCE. |