

ABAP Development in Eclipse

Tips and Tricks Session, Part 2

Thomas Fiedler & Ünal Akkaya

SAP SE

Key Outcomes and Objectives

Part 1

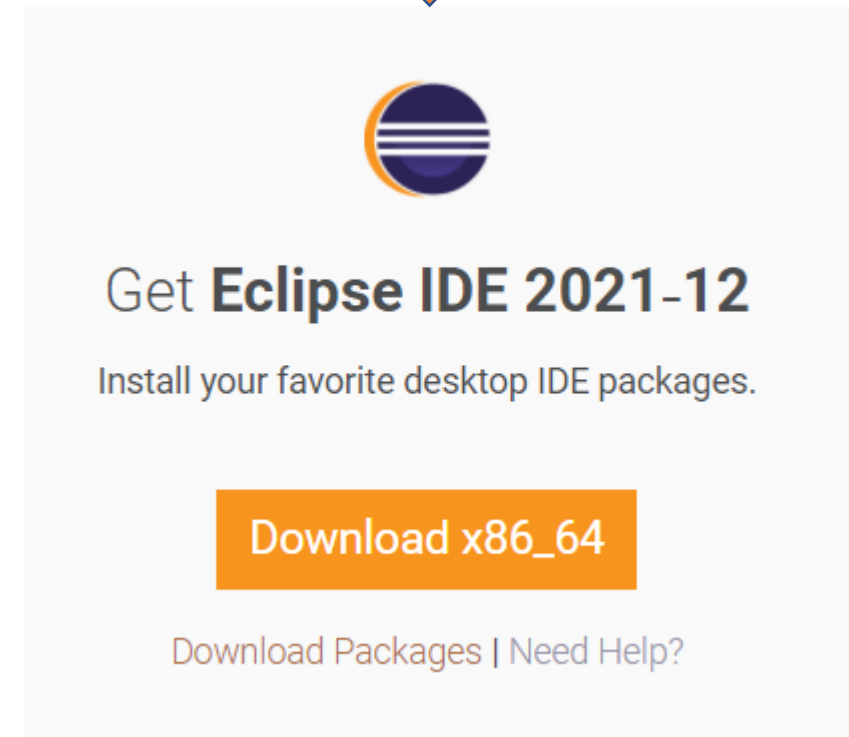
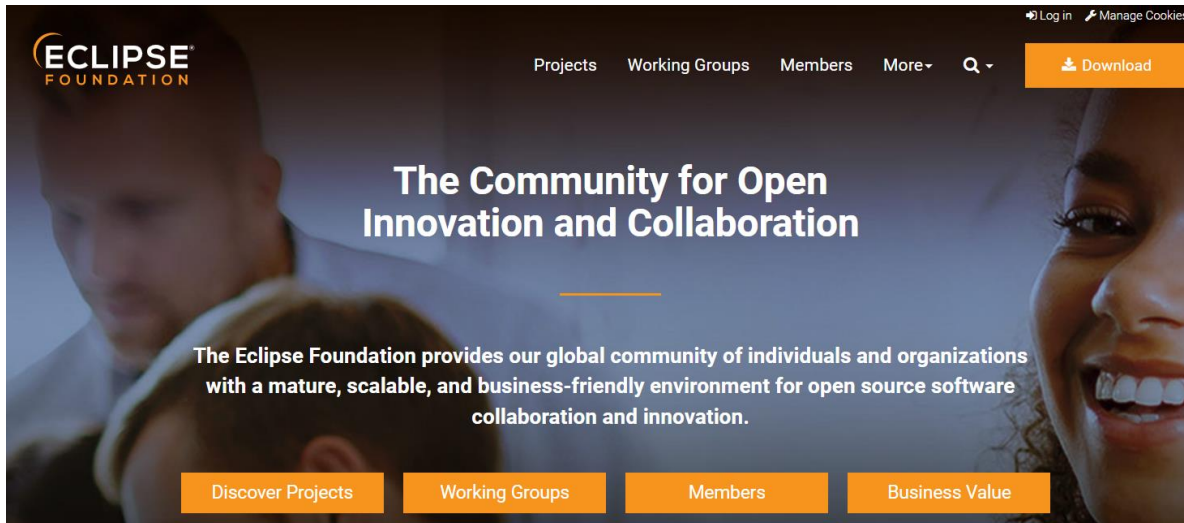
1. First Steps in ADT
2. Working with ADT
3. Basics of ADT

Agenda

- Getting Started with ABAP Developer Tools
- First Steps in ADT
 - Eclipse basics
 - Views, Editors & Perspectives
- Project Explorer
- Finding Objects
- Working with Development Objects
 - Report
 - Class
- Advanced
 - Quick Assist

Installation Eclipse

Install eclipse from the official site: <https://www.eclipse.org>



Installation ADT / Where to find?

Install ADT following the procedure on the official site: <https://tools.hana.ondemand.com/#abap>



ABAP Development Tools

This site describes how to install and update the front-end components of ABAP Development Tools (ADT) and SAP BTP ABAP Environment.

It also provides you with detailed information on how to prepare the relevant ABAP back-end system for working with ADT. In case you connect to SAP BTP

Prerequisites

Component	Prerequisite
Eclipse Platform	2021-09 (4.21, x86_64), 2021-12 (4.22, x86_64)
Operating System	<ul style="list-style-type: none">Windows 10, orApple Mac OS X 10.14 or higher (Note: AArch64 (M1) is supported, but the x86_64 version is required)
Java Runtime	JRE version 11 (64-Bit, LTS) or higher (*)
SAP GUI	<ul style="list-style-type: none">For Windows OS: SAP GUI for Windows 7.60 or higherFor Apple Mac OS: SAP GUI for Java 7.50 or higher (**) <p>SAP GUI is not required to work with SAP BTP ABAP Environment.</p>
Microsoft VC Runtime	For Windows OS: Microsoft Visual C++ 2013 (x64) for communication with the back-end system

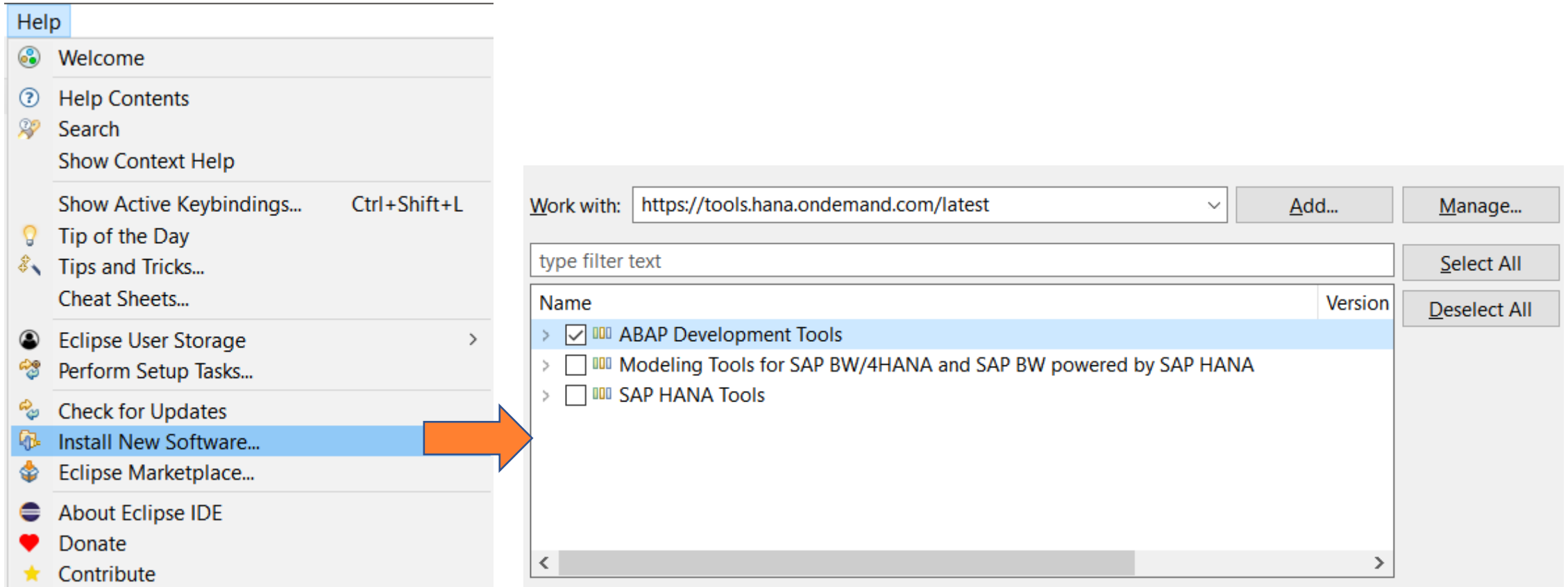
Procedure

To install the front-end component of ADT, proceed as follows:

1. Get an installation of [Eclipse 2021-12 \(x86_64\)](#) (e.g. [Eclipse IDE for Java Developers](#))
2. In Eclipse, choose in the menu bar **Help > Install New Software...**
3. Enter the URL <https://tools.hana.ondemand.com/latest>
4. Press **Enter** to display the available features.
5. Select **ABAP Development Tools** and choose **Next**.
6. On the next wizard page, you get an overview of the features to be installed. Choose **Next**.
7. Confirm the **license agreements** and choose **Finish** to start the installation.

Installation ADT / Update site

Update site: <https://tools.hana.ondemand.com/latest>

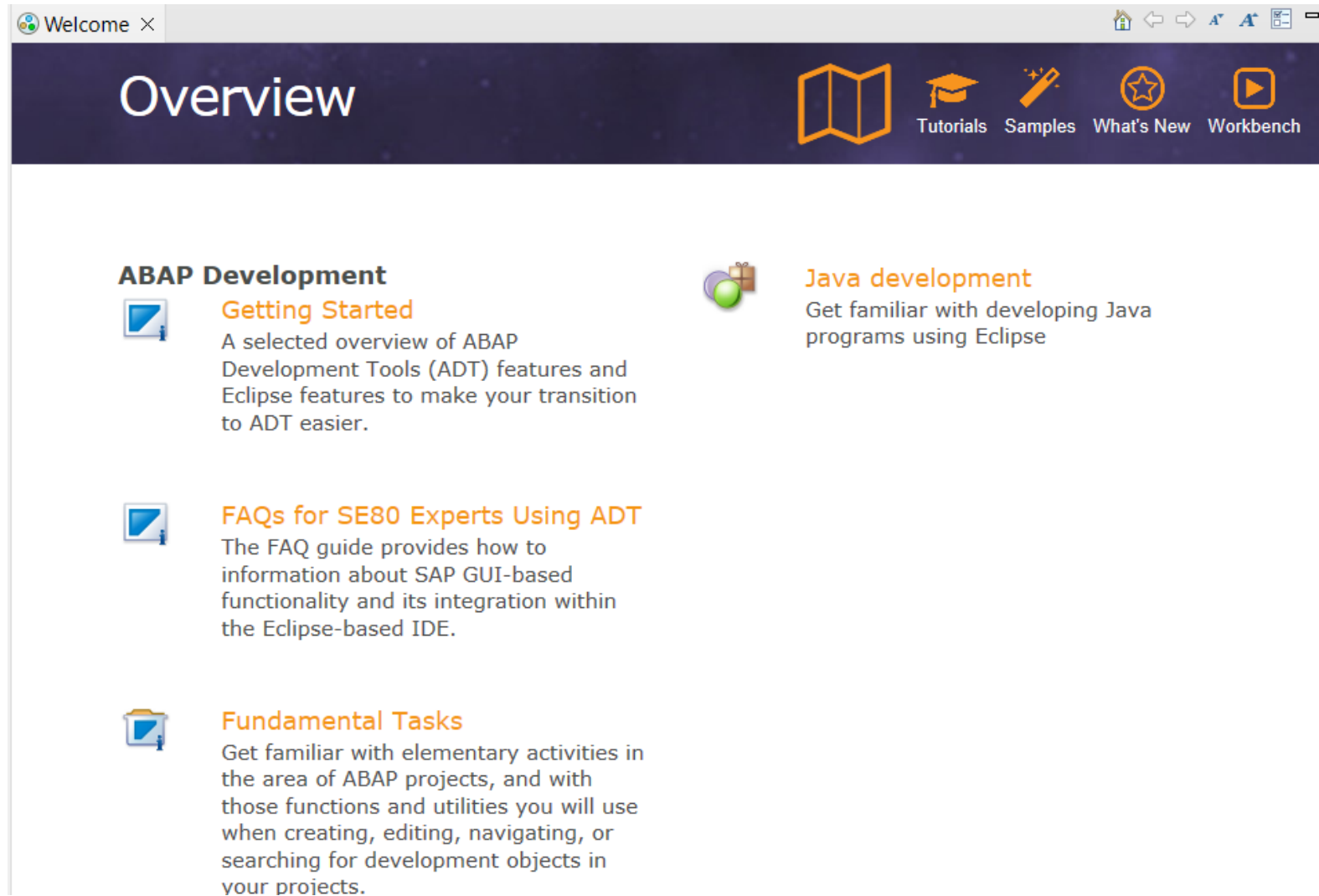


The screenshot shows the Eclipse IDE interface. On the left is the 'Help' menu, with 'Install New Software...' highlighted. An orange arrow points from this menu item to the 'Install New Software' dialog box on the right. The dialog box has a 'Work with:' field containing the URL 'https://tools.hana.ondemand.com/latest'. Below this is a search bar labeled 'type filter text'. A table lists available software:

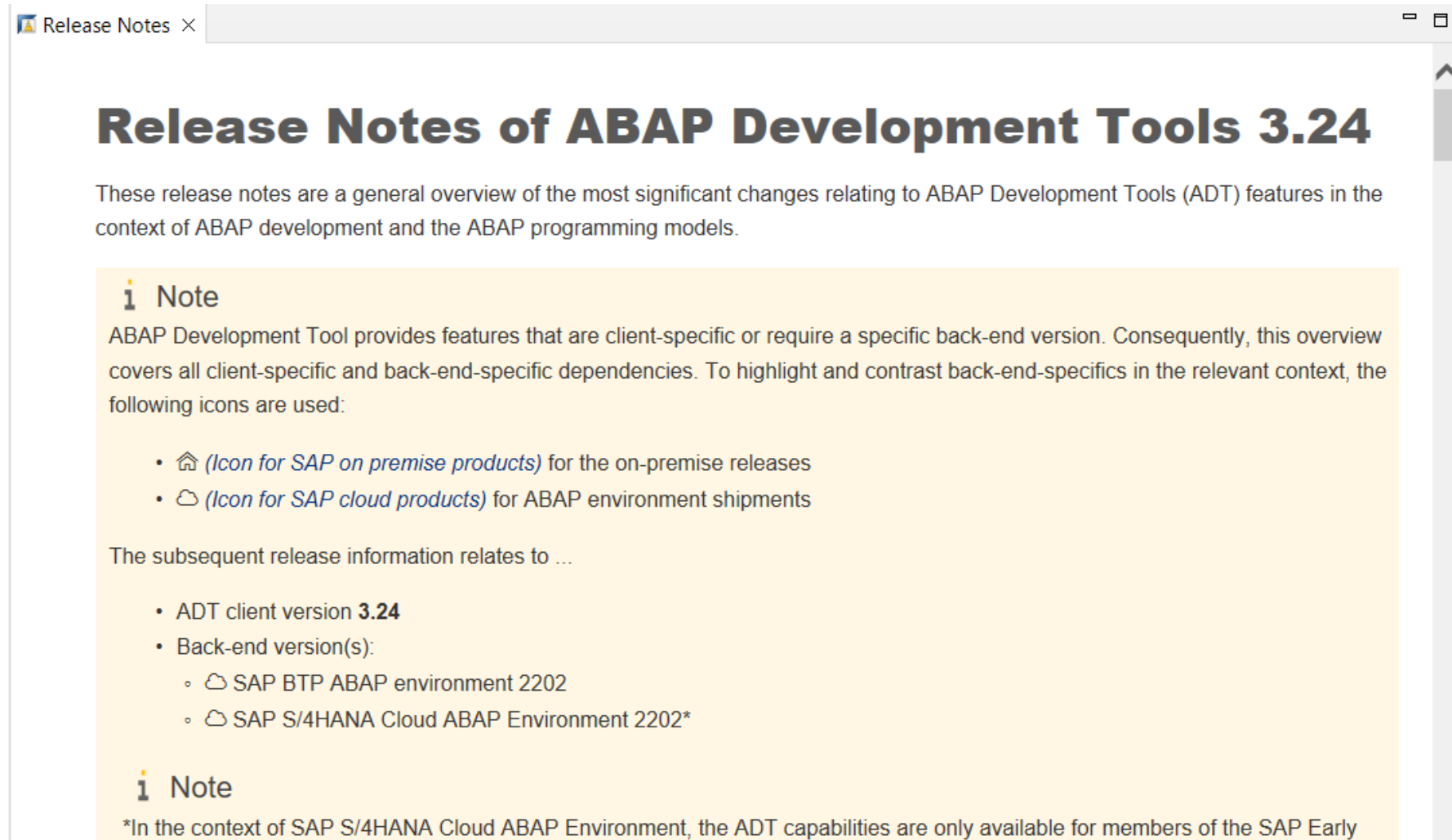
Name	Version
> <input checked="" type="checkbox"/> ABAP Development Tools	
> <input type="checkbox"/> Modeling Tools for SAP BW/4HANA and SAP BW powered by SAP HANA	
> <input type="checkbox"/> SAP HANA Tools	

Buttons on the right include 'Add...', 'Manage...', 'Select All', and 'Deselect All'.

Installation ADT / Getting Started



Installation ADT / Release Notes



Release Notes ×

Release Notes of ABAP Development Tools 3.24

These release notes are a general overview of the most significant changes relating to ABAP Development Tools (ADT) features in the context of ABAP development and the ABAP programming models.

i Note

ABAP Development Tool provides features that are client-specific or require a specific back-end version. Consequently, this overview covers all client-specific and back-end-specific dependencies. To highlight and contrast back-end-specifics in the relevant context, the following icons are used:

- 🏠 (Icon for SAP on premise products) for the on-premise releases
- ☁️ (Icon for SAP cloud products) for ABAP environment shipments

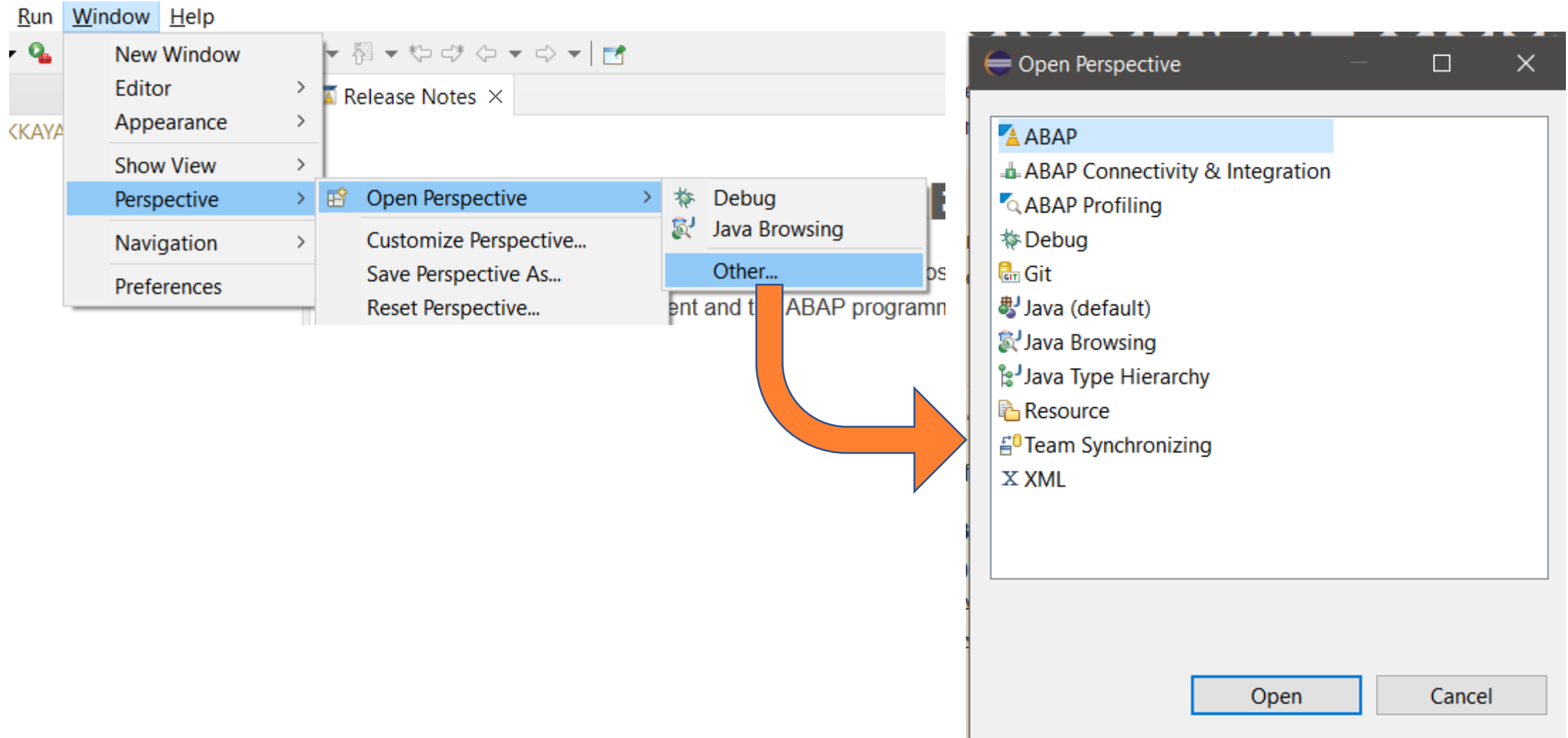
The subsequent release information relates to ...

- ADT client version **3.24**
- Back-end version(s):
 - ☁️ SAP BTP ABAP environment 2202
 - ☁️ SAP S/4HANA Cloud ABAP Environment 2202*

i Note

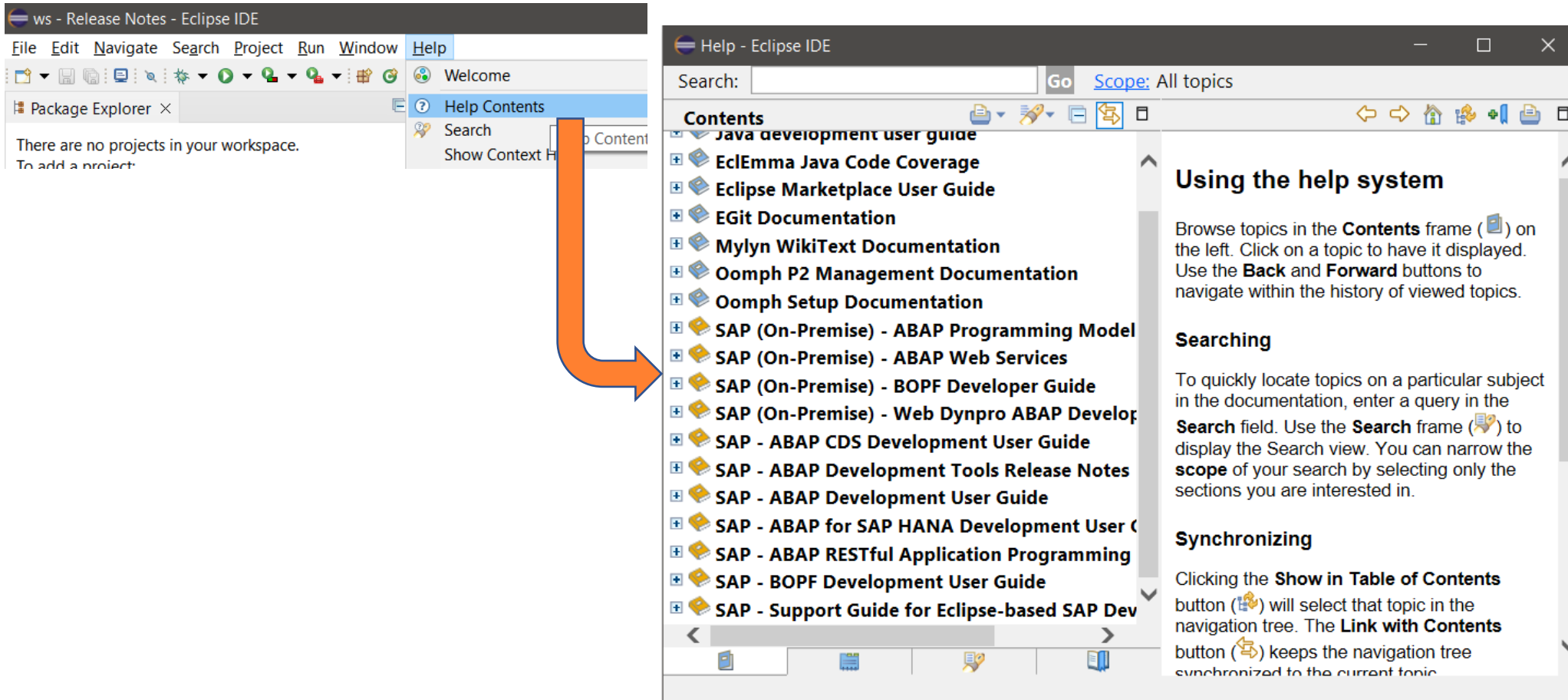
*In the context of SAP S/4HANA Cloud ABAP Environment, the ADT capabilities are only available for members of the SAP Early

ADT: Switch to ABAP Perspective



ADT: Build in Help

Find documentation in the build in ADT help.



ADT: Build in Help

Check out the *Getting Started*, *Tips and Tricks* and *FAQs for SE80 Experts Using ADT* documentation.

SAP - ABAP Development User | [SAP - ABAP Development User Guide](#)

About the ABAP Development

Getting Started

- Basic Tutorial
 - ABAP Perspective
 - ABAP Project
- Project Explorer
- Outline
- ABAP Editors and Views
 - Troubleshooting
 - Testing and Ensuring Q
 - Transport Organizer
- FAQ

Getting Started

Contents

- [Basic Tutorial](#)
- [FAQ](#)

SAP - ABAP Development User | [SAP - ABAP Development User Guide >](#)

About the ABAP Development

- Getting Started
- Concepts
- Tasks
- Reference
 - Released Object Types
 - List of Development Objec
 - Comparison of the ADT Fe
- Keyboard Shortcuts for AB.
- Syntax of ABAP Dictionary
- Logpoint Syntax
- Syntax for Breakpoint Conc
- Supported Change Function
- FAQs for SE80 Experts Usin
- Tips and Tricks (ABAP Core To**
- Views
- Editing ABAP Source Code

Tips and Tricks (ABAP Core Tools)

The following chapters provide an overview of tips and trick working within the ABAP core development scenario using the ABAP Development Tools (ADT) to make your daily work easier.

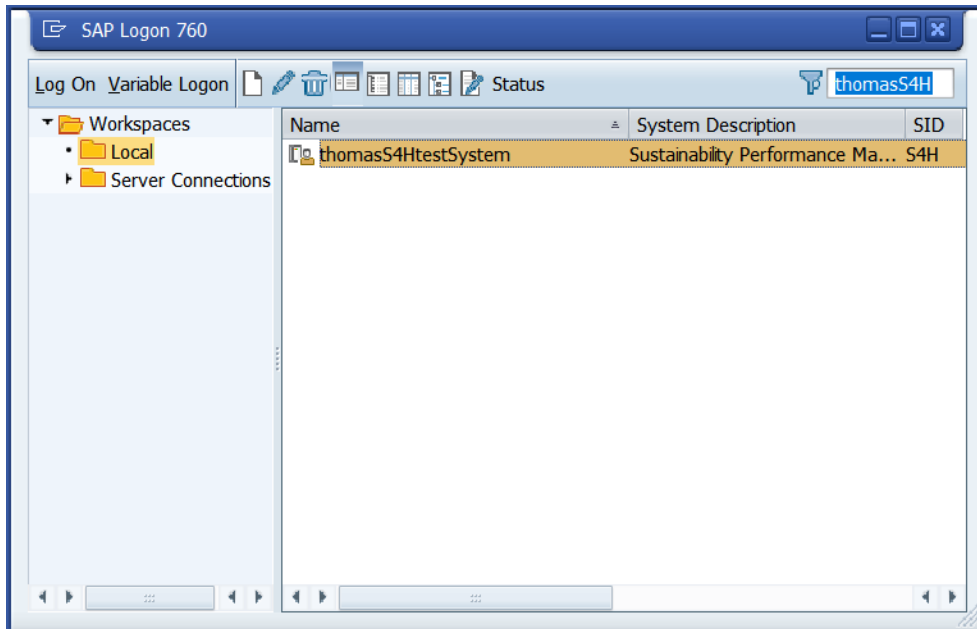
Scope

Focus is on ABAP core development working with the ADT features to create and edit development objects, such as, ABAP classes, programs and so on.

Target Group

Log on to your System via ABAP Project

Reuse the systems of the SAP Logon in ADT




New ABAP Project

System Connection

Associate the new project with an SAP system connection


Define a [new system connection](#) from scratch, or select an existing SAP Logon entry from the list:

thomasS4H

Name	Description	SID
 thomasS4HtestSystem		S4H

<

>



< Back

Next >

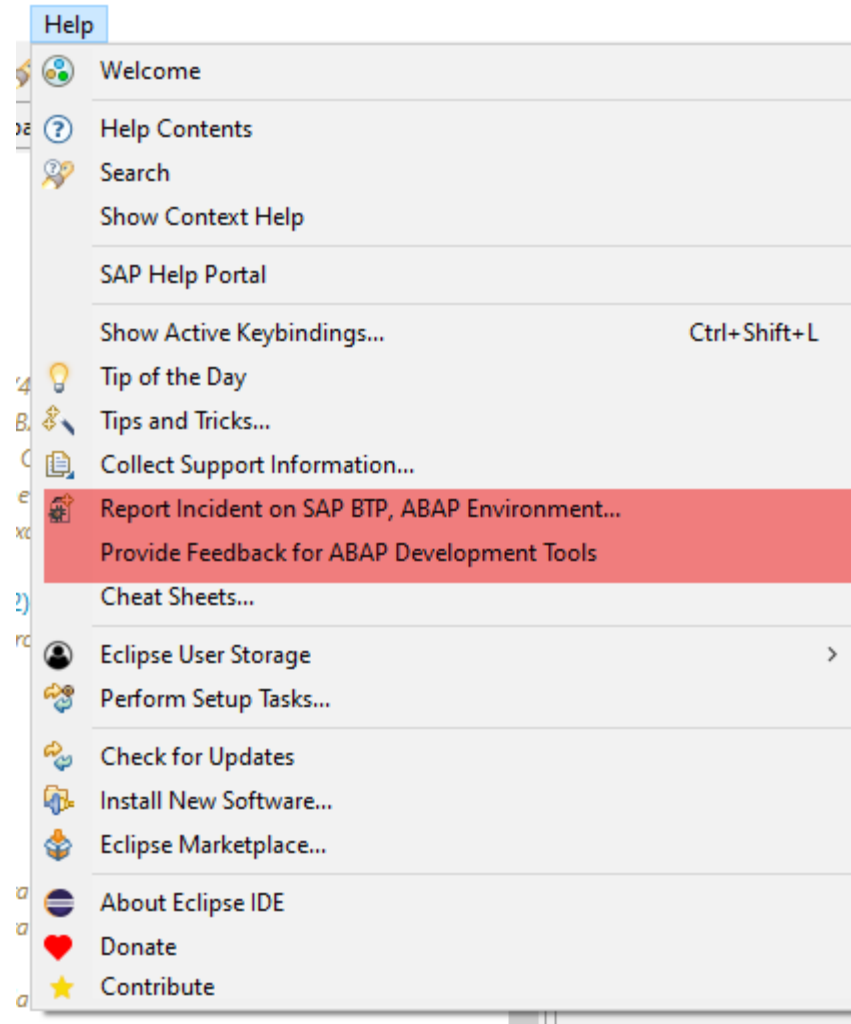
Finish

Cancel

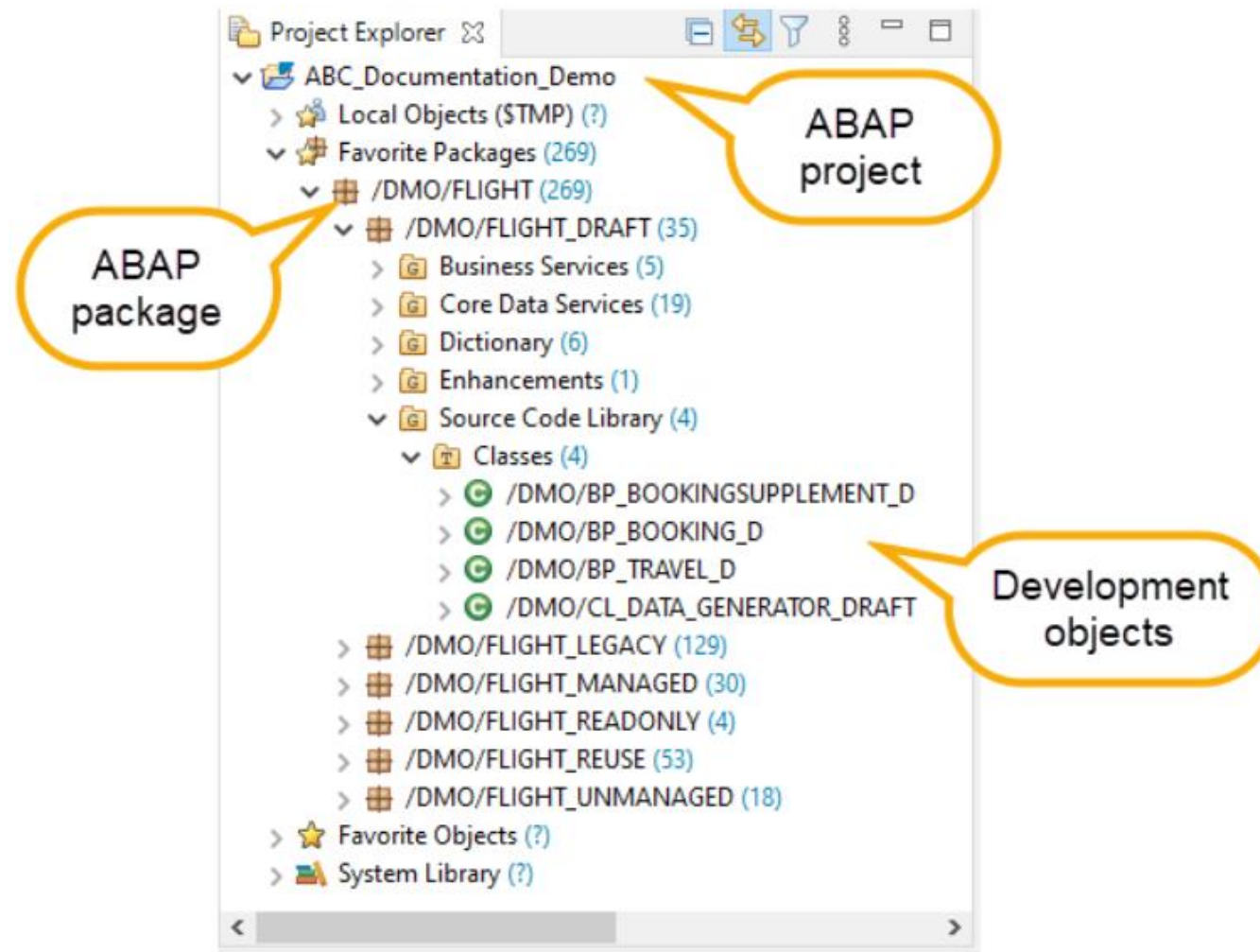
Key Outcomes and Objectives Part 2

1. First Steps in ADT → Part 1
2. Working with ADT → Part 1
3. Basics of ADT → Part 1
4. Important Views and Editors
5. Transport
6. Developer Efficiency

How to Engage with the ADT Team



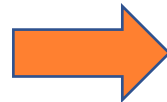
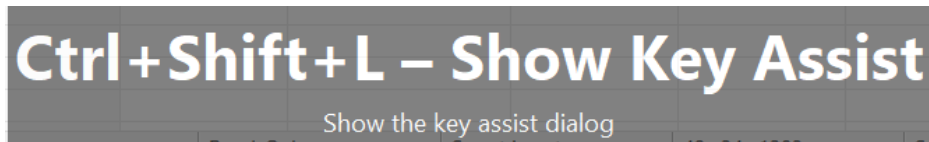
ABAP Project represents a SAP System



Developer Efficiency: Shortcuts

Useful Keyboard Shortcuts: <https://blogs.sap.com/2013/11/21/useful-keyboard-shortcuts-for-abap-in-eclipse/>

Show Key Assist: Press Ctrl + Shift + L

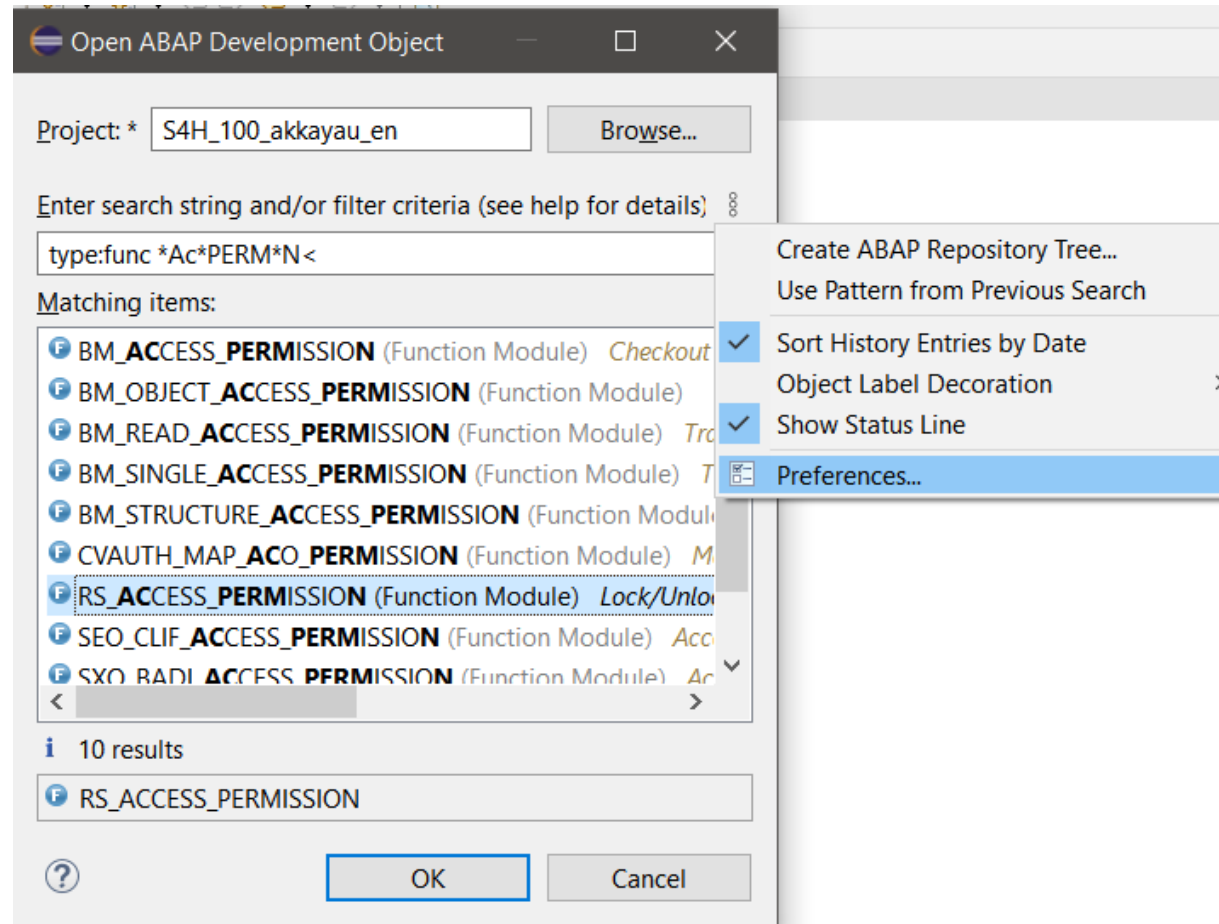


ABAP Where-used List	Ctrl+Shift+G
Activate	Ctrl+F3
Activate Editor	F12
Activate Multiple ABAP Development Objects	Ctrl+Shift+F3
Add Comment	Ctrl+<
Add Javadoc Comment	Alt+Shift+J
Assign to Attribute	Ctrl+2, F
Assign to Local Variable	Ctrl+2, L
Backward History	Alt+Left
Bind to context	Ctrl+Shift+P
Build All	Ctrl+B
Change Method Signature	Alt+Shift+C
Check ABAP Development Object	Ctrl+F2
Close	Ctrl+F4
Close All	Ctrl+Shift+F4
Collapse	Ctrl+Numpad_Subtract
Collapse All	Ctrl+Shift+Numpad_Divide
Compare with Last Used Project	Alt+Shift+C
Content Assist	Ctrl+Space
Context Information	Ctrl+Shift+Space
Copy	Ctrl+Insert

Press 'Ctrl+Shift+L' to open the preference page

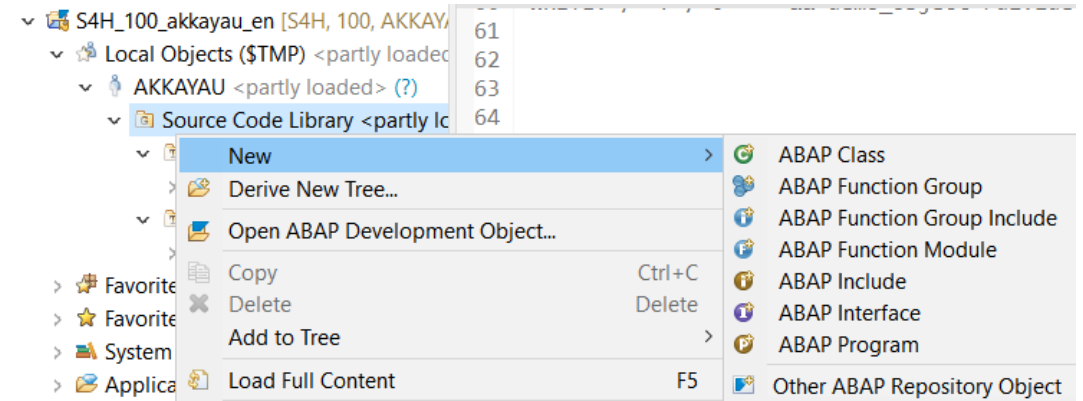
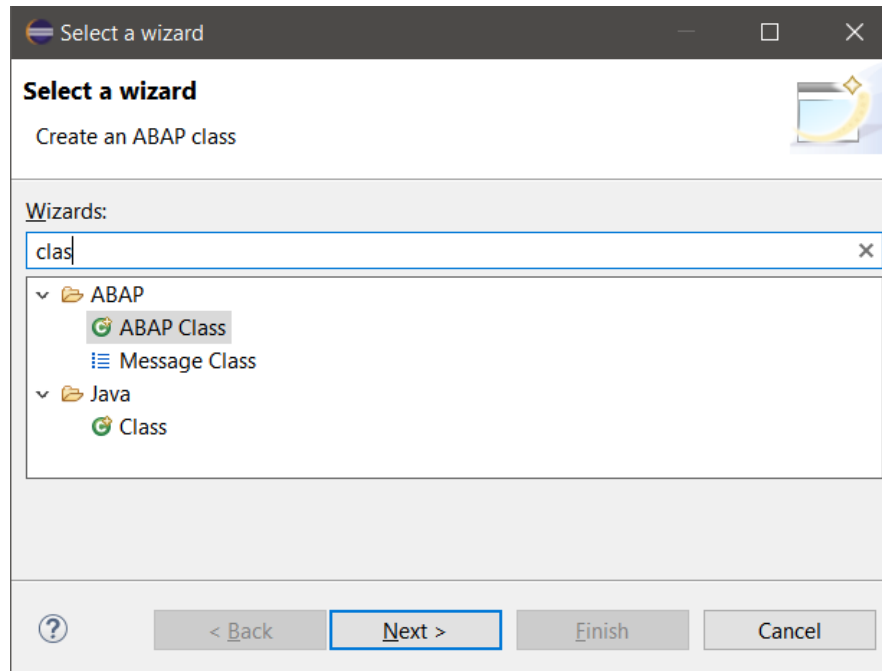
Find Objects

Ctrl + Shift + A: Open ABAP



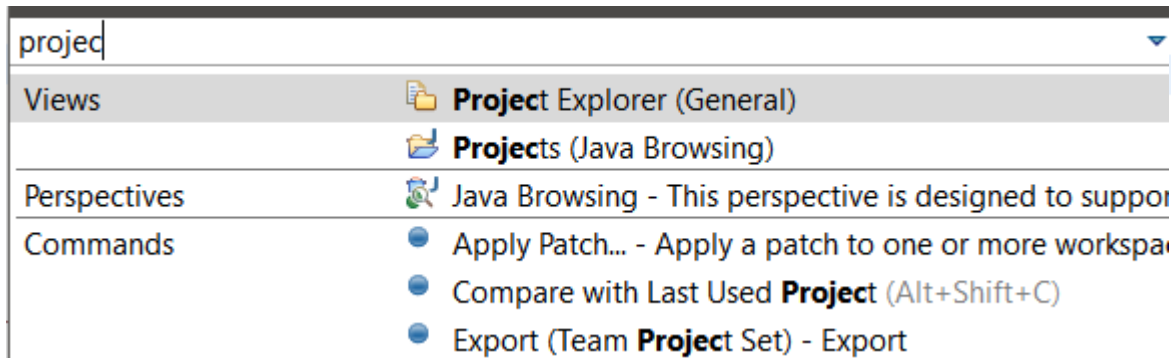
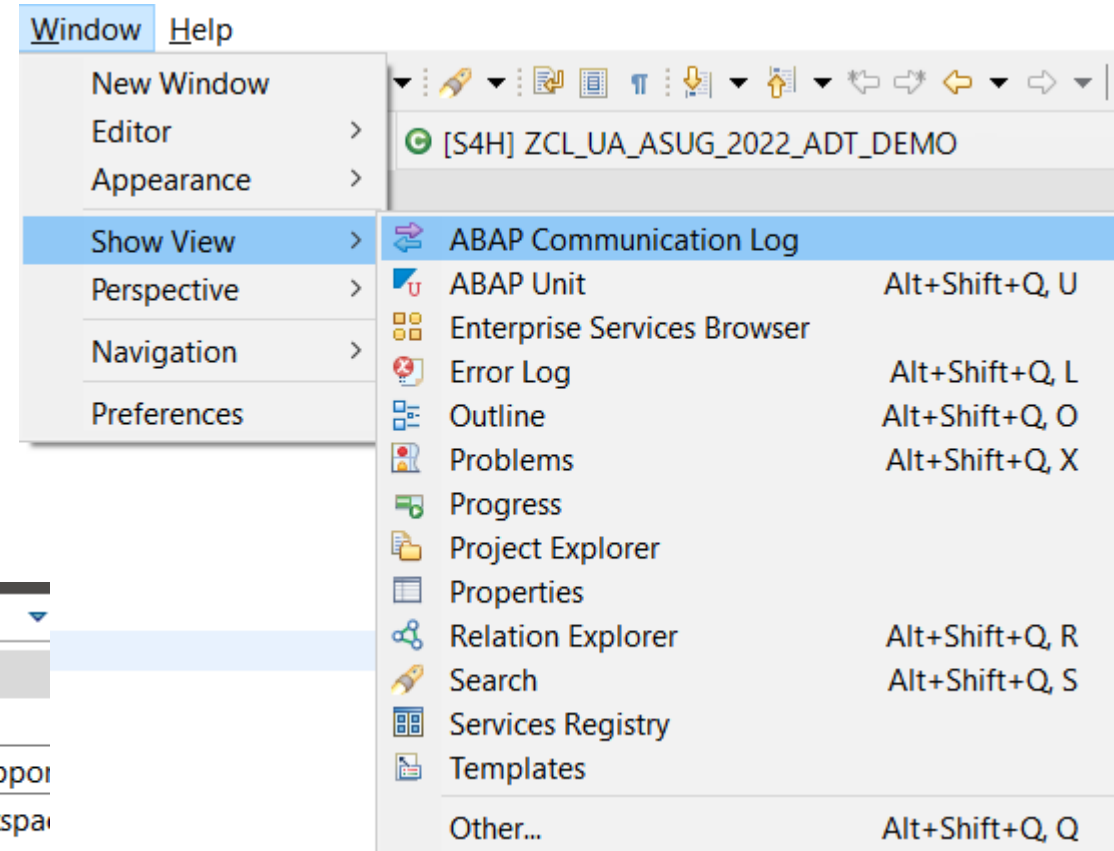
Create Objects

Ctrl + N or via Project Explorer



Find Views

Ctrl +3 or Menu -> Window -> Show View



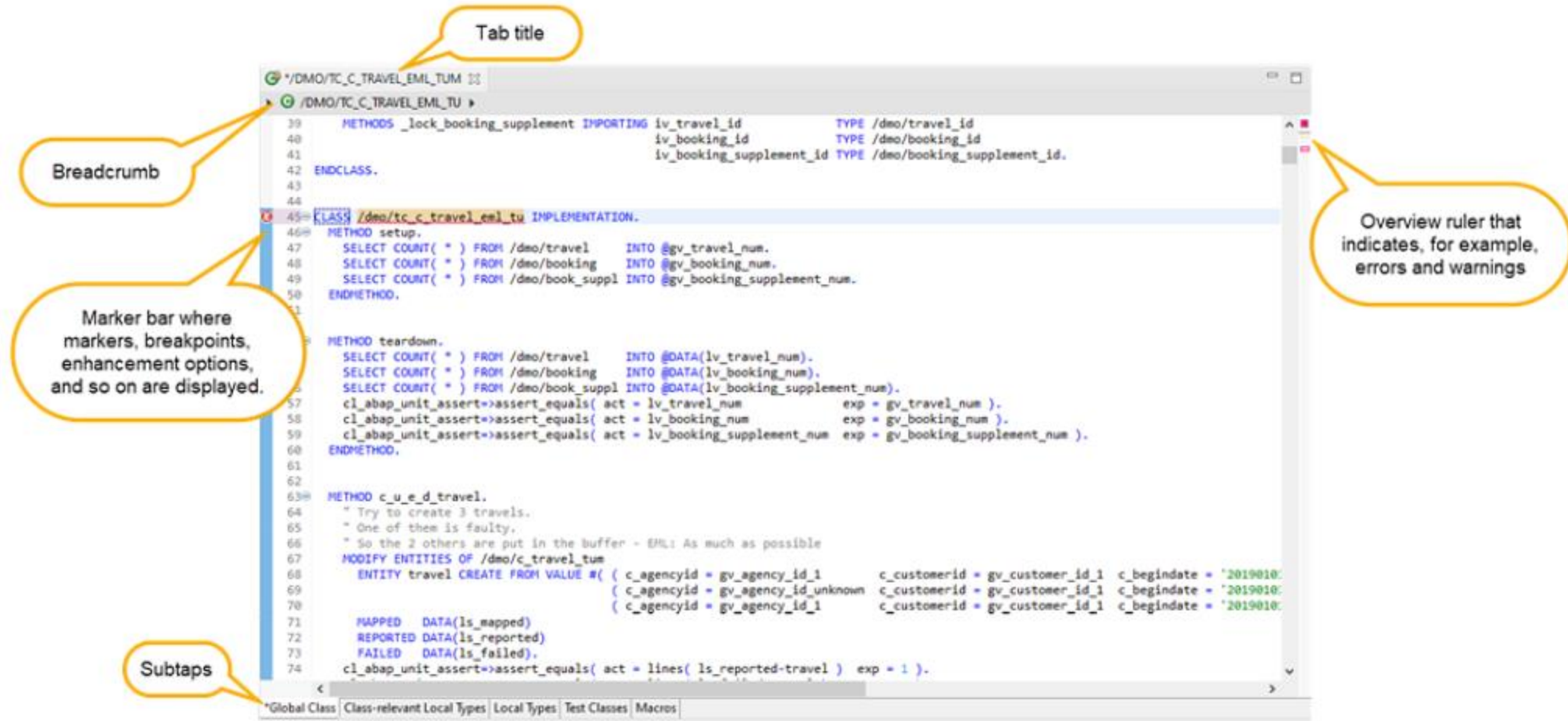
(Quick) Outline / Type hierarchy

Ctrl + O for Quick Outline, press Ctrl + O again to show inherited members
=> Similar for type hierarchy with Ctrl + T

The screenshot displays the SAP IDE interface with the following components:

- Outline View (Left):** A tree view showing the structure of the object `/DMO/TC_C_TRAVEL_EML_TUM`. It lists various methods and data elements, including `C_U_E_D_TRAVEL`, `CU_D_TRAVEL_W_CID`, `C_BOOKING`, `C_BOOKING_SUPPLEMENT`, `C_DEEP_INSERT`, `U_DEEP_INSERT`, `C_BOOKING_FAILED`, `LOCK`, `ROLLBACK_ENTITIES`, `TS_TRAVEL_ID`, `TT_TRAVEL_ID`, `GV_TRAVEL_NUM`, `GV_BOOKING_NUM`, `GV_BOOKING_SUPPLEMENT_NUM`, `SETUP`, `TEARDOWN`, `_LOCK_TRAVEL`, `_LOCK_BOOKING`, `_LOCK_BOOKING_SUPPLEMENT`, and `LTC`. A yellow callout bubble labeled "View toolbar" points to the toolbar of this view.
- Editor (Center):** Displays the source code of the method `c_u_e_d_travel`. The code includes comments and ABAP statements. A yellow callout bubble labeled "Structural elements" points to the code.
- Quick Outline (Right):** A pop-up window showing a condensed version of the outline view. It lists the same methods and data elements as the main outline view. A yellow callout bubble labeled "Quick outline displayed in the editor" points to this window. At the bottom of the window, it says "Press 'Ctrl+O' to show inherited members".

Source-based Editor



Form-based Editor

Data Element: /DMO/OVERALL_STATUS_TGMN

Data Type Information
Specify the data type of the data element

Category:

Data Type:

Length:

Field Labels
Provide field labels and set maximum lengths

Short: 10

Medium: 20

Long: 40

Heading: 55

Additional Properties

Search Help

Name:

Parameter:

Bidirectional Options

Basic Direction:

☒ BIDI Filtering

Parameter ID:

Component Name:

☐ Change Document Logging

☒ Input History

Table: Data Preview

```
@AbapCatalog.dataMaintenance : #ALLOWED  
define table sflight {  
  @AbapCatalog.foreignKey.ke  
  @AbapCatalog.foreignKey.sc  
  key mandt : s_mandt not n  
    with foreign key [0..*,1  
    where mandt = sflight.  
  @AbapCatalog.foreignKey.sc  
  key carrid : s_carr_id not  
    with foreign key scarr  
    where mandt = sflight
```

- Undo
- Revert File
- Save Ctrl+S
- Open ABAP Type Hierarchy F4
- Quick Type Hierarchy Ctrl+T
- Navigate To F3
- Open in Project Ctrl+Alt+P >
- Open With >
- Data Preview

[S4H] SFLIGHT [S4H] SFLIGHT x »2

Raw Data Show Log Max. Rows: 100

Filter pattern Entries Select Columns Add filter

MANDT	CARRID	CONNID	FLDATE
100	AA	0017	2018-09-27
100	AA	0017	2018-10-29
100	AA	0017	2018-11-30
100	AA	0017	2019-01-01
100	AA	0017	2019-02-02
100	AA	0017	2019-03-06
100	AA	0017	2019-04-07
100	AA	0017	2019-05-09
100	AA	0017	2019-06-10

```
"! Climbing is increasing the <strong>altitude</strong> of an aircraft. The opposite is {@link cl_demo_plane.meth:descent }.  
"! @parameter climb_angle | The angle between a horizontal plane and the actual flight path  
"! @parameter climb_rate | The vertical speed - rate of altitude with respect to time.  
METHODS climb IMPORTING climb_angle TYPE float climb_rate TYPE float.
```

climb

```
importing climb_angle type float  
          climb_rate type float
```

Documentation

Climbing is increasing the **altitude** of an aircraft. The opposite is [descent](#).

Parameters

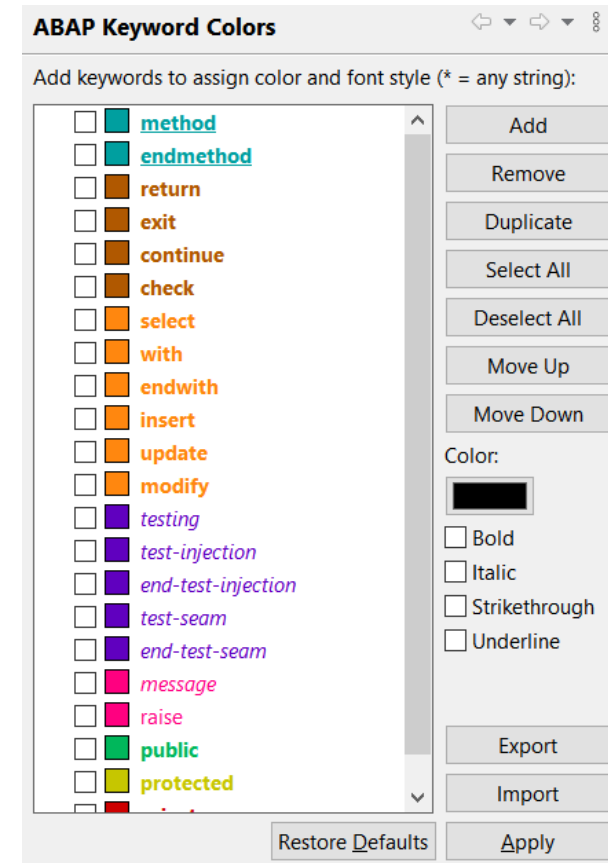
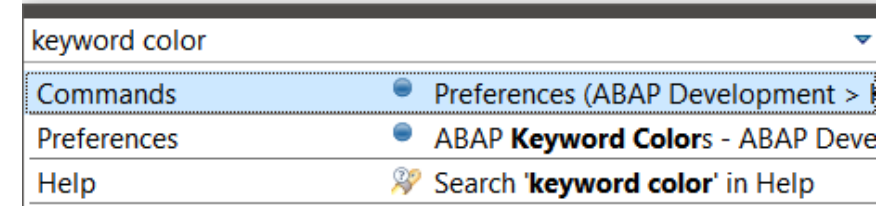
- ▢ **climb_angle** The angle between a horizontal plane and the actual flight path
- ▢ **climb_rate** The vertical speed - rate of altitude with respect to time.

Keyword Coloring

```
PROTECTED SECTION.  
PRIVATE SECTION.  
METHODS internal_divide2  
IMPORTING  
    i_var1          TYPE i  
    i_var2          TYPE i  
RETURNING  
    VALUE(r_result) TYPE i  
RAISING  
    zcx_zerodivide.
```

```
PROTECTED SECTION.  
PRIVATE SECTION.  
METHODS internal_divide2  
IMPORTING  
    i_var1          TYPE i  
    i_var2          TYPE i  
RETURNING  
    VALUE(r_result) TYPE i  
RAISING  
    zcx_zerodivide.
```

Ctrl + 3



Apply



Properties View

Properties

/DMO/ACTN_CD_01

Subtabs

General

API State

Toolbar

Package: [TEST FLIGHT01_LEGACY](#) -> [TEST FLIGHT01](#) -> [\\$TMP](#) -> .

Version: Active

Description: Flight Reference Scenario: Action Code

Responsible: 杨志 (杨志) Planning: Test Report

Created on: 2020-08-04 14:11:11

Last Changed by: 杨志 (杨志) Planning: Test Report

Last Changed on: 2020-08-04 14:11:11

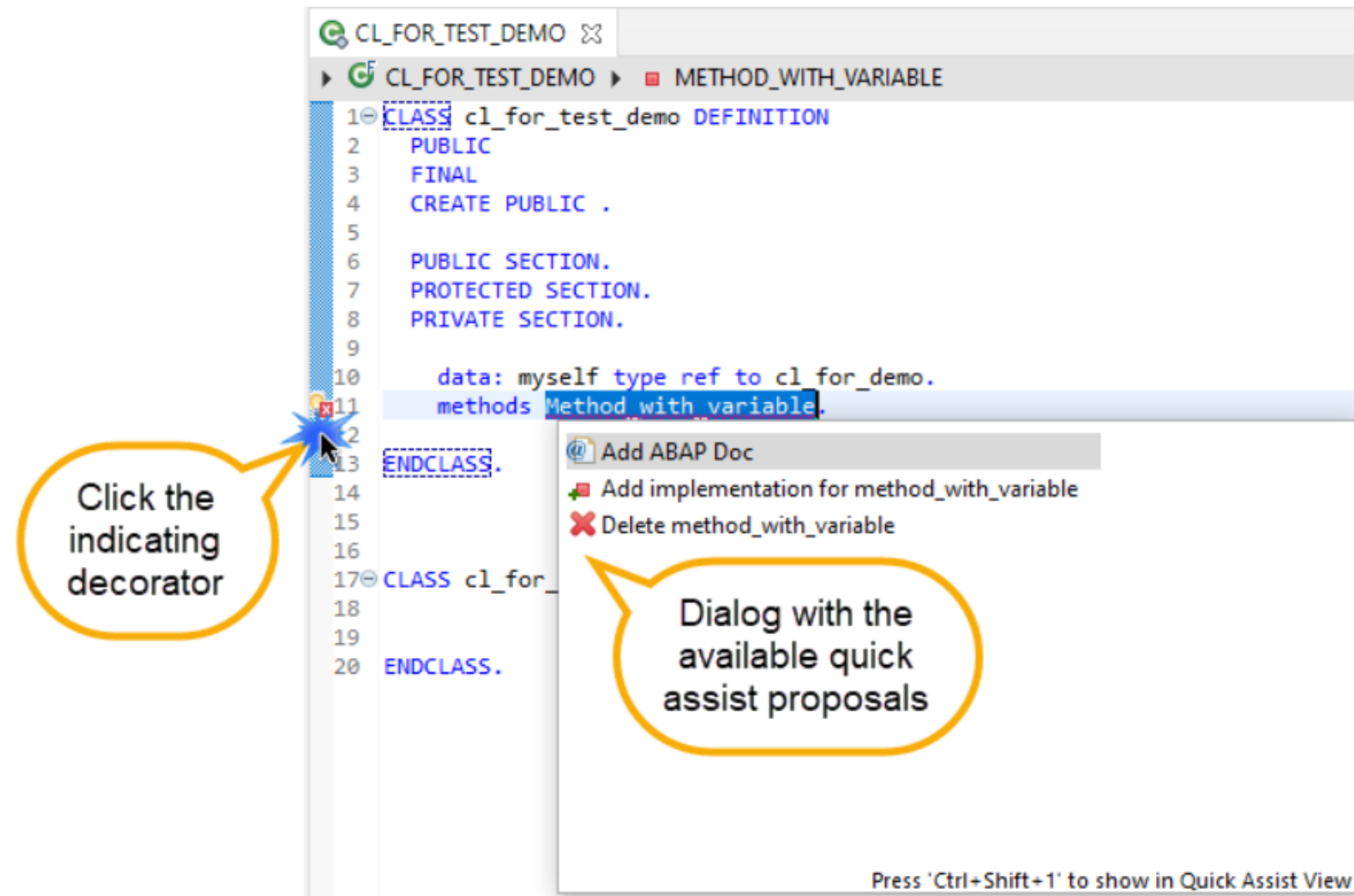
Original Language: 中文

Original System: .Net

Application Component:

Object details

Quick Assist (Ctrl + 1)



Troubleshooting

The screenshot displays the SAP ABAP Debugger interface with several key components highlighted by callouts:

- Debugger actions:** A callout points to the toolbar at the top of the debugger window, which contains icons for various debugging actions like stepping, running, and setting breakpoints.
- Position within the source code where the breakpoint is set:** A callout points to line 187 in the source code, which is highlighted in blue. The code at this line is: `<SELECT> FROM tadir`.
- Debug view displaying the current session and call stack:** A callout points to the left-hand pane, which shows the Project Explorer and the Call Stack. The call stack lists the sequence of method calls leading to the current breakpoint.
- Variables view:** A callout points to the right-hand pane, which displays the current state of variables. The table below shows the variables and their values:

Name	Value	Actual Type	Technical T
<Enter variable>			
SY-SUBRC	0	SYST_SUBRC	I
ME	(0x65"CLASS...	(Local) Class LT...	Ref to LT...
TAB_TADIR	[0x22(464)]Init...	\TYPE= %T000...	Standard
STR_MYTADIR	Structure: deep	\TYPE= %T000...	Structure: c
TEST	X	\TYPE= %T000...	C
L_RAW		XSTRING	XString
L_TAB_TADIR	[0x22(464)]Init...	\TYPE= %T000...	Standard T
L_TAB_TADIR2	[0x22(464)]Init...	\TYPE= %T000...	Standard T

Transport Organizer

The screenshot displays the SAP Transport Organizer interface. At the top, a window titled 'Request: ABC_DOCUMENTATION_DEMO (Workbench)' is shown. A callout bubble points to this title bar, stating 'Details about the selected transport request'. Below the title bar is the 'Properties' section, which includes fields for 'Short Description' (Local transport request for help server configuration), 'Long Description', 'Owner', 'Target', 'CTS Project', 'Status' (Modifiable), 'Last Changed', and 'Source Client' (000). Below the properties is the 'Objects' section, which contains a tree view on the left and a table on the right. A callout bubble points to the tree view, stating 'Transport Organizer view'. The tree view shows a hierarchy: 'All Objects' > 'ABC_DOCUMENTATION_DEMO' > 'Workbench' > 'Modifiable'. The table on the right has columns: 'Object Name', 'Description', 'Object Type', 'Program ID', 'Lock/Import Sta...', and 'IMG Activity'. Below the 'Objects' section is the 'Overview' tab, which shows a table of transport requests. A callout bubble points to this table, stating 'Structure and details of the transport requests and tasks'. The table has columns: 'Transport Request', 'Owner', 'Type', 'Description', and 'Last Changed'. The table contains three rows: 'ABC_DOCUMENTATION_DEMO' (Workbench), 'ABC_DOCUMENTATION_DEMO' (Modifiable), and 'ABC_DOCUMENTATION_DEMO' (Development/Correction Table Contents).

Request: ABC_DOCUMENTATION_DEMO (Workbench)

Details about the selected transport request

Properties

Short Description: Local transport request for help server configuration

Long Description:

Owner: SAP_BW_BW_210 - SAP_BW_BW_210

Target: SAP_BW_BW_210 - SAP_BW_BW_210

CTS Project:

Status: Modifiable

Last Changed: Thursday, July 23, 2015, 5:16:18 PM

Source Client: 000

Objects

Transport Organizer view

Overview

Transport Request	Owner	Type	Description	Last Changed
ABC_DOCUMENTATION_DEMO [user filter: SAP_BW_BW_210]				
Workbench				
Modifiable				
ABC_DOCUMENTATION_DEMO	SAP_BW_BW_210		Kernel home development	
ABC_DOCUMENTATION_DEMO	SAP_BW_BW_210		Local transport request for help server ...	Thursday, July 23, 2015, 5:16:18 PM
ABC_DOCUMENTATION_DEMO	SAP_BW_BW_210	Development/Correction Table Contents	Local transport request for help server ...	Thursday, July 23, 2015, 5:16:18 PM

Structure and details of the transport requests and tasks

Questions?

For questions after this session, contact us at uenal.akkaya@sap.com and thomas.fiedler@sap.com.

Thank you.