SAP TechEd
ON TOUR

DT266

Troubleshoot and Optimize ABAP Cloud Extensions in Cloud ERP

Steffen Mattes, SAP
Jürgen Reidl, SAP
Thomas Alexander Ritter, SAP
Thomas Fiedler, SAP

November 2025

Agenda

Today’s scenario – Overview

Today’s scenario – Exercises

Wrap up

Today’s scenario – Overview

Troubleshooting tools for RAP-based apps

ABAP CROSS TRACE

Designed for RAP-related ABAP functionalities

Provides insight into the RAP runtime framework, can trace data (like a payload trace in SAP Gateway).

FEED READER

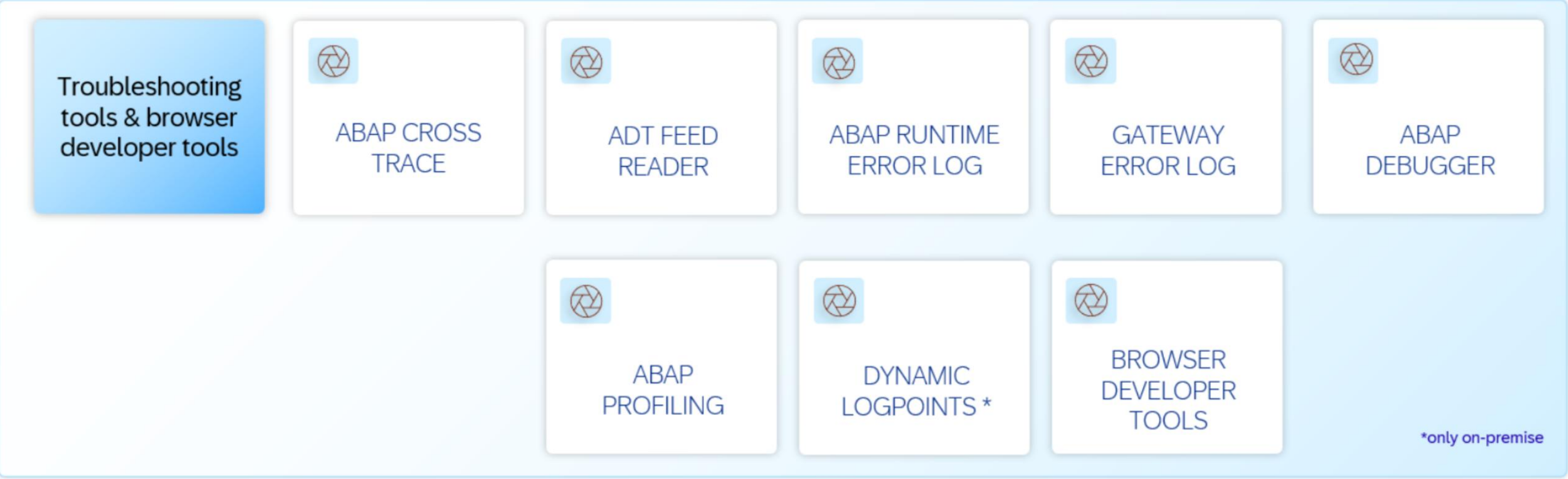
Used to read and display feeds from ABAP repository or other native feeds

Includes ABAP short dumps (ABAP Runtime Error Log) , system messages, gateway errors (Gateway Error Log, …).

ABAP PROFILING (ABAP TRACE)

Designed to trace any executable ABAP code for performance analysis.

Link: https://pages.community.sap.com/topics/abap-testing-analysis/troubleshooting



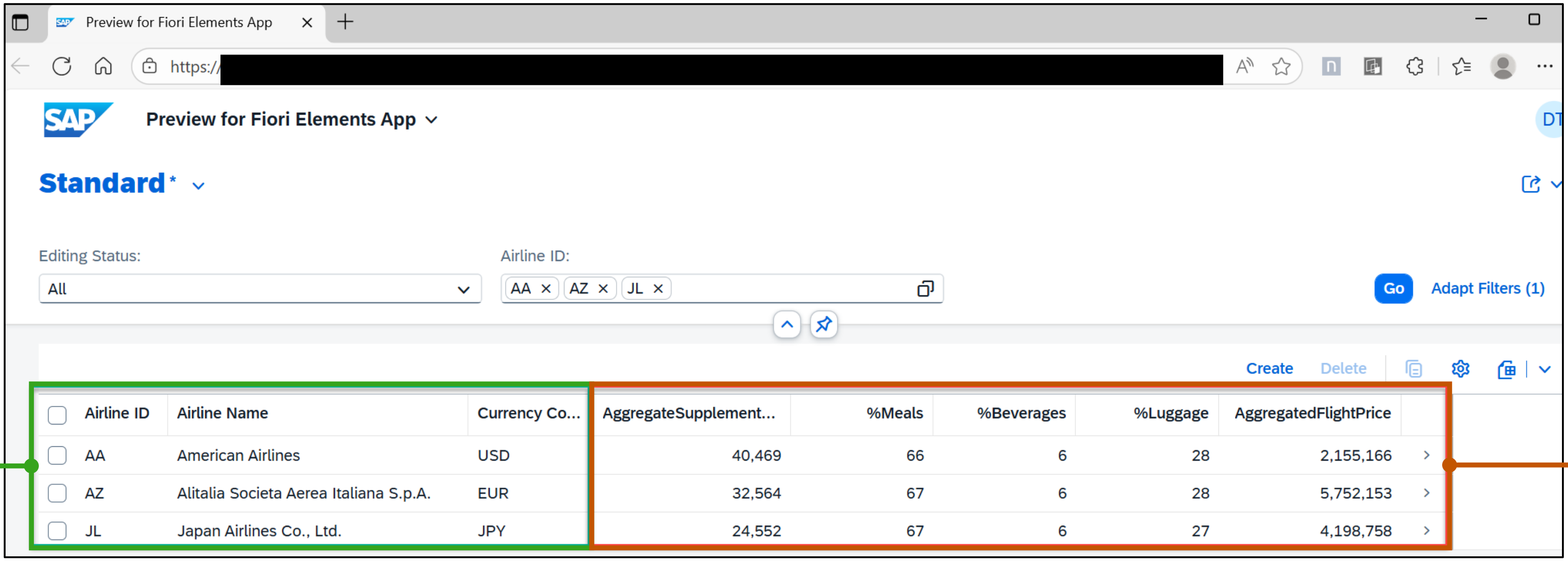
*only on-premise

Today’s scenario: Enhanced carrier information app

We have created a custom extension of a RAP App to calculate …

… the total of all corresponding flight prices & the total price of all corresponding supplements (like meal, beverage, luggage)

… the contribution in percentage (%Meals, %Beverages, %Luggage) of the supplement categories to the total price of the supplements


Preview for Fiori Elements App
SAP
Preview for Fiori Elements App
Standard*
Go
Adapt Filters (1)
Create
Delete

https://[redacted]

Editing Status: All

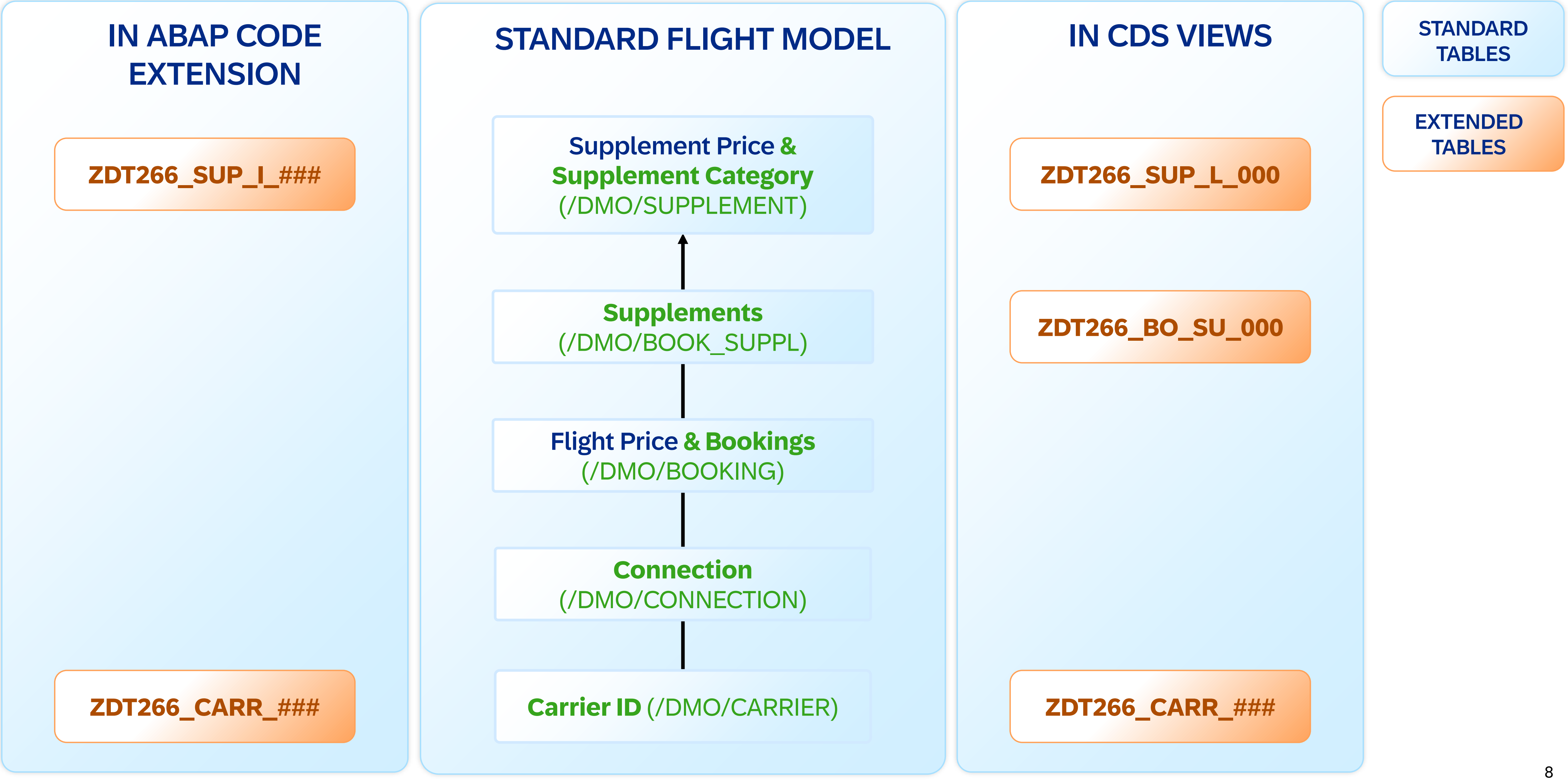
Airline ID: AA × AZ × JL ×

Airline ID	Airline Name	Currency Co...	AggregateSupplement...	%Meals	%Beverages	%Luggage	AggregatedFlightPrice
AA	American Airlines	USD	40,469	66	6	28	2,155,166
AZ	Alitalia Societa Aerea Italiana S.p.A.	EUR	32,564	67	6	28	5,752,153
JL	Japan Airlines Co., Ltd.	JPY	24,552	67	6	27	4,198,758

GREEN
RAP app directly generated from a database table

RED
Calculated fields in custom extension

Today’s scenario: Enhanced carrier information app (tables used)



IN ABAP CODE EXTENSION

ZDT266_SUP_I_###

ZDT266_CARR_###

STANDARD FLIGHT MODEL

Supplement Price & Supplement Category (/DMO/SUPPLEMENT)

Supplements (/DMO/BOOK_SUPPL)

Flight Price & Bookings (/DMO/BOOKING)

Connection (/DMO/CONNECTION)

Carrier ID (/DMO/CARRIER)

IN CDS VIEWS

ZDT266_SUP_L_000

ZDT266_BO_SU_000

ZDT266_CARR_###

STANDARD TABLES

EXTENDED TABLES

Today’s scenario – Exercises

DT266 – Troubleshoot and optimize ABAP cloud extensions in cloud ERP

DT266 HANDS-ON EXERCISES

FUNCTIONAL ANALYSIS

- Getting started
- 1 - Usage of debugger in ADT and the feed reader for runtime errors
- 2 - Usage of the memory inspector
- 3 - Usage of the ABAP cross trace

PERFORMANCE ANALYSIS

- 4 - Performance analysis and improvement using ABAP trace and table comparison tool
- 4.1 - Creation and analysis of an ABAP trace
- 4.2 - Use table buffering to improve the performance
- 4.3 - Use secondary index & key to improve the performance

Optional exercises

- 4.4 - Usage of the table comparison tool
- 4.5 - Performance of nested loops
- 5 - SQL trace analysis in SAP HANA SQL analyzer

PREREQUISITES

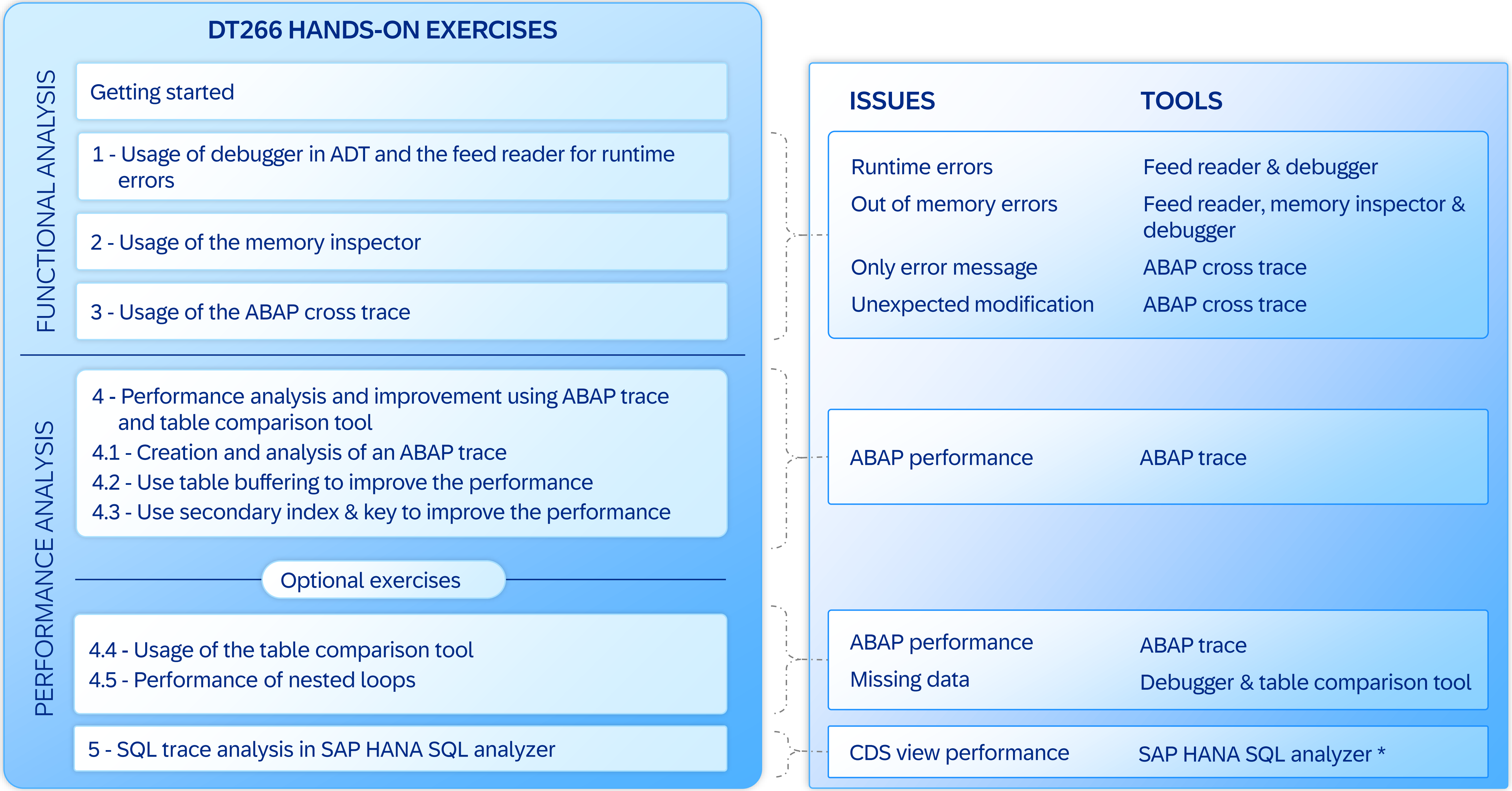
✅ Set up your ABAP Development Tools for Eclipse

✅ Access to an appropriate ABAP system: (here: APB, see handout)

ACCESS THE EXERCISES

https://github.com/SAP-samples/teched2025-DT266 (Link)

DT266 – Troubleshoot and optimize ABAP cloud extensions in cloud ERP



DT266 HANDS-ON EXERCISES

FUNCTIONAL ANALYSIS

- Getting started
- 1 - Usage of debugger in ADT and the feed reader for runtime errors
- 2 - Usage of the memory inspector
- 3 - Usage of the ABAP cross trace

PERFORMANCE ANALYSIS

- 4 - Performance analysis and improvement using ABAP trace and table comparison tool
- 4.1 - Creation and analysis of an ABAP trace
- 4.2 - Use table buffering to improve the performance
- 4.3 - Use secondary index & key to improve the performance

Optional exercises

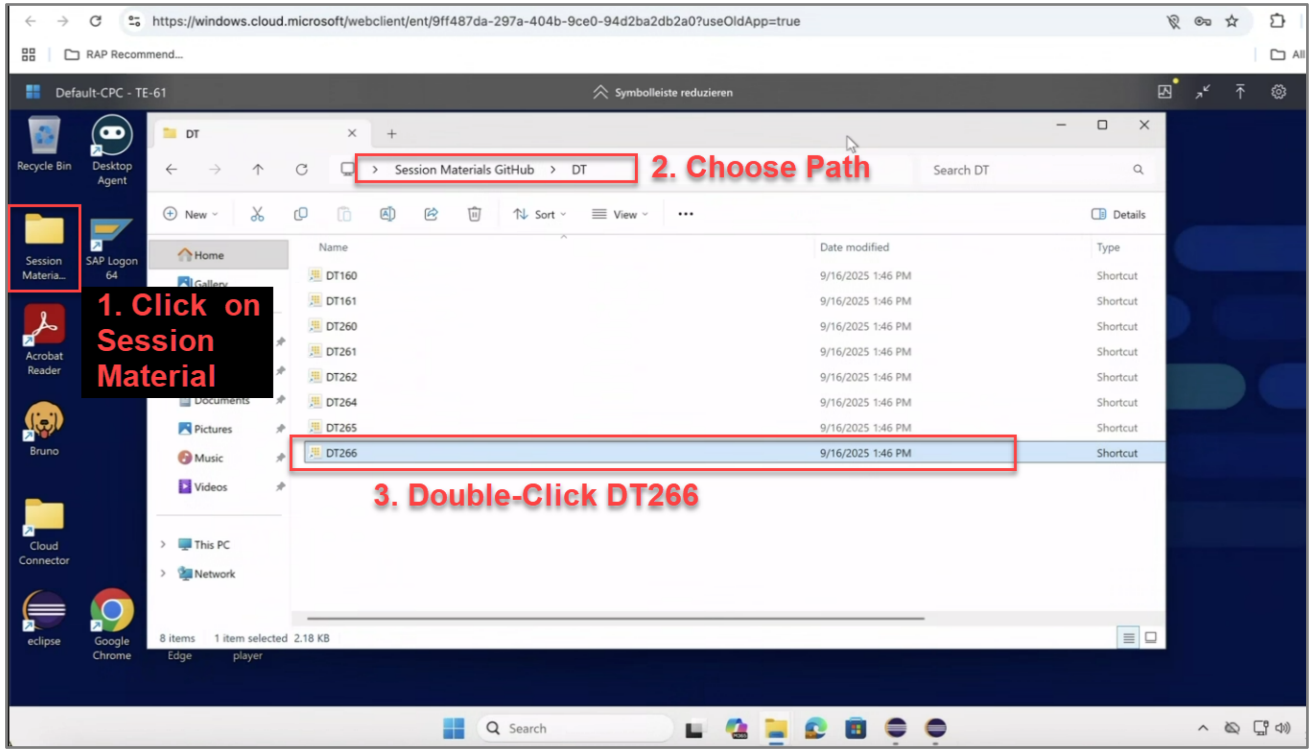
- 4.4 - Usage of the table comparison tool
- 4.5 - Performance of nested loops
- 5 - SQL trace analysis in SAP HANA SQL analyzer

ISSUES	TOOLS
Runtime errors	Feed reader & debugger
Out of memory errors	Feed reader, memory inspector & debugger
Only error message	ABAP cross trace
Unexpected modification	ABAP cross trace
ABAP performance	ABAP trace
ABAP performance	ABAP trace
Missing data	Debugger & table comparison tool
CDS view performance	SAP HANA SQL analyzer *

* In Visual Studio Code

The DT266 hands-on exercises – Working on GitHub (1)

Follow the step-by-step instructions provided on GitHub

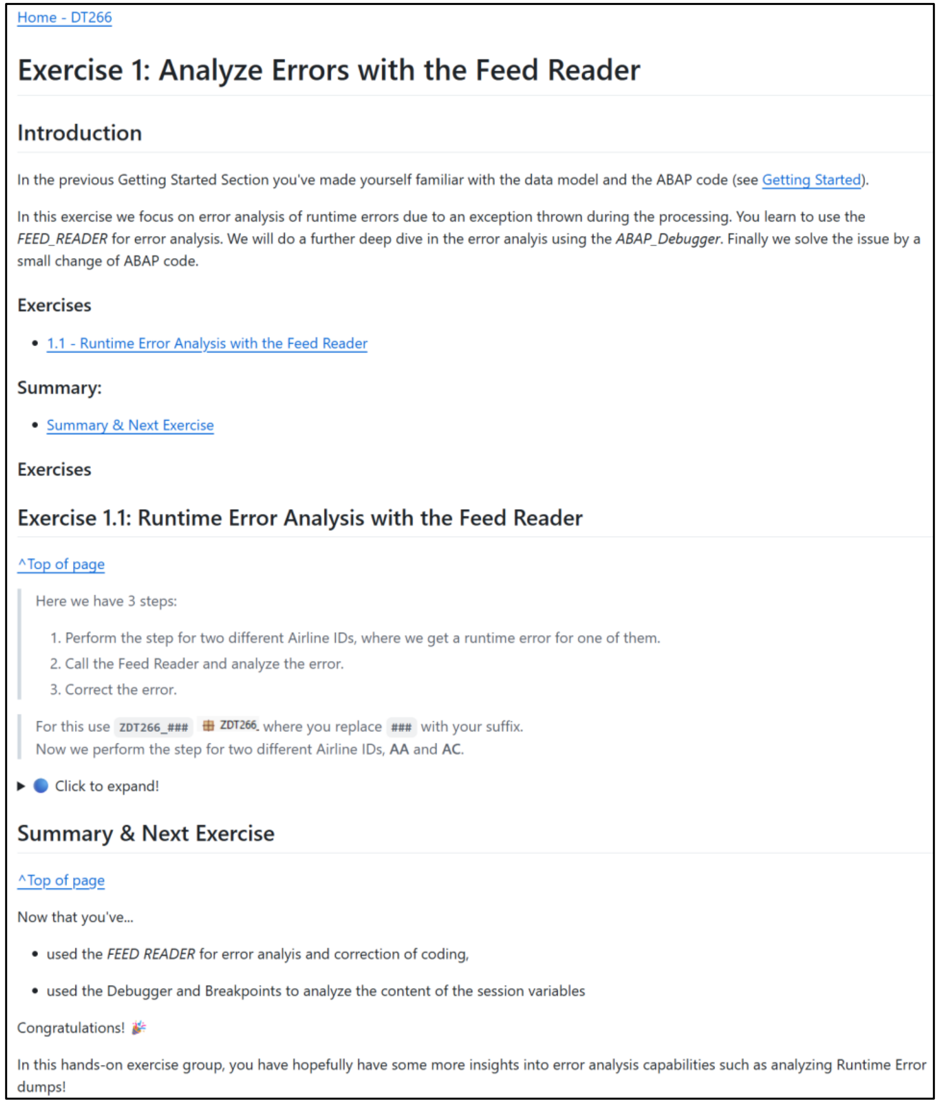


1. Click on Session Material

2. Choose Path

3. Double-Click DT266

Name	Date modified	Type
DT160	9/16/2025 1:46 PM	Shortcut
DT161	9/16/2025 1:46 PM	Shortcut
DT260	9/16/2025 1:46 PM	Shortcut
DT261	9/16/2025 1:46 PM	Shortcut
DT262	9/16/2025 1:46 PM	Shortcut
DT264	9/16/2025 1:46 PM	Shortcut
DT265	9/16/2025 1:46 PM	Shortcut
DT266	9/16/2025 1:46 PM	Shortcut



Home - DT266

Exercise 1: Analyze Errors with the Feed Reader

Introduction

In the previous Getting Started Section you've made yourself familiar with the data model and the ABAP code (see Getting Started).

In this exercise we focus on error analysis of runtime errors due to an exception thrown during the processing. You learn to use the *FEED_READER* for error analysis. We will do a further deep dive in the error analyis using the *ABAP_Debugger*. Finally we solve the issue by a small change of ABAP code.

Exercises

- 1.1 - Runtime Error Analysis with the Feed Reader

Summary:

- Summary & Next Exercise

Exercises

Exercise 1.1: Runtime Error Analysis with the Feed Reader

^Top of page

> Here we have 3 steps:
>
> 1. Perform the step for two different Airline IDs, where we get a runtime error for one of them.
> 2. Call the Feed Reader and analyze the error.
> 3. Correct the error.

> For this use `ZDT266_###` ⊞ ZDT266, where you replace `###` with your suffix.
> Now we perform the step for two different Airline IDs, **AA** and **AC**.

▸ 🔵 Click to expand!

Summary & Next Exercise

^Top of page

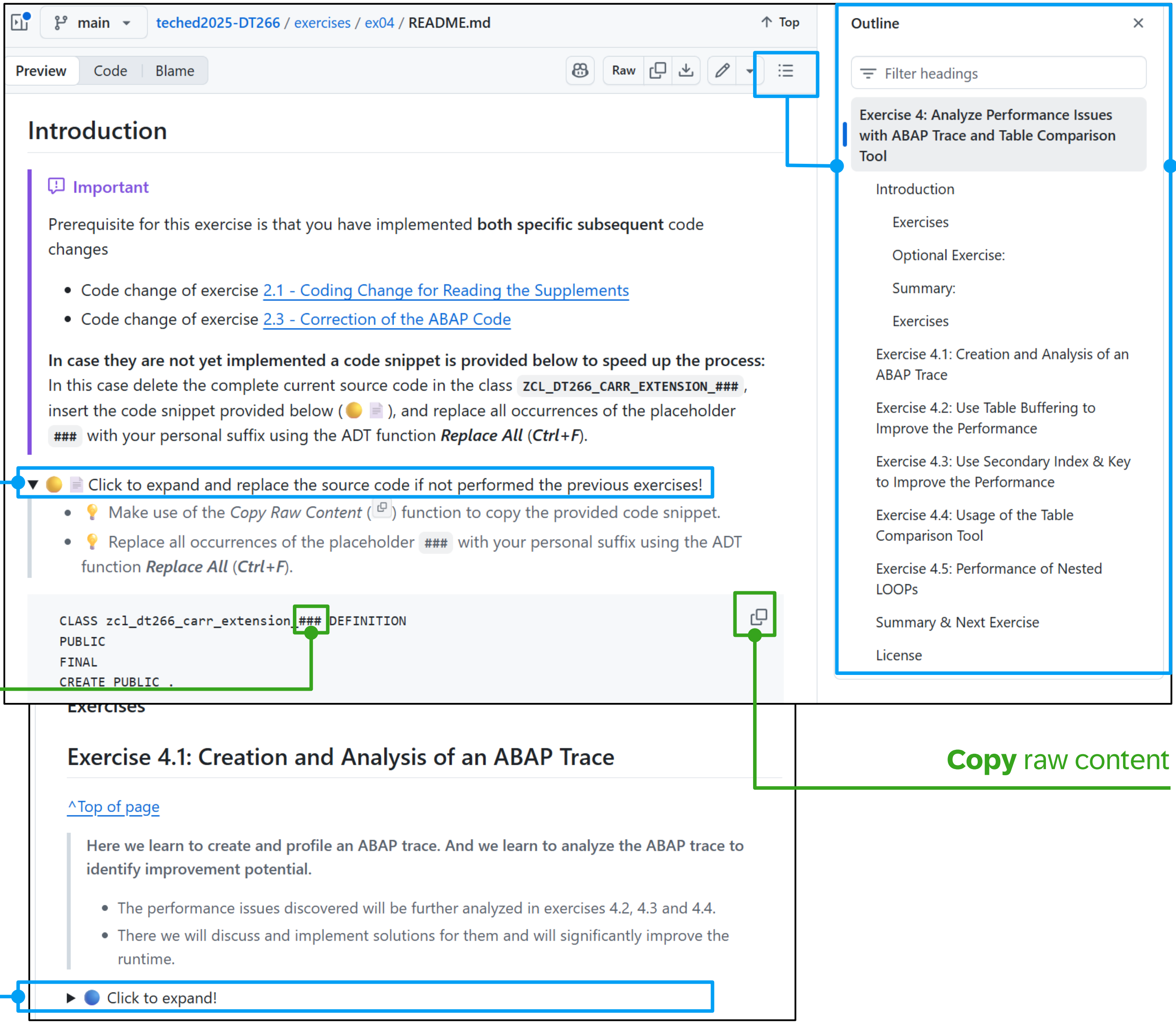
Now that you've...

- used the *FEED READER* for error analyis and correction of coding,
- used the Debugger and Breakpoints to analyze the content of the session variables

Congratulations! 🎉

In this hands-on exercise group, you have hopefully have some more insights into error analysis capabilities such as analyzing Runtime Error dumps!

The DT266 hands-on exercises – Working on GitHub (2)



Exercise 4: Analyze Performance Issues with ABAP Trace and Table Comparison Tool
- Introduction
  - Exercises
  - Optional Exercise:
  - Summary:
  - Exercises
- Exercise 4.1: Creation and Analysis of an ABAP Trace
- Exercise 4.2: Use Table Buffering to Improve the Performance
- Exercise 4.3: Use Secondary Index & Key to Improve the Performance
- Exercise 4.4: Usage of the Table Comparison Tool
- Exercise 4.5: Performance of Nested LOOPs
- Summary & Next Exercise
- License

Introduction

> Important
>
> Prerequisite for this exercise is that you have implemented **both specific subsequent** code changes
>
> - Code change of exercise 2.1 - Coding Change for Reading the Supplements
> - Code change of exercise 2.3 - Correction of the ABAP Code
>
> **In case they are not yet implemented a code snippet is provided below to speed up the process:** In this case delete the complete current source code in the class `ZCL_DT266_CARR_EXTENSION_###`, insert the code snippet provided below (🟡 📄), and replace all occurrences of the placeholder `###` with your personal suffix using the ADT function ***Replace All*** (***Ctrl+F***).

▼ 🟡 📄 Click to expand and replace the source code if not performed the previous exercises!

- 💡 Make use of the *Copy Raw Content* (⧉) function to copy the provided code snippet.
- 💡 Replace all occurrences of the placeholder `###` with your personal suffix using the ADT function ***Replace All*** (***Ctrl+F***).

```
CLASS zcl_dt266_carr_extension_### DEFINITION
PUBLIC
FINAL
CREATE PUBLIC .
```

Exercises

Exercise 4.1: Creation and Analysis of an ABAP Trace

^Top of page

> Here we learn to create and profile an ABAP trace. And we learn to analyze the ABAP trace to identify improvement potential.
>
> - The performance issues discovered will be further analyzed in exercises 4.2, 4.3 and 4.4.
> - There we will discuss and implement solutions for them and will significantly improve the runtime.

▶ 🔵 Click to expand!

Use outline for a **quick overview**

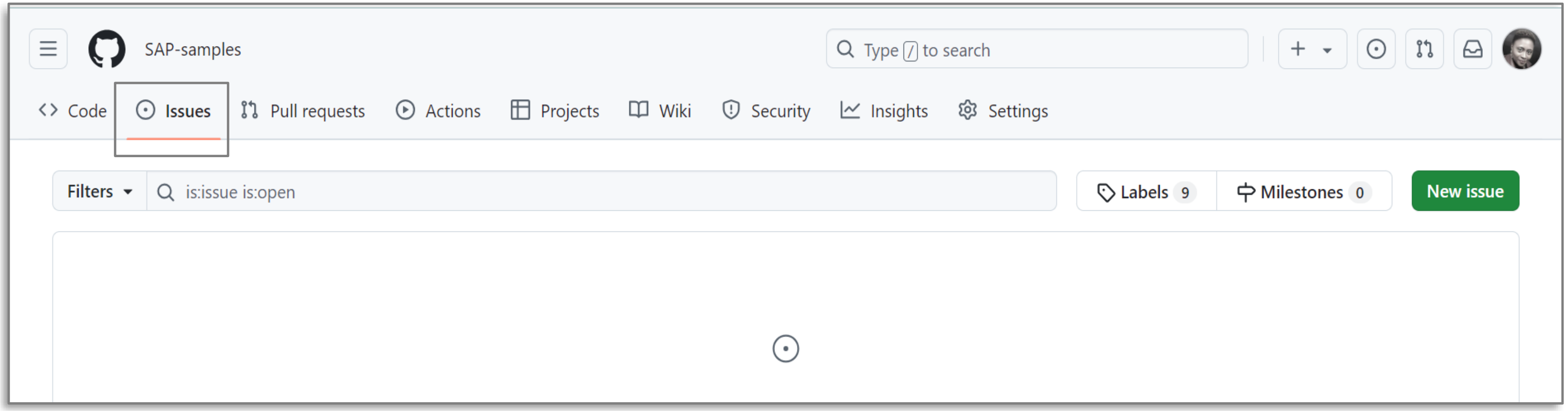
Replace all occurrences of ### with your suffix

Copy raw content

Expand to see details

The DT266 hands-on exercises – Working on GitHub (3)

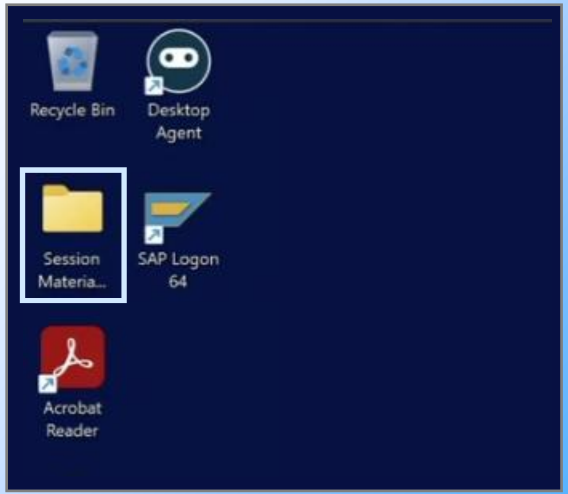
Open GitHub Issues to get help from the SAP experts



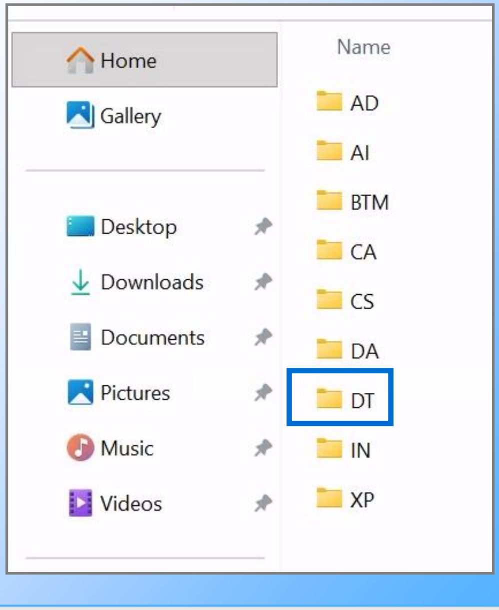
is:issue is:open

How to access the exercise material

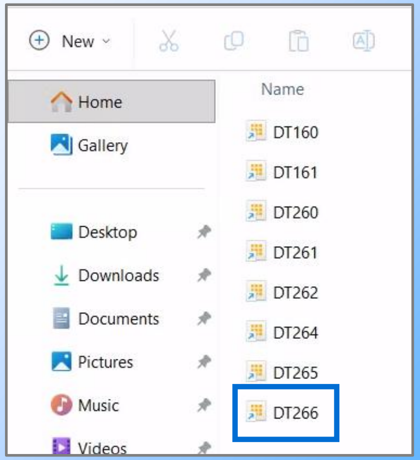
(1) Open the folder **Session Materials GitHub** on the desktop



(2) Open the folder **DT**

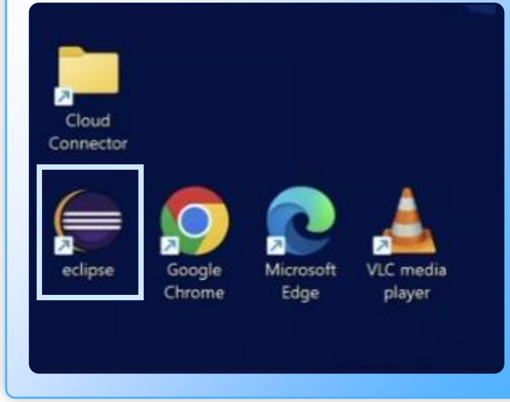


(3) Double-click the shortcut Open the folder **DT266** to navigate to the hands-on exercise

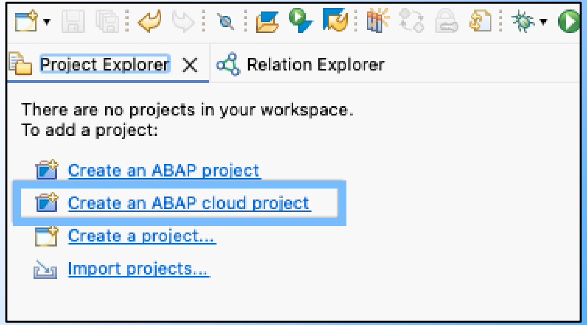


How to access your package in Eclipse

(1) Double-click the eclipse icon on the desktop to start it



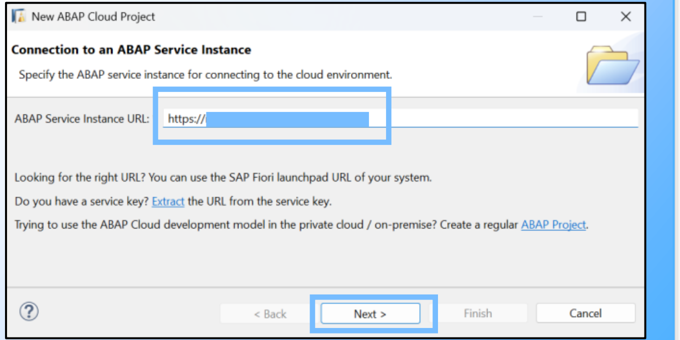
(2) Create a new ABAP Cloud Project



There are no projects in your workspace.
To add a project:

- Create an ABAP project
- Create an ABAP cloud project
- Create a project...
- Import projects...

(3) Enter the URL of system APB:
https://[redacted]
and click next



New ABAP Cloud Project

Connection to an ABAP Service Instance

Specify the ABAP service instance for connecting to the cloud environment.

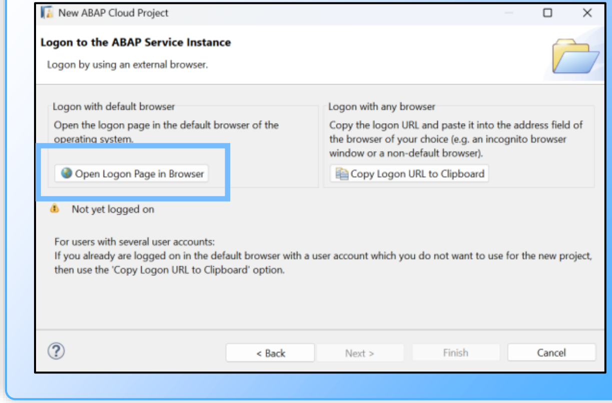
ABAP Service Instance URL: https://[redacted]

Looking for the right URL? You can use the SAP Fiori launchpad URL of your system.

Do you have a service key? Extract the URL from the service key.

Trying to use the ABAP Cloud development model in the private cloud / on-premise? Create a regular ABAP Project.

(4) Click on open logon page in browser



New ABAP Cloud Project

Logon to the ABAP Service Instance

Logon by using an external browser.

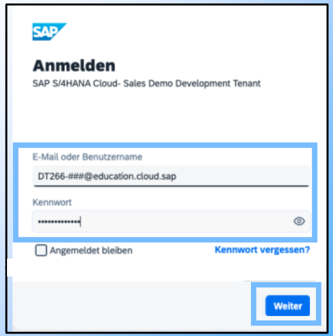
Logon with default browser
Open the logon page in the default browser of the operating system.

Logon with any browser
Copy the logon URL and paste it into the address field of the browser of your choice (e.g. an incognito browser window or a non-default browser).

Not yet logged on

For users with several user accounts:
If you already are logged on in the default browser with a user account which you do not want to use for the new project, then use the 'Copy Logon URL to Clipboard' option.

(5) Enter user name and password



Anmelden

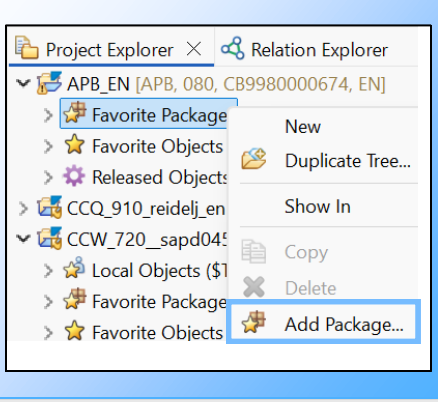
SAP S/4HANA Cloud- Sales Demo Development Tenant

E-Mail oder Benutzername: DT266-###@education.cloud.sap

Kennwort: ••••••••••••••

□ Angemeldet bleiben

(6) Add your package to favorite packages

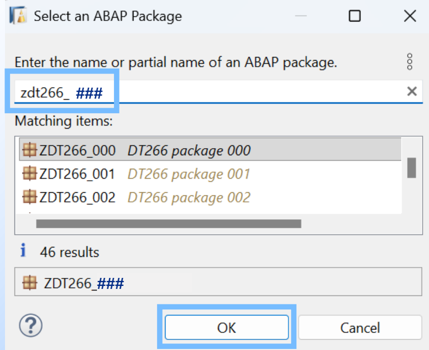


APB_EN [APB, 080, CB9980000674, EN]
- Favorite Package
- Favorite Objects
- Released Objects

CCQ_910_reidelj_en

CCW_720_sapd045
- Local Objects ($T
- Favorite Package
- Favorite Objects

(7) Enter with your group number ###



Select an ABAP Package

Enter the name or partial name of an ABAP package.

zdt266_ ###

Matching items:

ZDT266_000	DT266 package 000
ZDT266_001	DT266 package 001
ZDT266_002	DT266 package 002

46 results

ZDT266_###

LET'S GET STARTED!

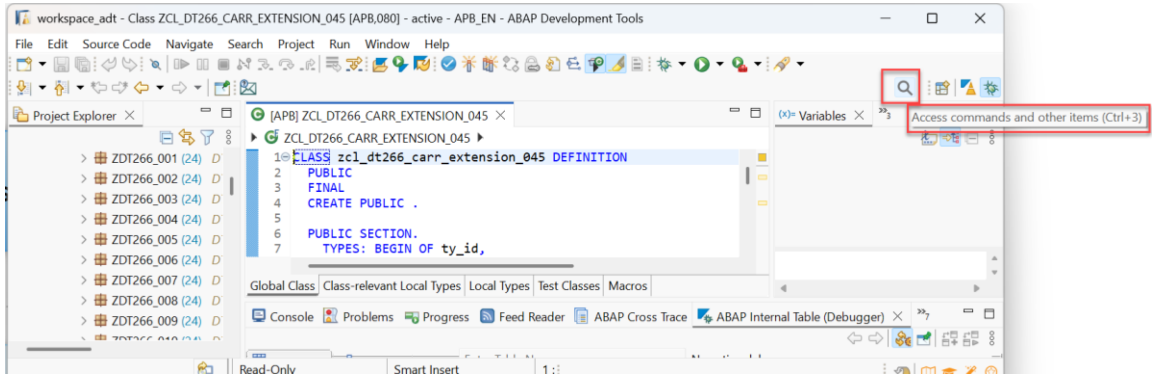
Move at your own pace!

Feel free to ask questions!

ACCESS THE EXERCISES

https://github.com/SAP-samples/teched2025-DT266 (Link)

ACCESS THE TOOLs



```
CLASS zcl_dt266_carr_extension_045 DEFINITION
  PUBLIC
  FINAL
  CREATE PUBLIC .

  PUBLIC SECTION.
    TYPES: BEGIN OF ty_id,
```

ABAP Cloud-related sessions at SAP TechEd 2025

AD808 | SAP Build: ABAP Cloud road map

ST109 | SAP Build: ABAP Cloud strategy

AD104 | Boost your ABAP development with SAP Joule for developers

AD162 | Explore Joule for developers, ABAP AI capabilities ⚙

AD201 | Explore ABAP Cloud in SAP Build – what's in for ABAP developers?

AD163 | Fusion Development with ABAP Cloud in SAP Build ⚙

AD105 | Introduction to ABAP Cloud and how to get started

AD164 | Get started with ABAP Cloud for classic ABAP developers ⚙

DT814 | Extensibility innovations road map for SAP Business Suite and cloud ERP

ST113 | Journey to unified extensibility in SAP Business Suite and cloud ERP

⚙ Hands-on workshop

DT200 | Unravel the mysteries of Clean Core extensions in Cloud ERP Private

DT260 | Modernize classic extensions to clean core in Cloud ERP Private ⚙

DT201 | Extensibility with ABAP Cloud and AI in SAP Cloud ERP

DT261 | Build on-stack extensions with ABAP Cloud in Cloud ERP Public ⚙

DT266 | Troubleshoot and Optimize ABAP Cloud Extensions in Cloud ERP ⚙

Visit our demo stations on the Show Floor:

AD900 | Application Development with ABAP Cloud, CAP, SAPUI5 in SAP Build

DT916 | Discover the extensibility options for cloud ERP

# See Blog post

SAP TechEd
ON TOUR

Steffen Mattes:
steffen.mattes@sap.com

Jürgen Reidl
juergen.reidl@sap.com

Thank you!

SAP Bring out your best.