

Hands-on Lab: Fusion Development with ABAP Cloud in SAP Build

asUG Tech Connect

In partnership with **SAP TechEd**

Agenda

Exercise 1

- Start your development by creating an ABAP project in the SAP Build Lobby

Exercise 2

- Create a Shopping Cart business object, projection and service with the Graphical Modeler in SAP Build Code

Exercise 3

- Implement transactional behavior in ADT, create validations and determinations

Exercise 4

- Create, preview and adapt a SAP Fiori elements application

Get started

asUG Tech Connect

In partnership with SAP TechEd

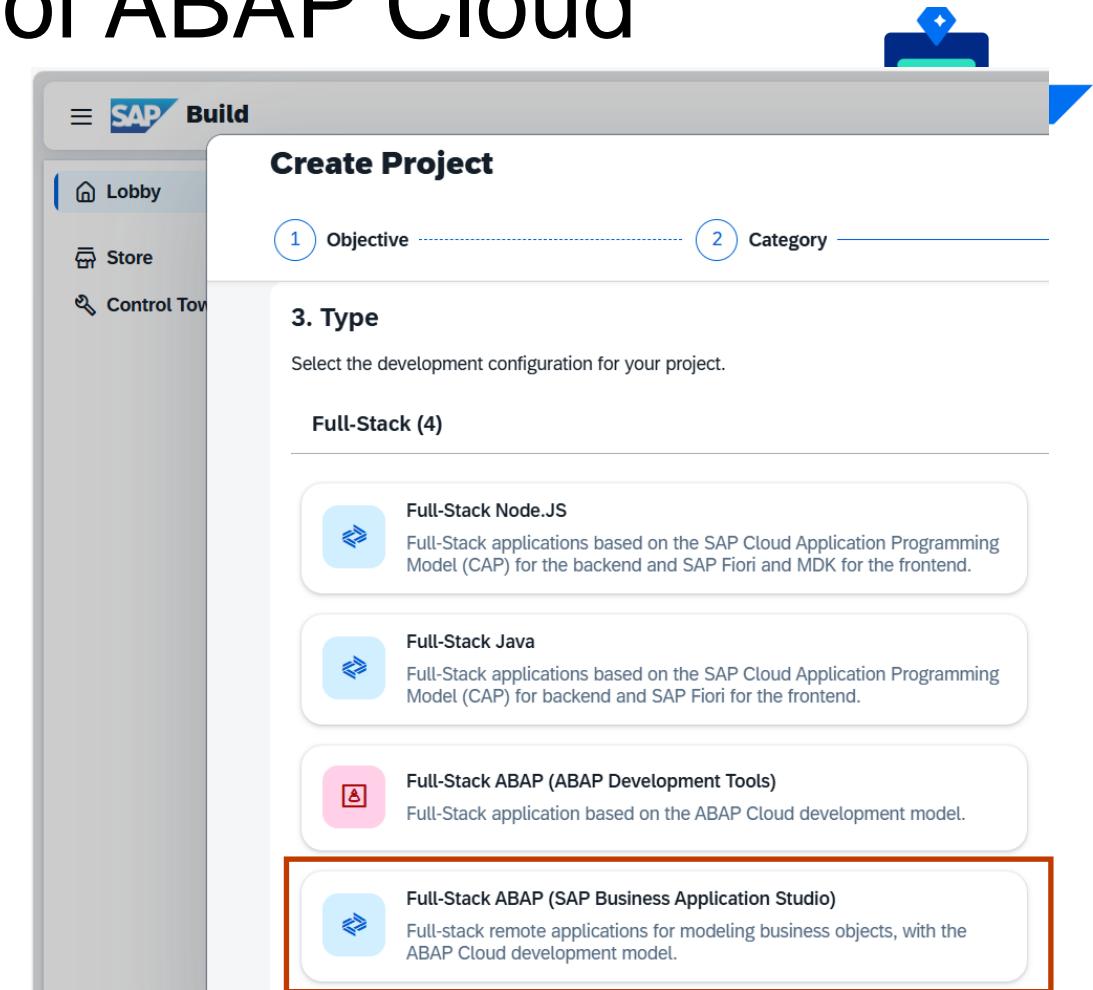
Introduction

asUG Tech Connect

In partnership with SAP TechEd

What you will do in this hands-on workshop – Practice fusion development of ABAP Cloud

1. SAP Build Lobby - Create a Full-Stack ABAP project (SAP Business Application Studio)



What you will do in this hands-on workshop – Practice fusion development of ABAP Cloud

1. SAP Build Lobby - Create a Full-Stack ABAP project (SAP Business Application Studio)
2. SAP Business Application Studio - Graphical Modeler



The screenshot shows two entity definitions in the SAP Business Application Studio Graphical Modeler:

Cart *Business Object*

OrderUuid	abap.raw(16)	
Ab	Currency	abap.cuky
Ab	Notes	abap.char(100)
Ab	OrderID	abap.numc(8)
1d	RequestDeliveryDate	abap.datn
12	TotalPrice	abap.curr(11,2)

Item *Entity*

ItemUuid	abap.raw(16)	
1d	ParentUuid	abap.raw(16)
Ab	ItemID	abap.numc(8)
Ab	OrderedItem	abap.char(40)
Ab	Quantity	abap.numc(4)
12	ItemPrice	abap.curr(11,2)
Ab	Currency	abap.cuky
12	ItemUnitPrice	abap.curr(11,2)

At the bottom right, there is a toolbar with icons for zooming, search, and other functions. At the bottom center, it says "Layout: German".

What you will do in this hands-on workshop – Practice fusion development of ABAP Cloud

1. SAP Build Lobby - Create a Full-Stack ABAP project (SAP Business Application Studio)
2. SAP Business Application Studio – Graphical Modeler
3. ABAP Development Tools - Implementation of Business Logic



A screenshot of the SAP ABAP Development Tools interface. The main window shows a code editor with ABAP code. A red box highlights a section of code from line 36 to line 45. The code defines a validation method and several draft actions, including a 'Prepare' action that triggers the validation. Below the code editor is a navigation bar with icons for file operations like 'New', 'Open', 'Save', etc. The bottom of the screen shows the SAP logo and the text 'SAP ABAP Development Tools'.

```
35
36 validation validateRequestDeliveryDate on save { create; field RequestDeliveryDate; }
37
38 draft action Activate optimized;
39 draft action Discard;
40 draft action Edit;
41 draft action Resume;
42 //draft determine action Prepare;
43 draft determine action Prepare
44 {
45   validation validateRequestDeliveryDate;
46 }

64
65 association _Items { create; with draft; }
66
67 }
```

A screenshot of the SAP ABAP Development Tools interface, specifically the Predict RAP Business Logic feature. The code editor shows a partial method definition:

```
METHOD validateRequestDeliveryDate.
ENDMETHOD.
```

A tooltip window titled "Predict RAP Business Logic" is open over the method name, providing information or suggestions related to the prediction process.

What you will do in this hands-on workshop – Practice fusion development of ABAP Cloud

1. SAP Build Lobby - Create a Full-Stack ABAP project (SAP Business Application Studio)
2. SAP Business Application Studio – Graphical Modeler
3. ABAP Development Tools - Implementation of Business Logic
4. SAP Business Application Studio - Create, preview and adapt a SAP Fiori elements application



The screenshot shows a Fiori application interface. At the top, there's a header with the SAP logo and the text "Preview for Fiori Elements App". Below the header, there's a dropdown menu set to "Standard". The main area features a table with four columns: "Notes", "OrderID", "RequestDeliveryDate", and "TotalPrice". Above the table, there's a search bar with the placeholder "All". To the right of the table, there are buttons for "Create", "Delete", and other actions. At the bottom of the screen, there's a message: "No data found" followed by the sub-instruction "Try adjusting the search or filter criteria." A cartoon character icon of a person looking through binoculars is positioned in the center of the table area.

Exercise 1:

Start your development by creating an ABAP project in the SAP Build Lobby

asUG Tech Connect

In partnership with **SAP TechEd**

Exercise 1: Create a Full-Stack ABAP (SAP Business Application Studio) Project

Create a Full-Stack ABAP (SAP Business Application Studio) Project

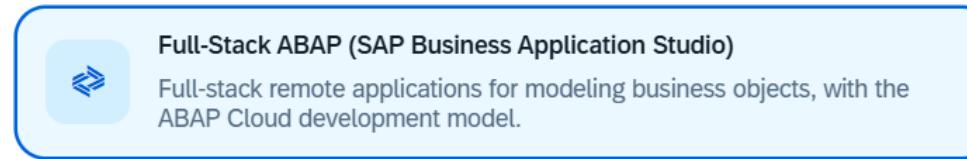
Connect to backend system

H63

Use a pre-created package

ZAD163-A##

Create and start Dev Space
in SAP Business Application Studio



Create Project

4. System and Package

Specify the ABAP system and package to be used

Before proceeding, please make sure you have all the [prerequisites](#)

System

System: *
H63

System ID: H63
System Client: 100

Package

Existing
ZAD163_Z01

New

Create Project

6. Name

Give your project a name.

Name: *
ZAD163_Z01

Description:
Project for ZAD163_Z01

CSN File Name: *
ZAD163_Z01

Dev Space: *
ZAD163_Z01

You can run up to two dev spaces at a time. For more options, go to the [Dev Space Manager](#).

Previous **Review** Cancel

Exercise 2 :

Create a Shopping

Cart business

object,

projection and

service with the

Graphical Modeler

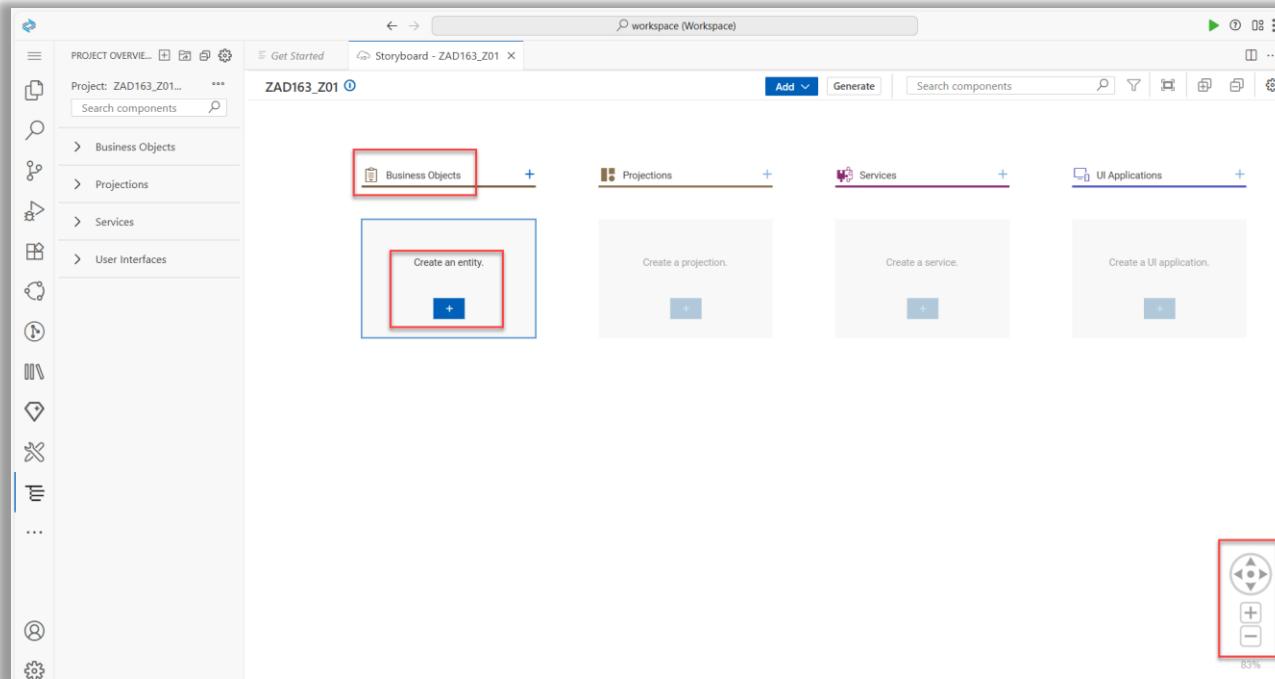
in SAP Build Code

asUG Tech Connect

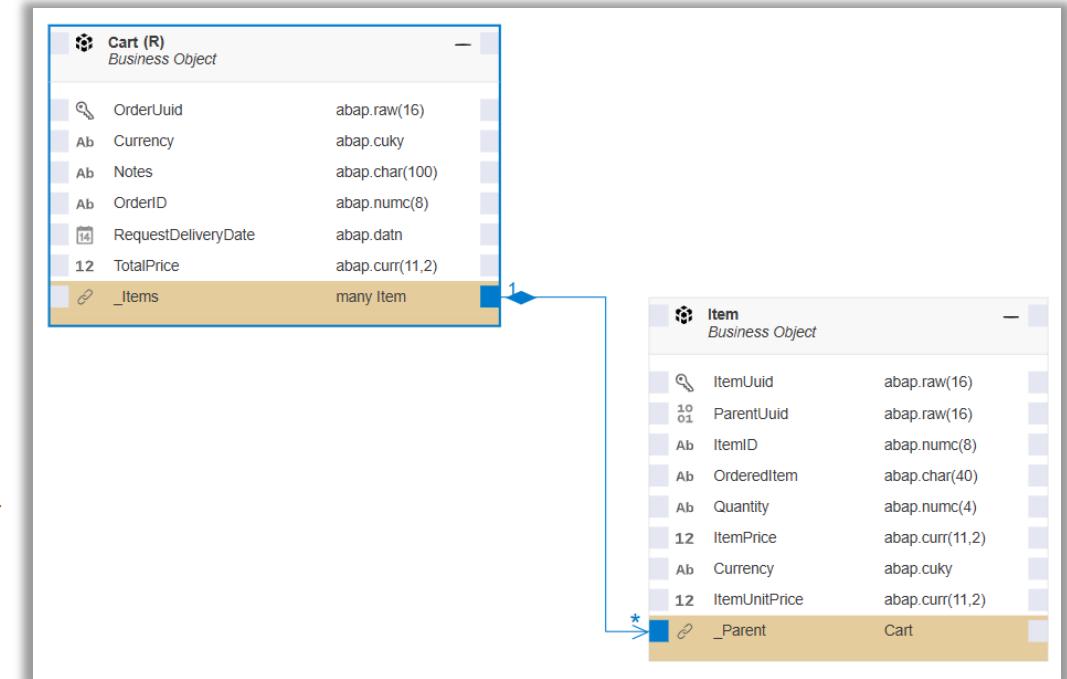
In partnership with **SAP TechEd**

Exercise 2: Model the ShoppingCart Business Object

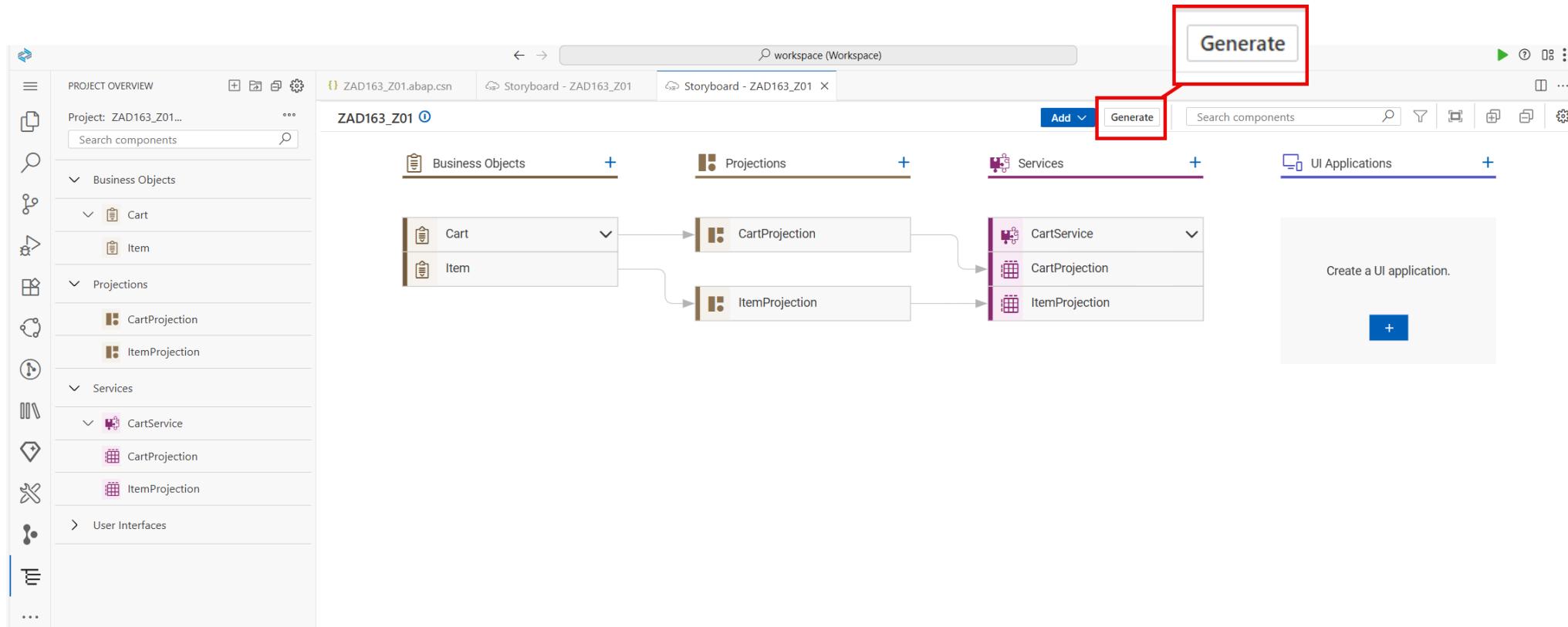
Storyboard



Graphical Modeller



Exercise 2: Add Projection and Service Layer, Generate ABAP artefacts

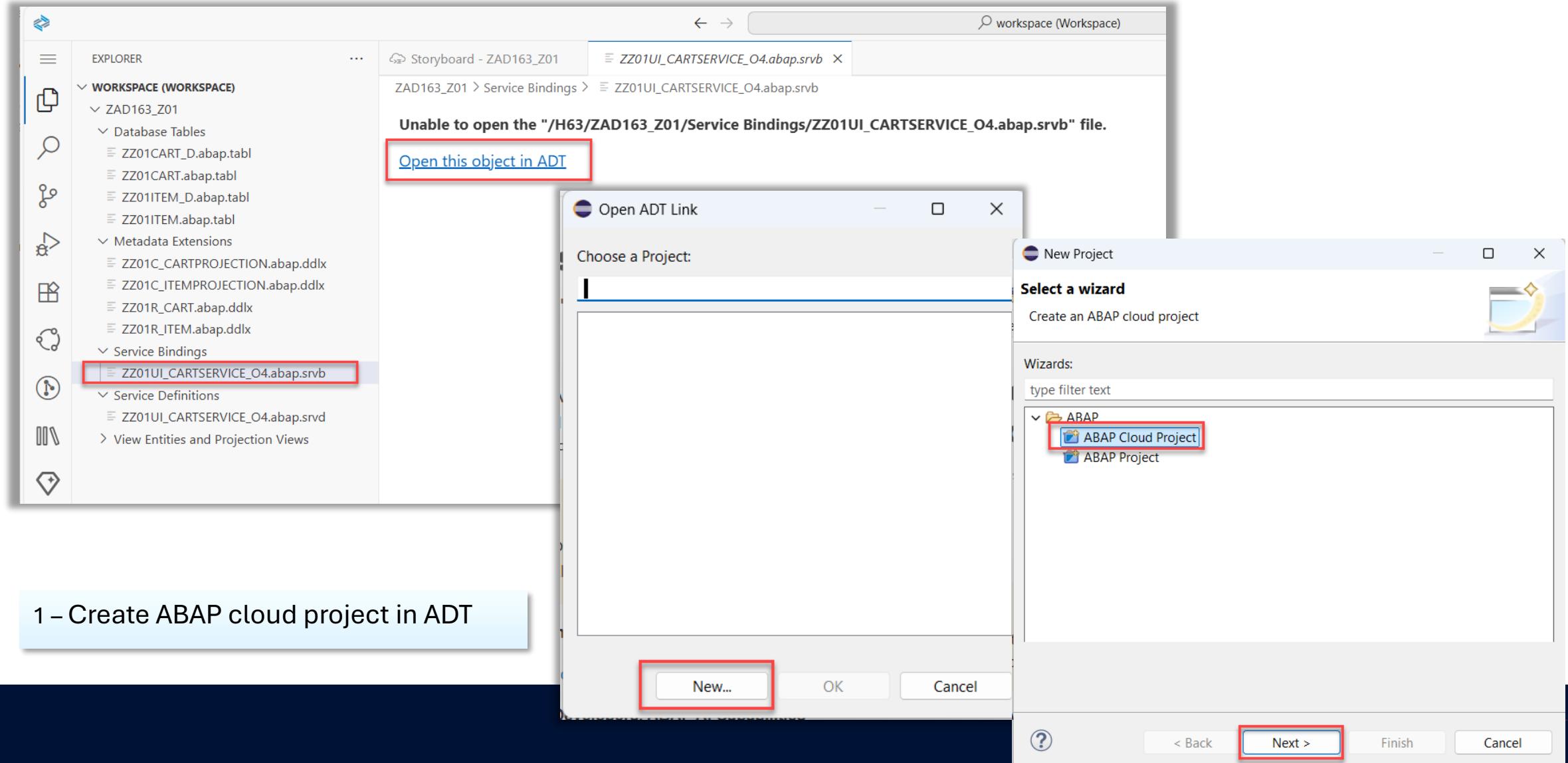


Exercise 3: Implement transactional behavior in ADT, create validations and determinations

asUG Tech Connect

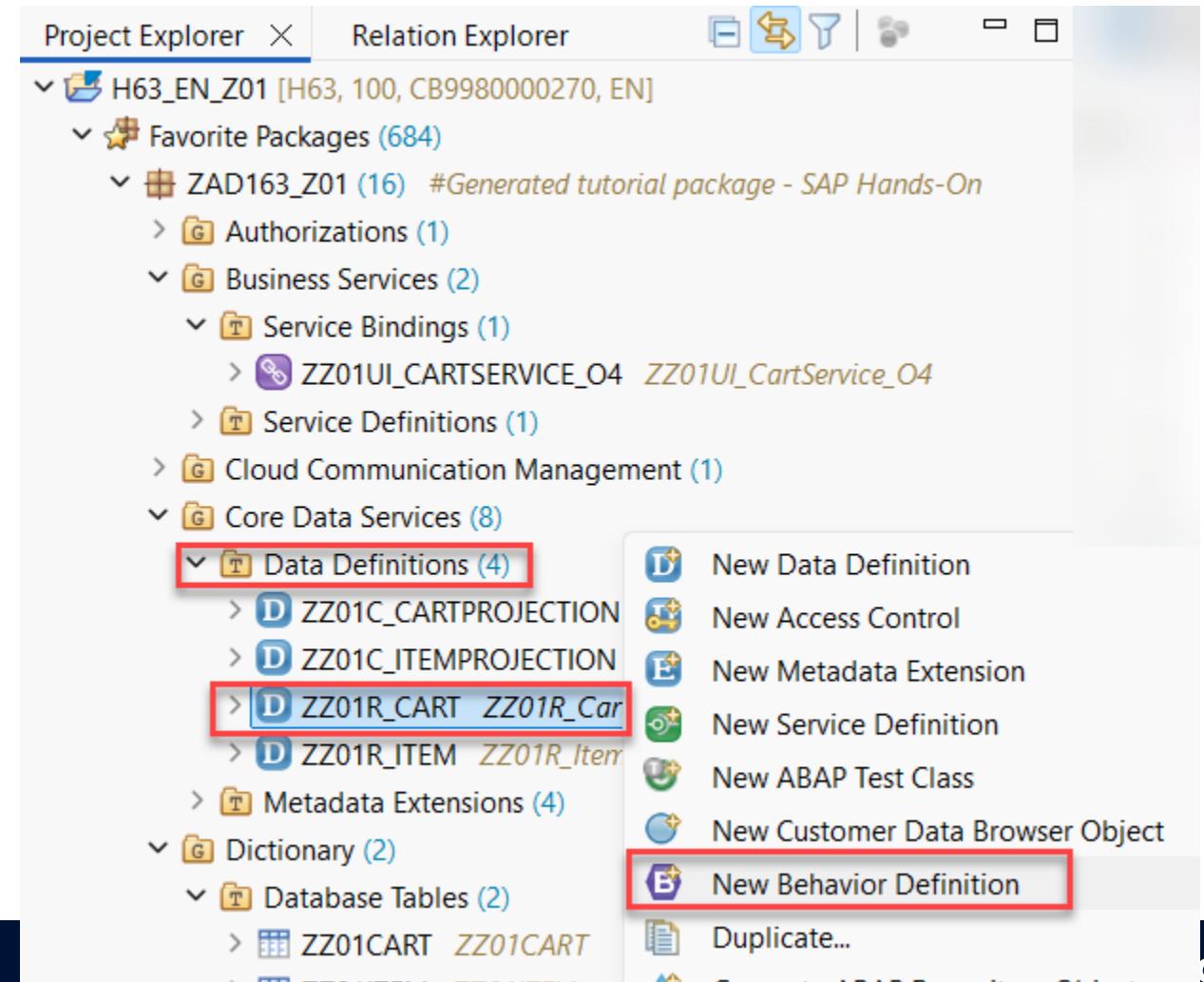
In partnership with **SAP TechEd**

Exercise 3: Create ABAP cloud project in ADT



Exercise 3: Add transactional behavior

```
1 @EndUserText.label : 'ZZ01CART'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zz01cart {
7
8     key client          : abap.clnt not null;
9     key order_uuid      : abap.raw(16) not null;
10    currency           : abap.cuky;
11    notes               : abap.char(100);
12    order_id            : abap.numc(8);
13    request_delivery_date : abap.datn;
14    @Semantics.amount.currencyCode : 'zz01cart.currency'
15    total_price         : abap.curr(11,2);
16    local_created_by    : abp_creation_user;
17    local_created_at    : abp_creation_tstmp;
18    local_last_changed_by : abp_locinst_lastchange_user;
19    local_last_changed_at : abp_locinst_lastchange_tstmp;
20    last_changed_at     : abp_lastchange_tstmp;
21
22 }
```



Exercise 3: Implement determinations and validations using Joule

The screenshot shows a code editor in SAP Studio with a red box highlighting the validation logic. Below the editor is a navigation bar with icons for file operations like New, Open, Save, etc. A large red box highlights the validation logic in the code.

```
35
36 validation validateRequestDeliveryDate on save { create; field RequestDeliveryDate; }
37
38 draft action Activate optimized;
39 draft action Discard;
40 draft action Edit;
41 draft action Resume;
42 //draft determine action Prepare;
43 draft determine action Prepare
{
44
45   validation validateRequestDeliveryDate;
}
```

Below the code editor, there is a navigation bar with icons for file operations like New, Open, Save, etc. A large red box highlights the validation logic in the code.

```
64
65 association _Items { create; with draft; }
66
67 }
```

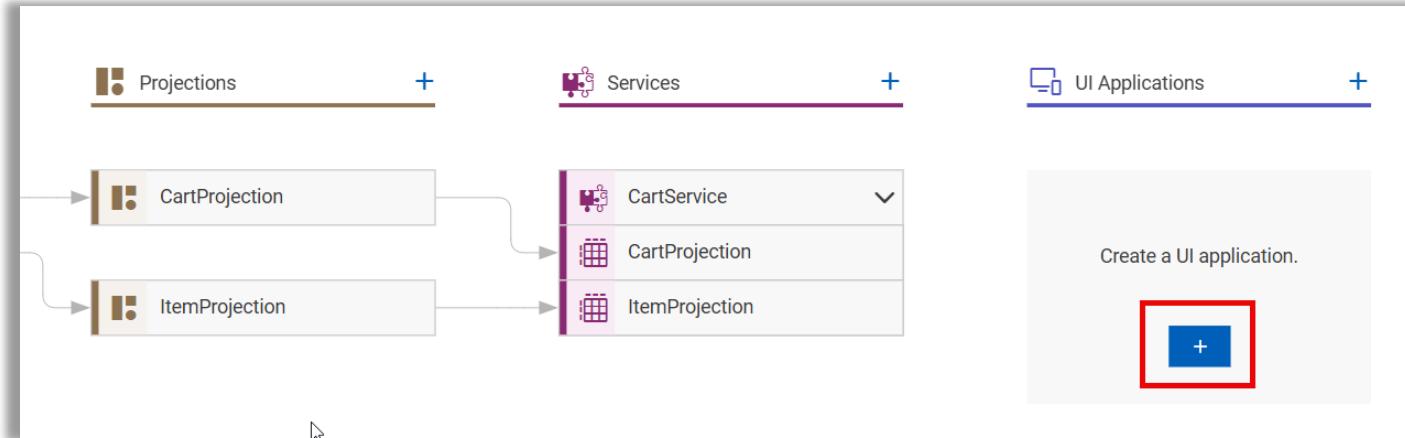
A modal dialog titled "Predict RAP Business Logic" is open, containing the text "METHOD validateRequestDeliveryDate." and "ENDMETHOD.".

Exercise 4: Create, preview and adapt a SAP Fiori elements application

asUG Tech Connect

In partnership with **SAP TechEd**

Exercise 4: Create, preview and adapt a SAP Fiori elements application



SAP Fiori generator

Template Selection
Template: List Report Page

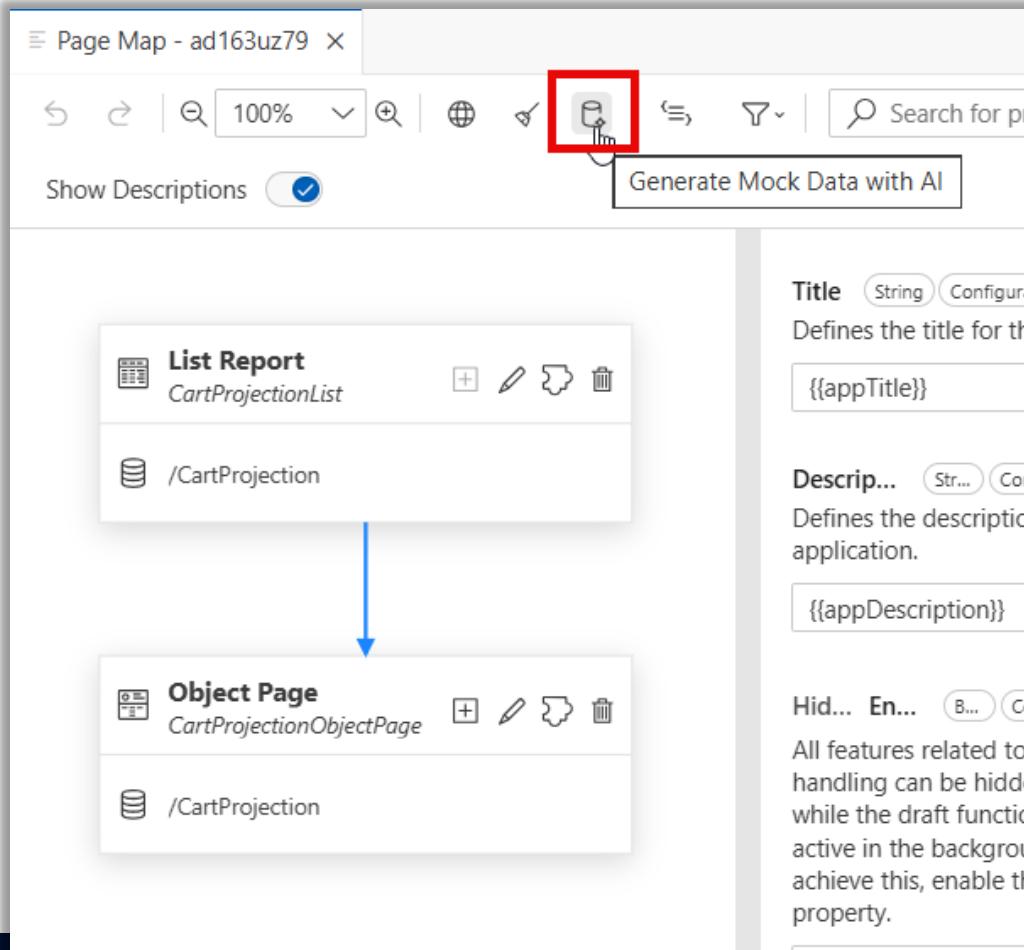
Data Source and Service Selection
Data Source: Connect to a System
System: H63 (BTP)
Service: ZZ00UI_CARTSERVICE_O4 > ZZ00UI_CARTSERVICE_O4 (001) - OData V4

Entity Selection
Main Entity: CartProjection
Navigation Entity: None
Generate Annotations: Yes
Table Type: Responsive

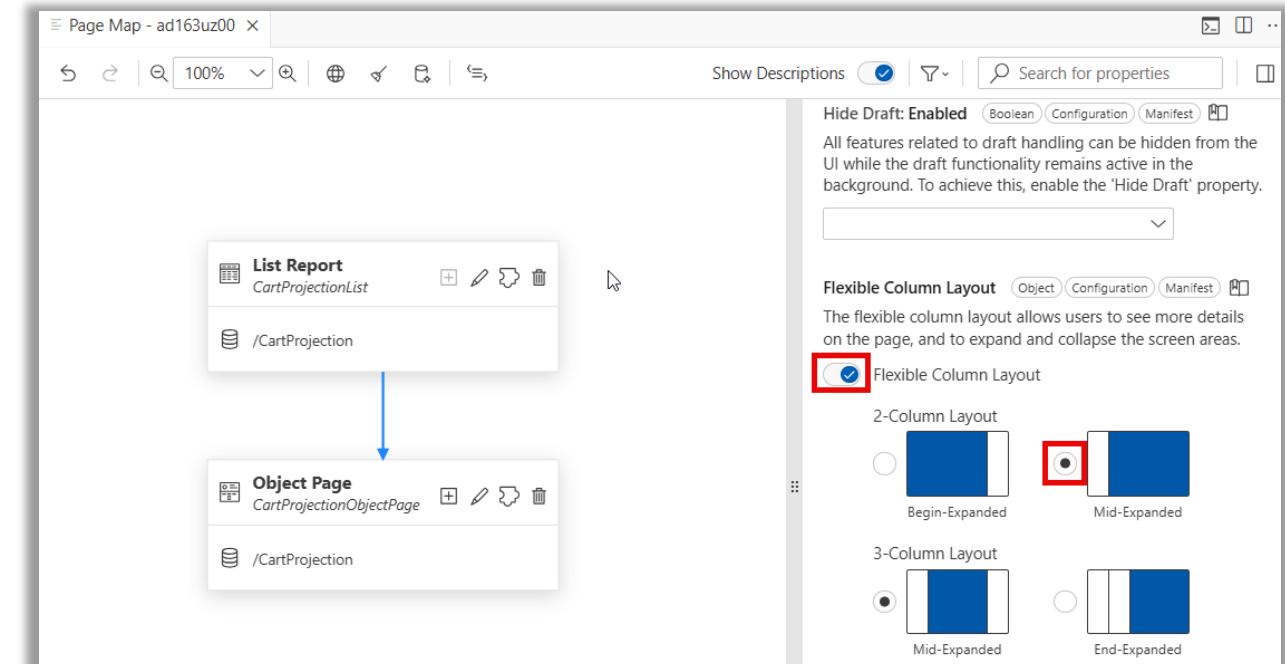
Project Attributes
Module Name: ad163uz00
Application Title: Shopping Cart App AD163UZ00
Application Namespace: Shopping Cart App AD163UZ00
Description: A Fiori application.
Project Folder Path: /home/user/projects
Minimum SAPUI5 Version: 1.136.8 (Source system version)
Enable TypeScript: Yes No
Add Deployment Configuration: Yes No
Add SAP Fiori Launchpad Configuration: Yes No
Use Virtual Endpoints for Local Preview: Yes No
Configure Advanced Options: Yes No

Buttons
< Back **Finish**

Exercise 4: Create, preview and adapt a SAP Fiori elements application



The screenshot shows the SAP Fiori Elements Page Map interface. At the top right, there is a button labeled "Generate Mock Data with AI". This button is highlighted with a red rectangle and a mouse cursor icon pointing to it. Below the header, there is a "Show Descriptions" toggle switch which is turned on (blue). The main content area displays two components: a "List Report" and an "Object Page", both associated with the path "/CartProjection". To the right of these components, there is descriptive text for each, such as "Title" and "Description". A blue arrow points from the "List Report" section down to the "Object Page" section.



The screenshot shows the SAP Fiori Elements Page Map interface. On the right side, there is a configuration panel for "Flexible Column Layout". It includes a checkbox labeled "Flexible Column Layout" which is checked (blue). Below this, there are two options: "2-Column Layout" and "3-Column Layout". Under "2-Column Layout", there are two radio button options: "Begin-Expanded" (unselected) and "Mid-Expanded" (selected, indicated by a red border). Under "3-Column Layout", there are two radio button options: "Mid-Expanded" (selected, indicated by a red border) and "End-Expanded" (unselected). A blue arrow points from the "Object Page" section down to the "Flexible Column Layout" configuration panel.

Let's start with the asUG Tech Connect Hands-on Lab

In partnership with **SAP TechEd**

Thank you!

asUG Tech Connect

In partnership with **SAP TechEd**