

Modern ABAP

Which SAP NetWeaver Version do you use?

```
DATA lt_umsatz TYPE tt_umsatz OCCURS 10 WITH HEADER LINE.  
DATA lv_gesamt_umsatz TYPE P LENGTH 16 DECIMALS 9.  
  
LOOP AT lt_umsatz.  
  
    ADD lt_umsatz-umsatz TO lv_gesamt_umsatz.  
  
ENDLOOP.
```

Inline Declarations

Since NW 7.40

```
DATA lv_test TYPE i.  
lv_test = 1.
```

instead

```
DATA(lv_test) = 1.
```

Advantages:

- Less lines
- Fewer "declaration headers" in methods, thus avoid having unneeded variables around
- Type inference, so less problems when refactoring code

Inline Declarations

```
obj→meth(  
    EXPORTING  
        iv_value = 42  
    IMPORTING  
        ev_result = DATA(lv_test)  
).
```

```
LOOP AT ... INTO DATA(ls_line).
```

```
LOOP AT ... ASSIGNING FIELD-SYMBOL(<ls_line>).
```

String Templates


```
WRITE |Hello { sy-uname }|.
```

```
WRITE |Hello | && sy-uname.
```

Constructor Expressions

Since NW 7.40

Classes and Type Conversion

```
NEW cl_meetup( CONV meetup_attendees( 11 ) )→get_topic( ).  
NEW cl_meetup( CAST cl_my_object( cl_factory⇒get( ) ) ).      " Nested™
```

Exceptions

```
RAISE EXCEPTION TYPE cx_exception  
  MESSAGE ID 'MEETUP'  
  TYPE 'E'  
  NUMBER '11'  
  WITH 'Cologne'.  
  
RAISE EXCEPTION NEW cx_exception_fancy( ).
```

Conditional Statements

```
DATA(lv_drink) = SWITCH meetup_drink( sy-daytime
    WHEN 'MORNING'
        THEN 'Milk Coffee'
    WHEN 'NOON'
        THEN 'Espresso'
    ELSE
        'Cacao'
).
```

```
DATA(lv_food) = COND meetup_food(
    WHEN sy-outside_temperature > 25
        THEN 'Ice Cream'
    ELSE
        'Sandwich'
).
```

Reference Statement

```
GET REFERENCE OF lt_pos_data INTO lr_pos_data.
```

```
lr_pos_data = REF #( lt_pos_data ).
```

Table Expressions

```
lt_top_products = FILTER #( lt_products WHERE turnover ≥ lc_cash_cow_turnover ).
```

```
IF NOT line_exists( lt_products[ name = 'Club Mate' ] ).  
    me→buy_mate( ).  
ELSE.  
    lt_products[ name = 'Club Mate' ]→drink( )    "⚠ cx_sy_itab_line_not_found  
ENDIF.
```

```
lt_abap_devs = FILTER #( lt_employees USING KEY skills WHERE  
    abap_objects = abap_true  
    hana = abap_true  
).
```

FOR Expression

```
lt_product_range = VALUE #( FOR lt_products IN <ls_> (  
    sign = 'I'  
    option = 'EQ'  
    low = <ls_>-matnr  
) ).
```

xsdbool

```
DATA(lv_editable) = xsdbool( sy-ucomm = 'EDIT' )
```


Increment and decrement variables

Since NW 7.54

```
lv_var += 1  
lv_var -= 1
```

ABAP SQL

formerly open SQL

```

SELECT
  uuid( ) as guid,           " 7.54
  matnr,                     " 7.40 SP05
  maktx,
  'X' as translated          " 7.50
FROM
  maktx
CROSS JOIN                   " 7.51
  @lt_whatever as lt_        " 7.52
WHERE
  spras = 'D' AND
  maktx IS NOT INITIAL       " 7.53
INTO TABLE
  @DATA(lt_).                " 7.40 SP08

```

- uuid() counterpart for HANA SYSUUID function

Which IDE do you use?



ABAP Development Tools (ADT)

formerly ABAP in Eclipse (AiE)

ABAP Development Tools

workspace-2019-09 - H02_100_bauer_en/.adt/classlib/classes/cl_abap_random/cl_abap_random.aclass - Eclipse IDE

File Edit Source Code Navigate Search Project Run Window Help

Project Explorer Systems

- DM1_100_bauer_en [DM1, 100, BAUER, EN]
- DM1_250_bauer_en [DM1, 250, BAUER, EN]
- DM1_260_bauer_en [DM1, 260, BAUER, EN]
- H02_100_bauer_en [H02, 100, BAUER, EN]
- H03_100_bauer_en [H03, 100, BAUER, EN]
- H09_100_bauer_en [H09, 100, BAUER, EN]

[H02] CL_ABAP_RANDOM

```
96      importing
97      !SEED type I .
98      ENDCLASS.
99
100
101
102 CLASS CL_ABAP_RANDOM IMPLEMENTATION.
103
104
105 method CREATE .
106
107     CREATE OBJECT prng.
108
109 *   No initialisation if seed is not given.
110 *   Kernel's PRNG does an implicit initialisation upon the first request for a number.
111     prng->left = 1.
112
113 IF seed IS SUPPLIED.
114     prng->InitPrng( seed ).
115 ENDIF.
116 endmethod.
117
118
119
120 method CREATE_TESTER .
121
122     CREATE OBJECT prng.
123
124 *   No initialisation if seed is not given.
125 *   Kernel's PRNG does an implicit initialisation upon the first request for a number.
126     prng->left = 1.
```

Global Class Class-relevant Local Types Local Types Test Classes (non existent) Macros

Outline Properties ABAP Element L... ABAP Language...

CL_ABAP_RANDOM

- ABAP_PRNG_STATE
- P31_0
- FALSE FALSE
- TRUE TRUE
- INITF flag: state computed
- LEFT number of left states
- NEXT kernel: pointer to next state
- STATE PRNG state array
- CREATE Factory
- CREATE_TESTER Factory for standard test PRNG
- DECFLOAT16 Compute a packed RN of type decfloat16
- DECFLOAT34 Compute a packed RN of type decfloat34
- FLOAT Compute a floating-point RN in [0,1)
- INITPRNG Initialise a PRNG (compute state array)
- INT Compute a signed integer RN
- INT31 Compute a 31 bit integer RN (i.e. 0 <= value <= 0x7FFFFFFF)
- INT8
- INT8INRANGE compute a 64 bit integer RN in [a,b]
- INTINRANGE compute a random integer RN in [a,b]
- PACKED Compute a packed RN (31 digits, no decimals)
- PACKEDINRANGE compute a packed RN in [a,b] (31 digits, no decimals)
- SEED compute a random seed (integer)
- TESTINIT special initialisation for standard test

Problems Templates Bookmarks Task Reposito... Task List Feed Reader Transport Org... ABAP Unit Search (x)= Variables History ABAP Trace R... ABAP Traces Dictionary Log ABAP Type Hi... ATC Problems Error Log

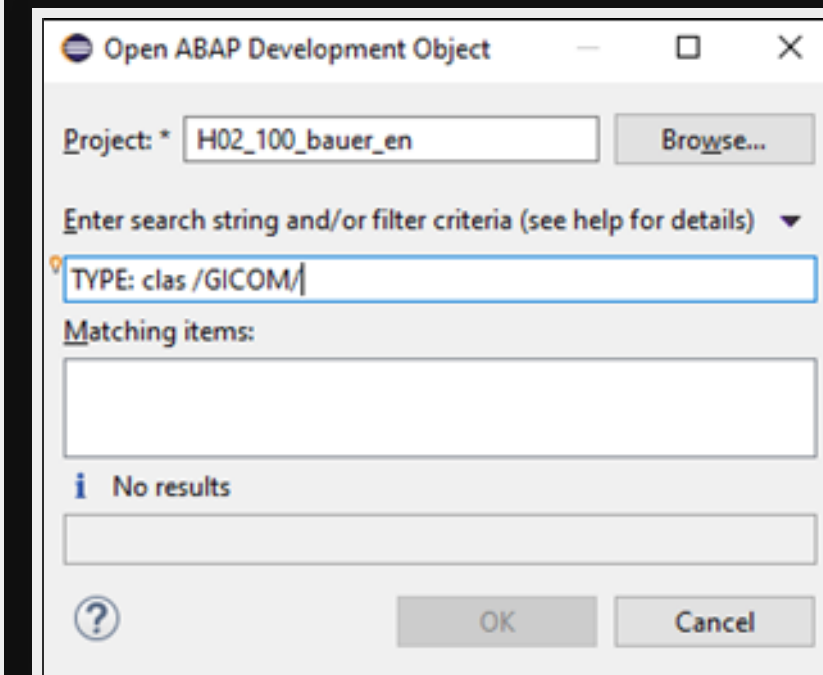
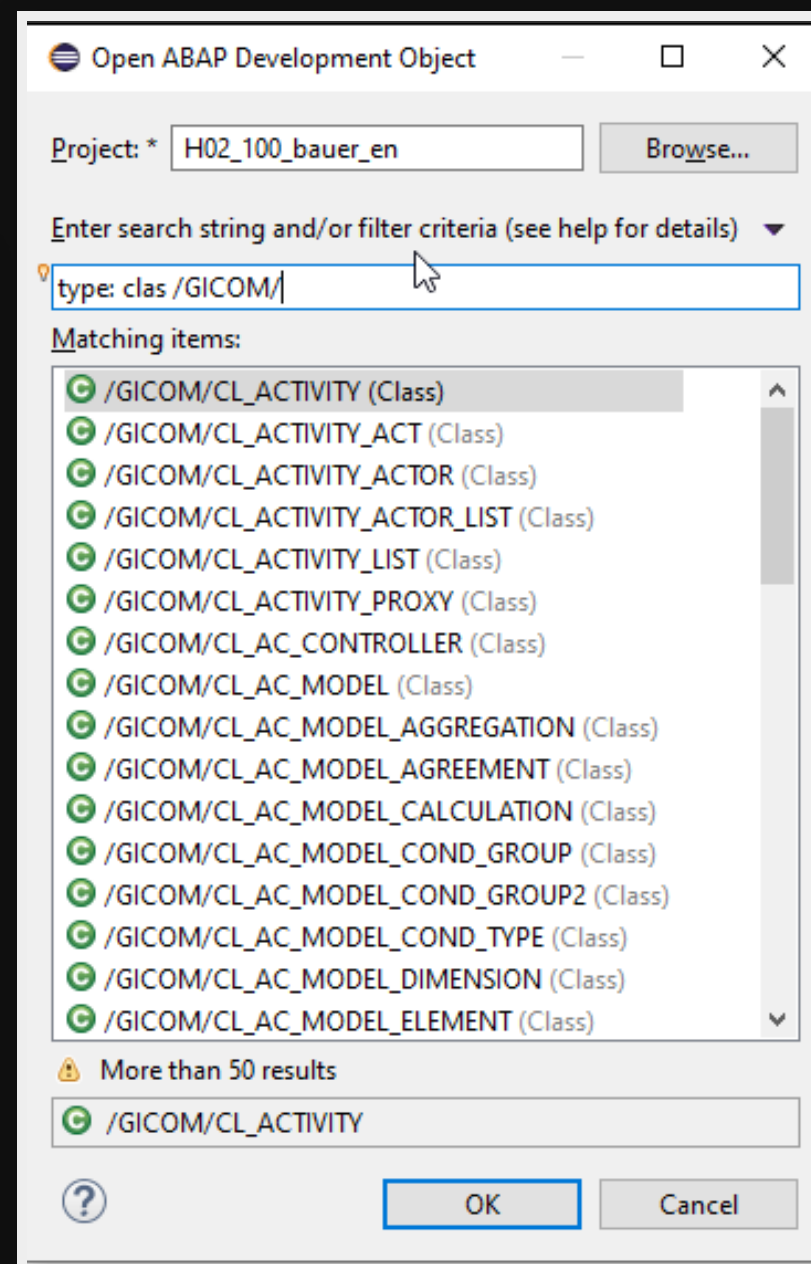
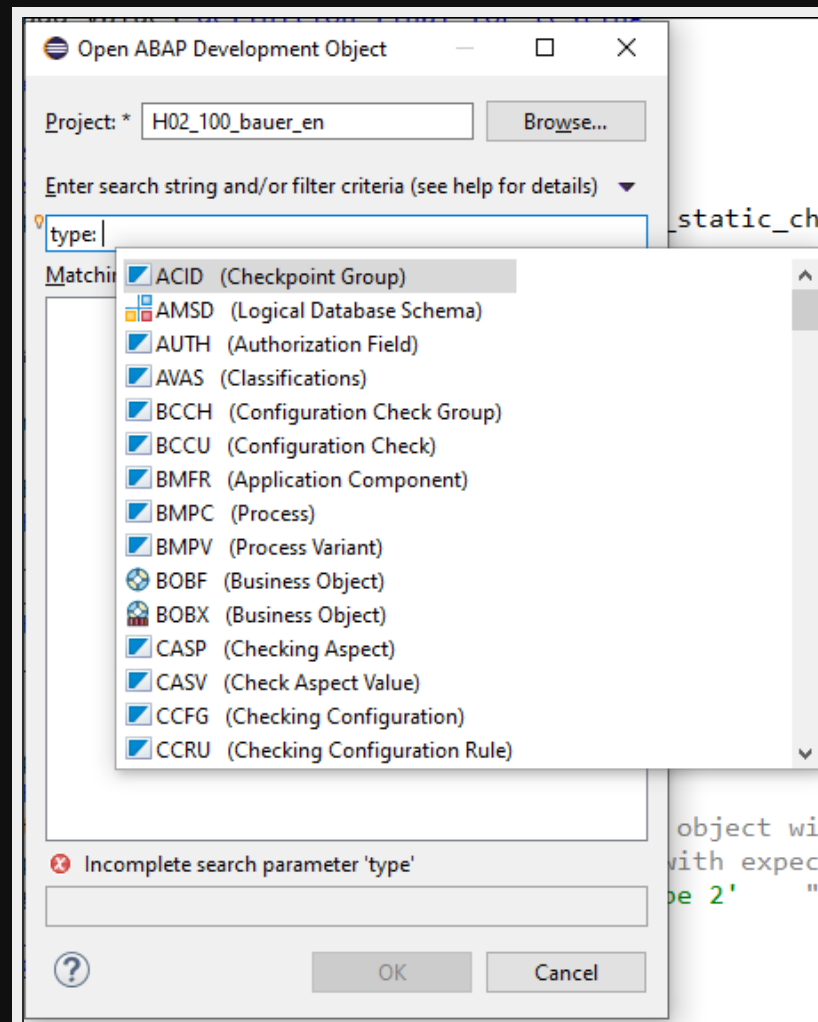
0 errors, 1 warning, 0 others

Description	Resource	Path	Location	Type
Warnings (1 item)				
The global class "CL_ABAP_RANDOM" is not registered in Memory Inspector for the attribute "NEXT". Refer to the file include/abmi	cl_abap_random.aclass	/H02_100_bauer_en...	Unknown	ABAP Syntax ...

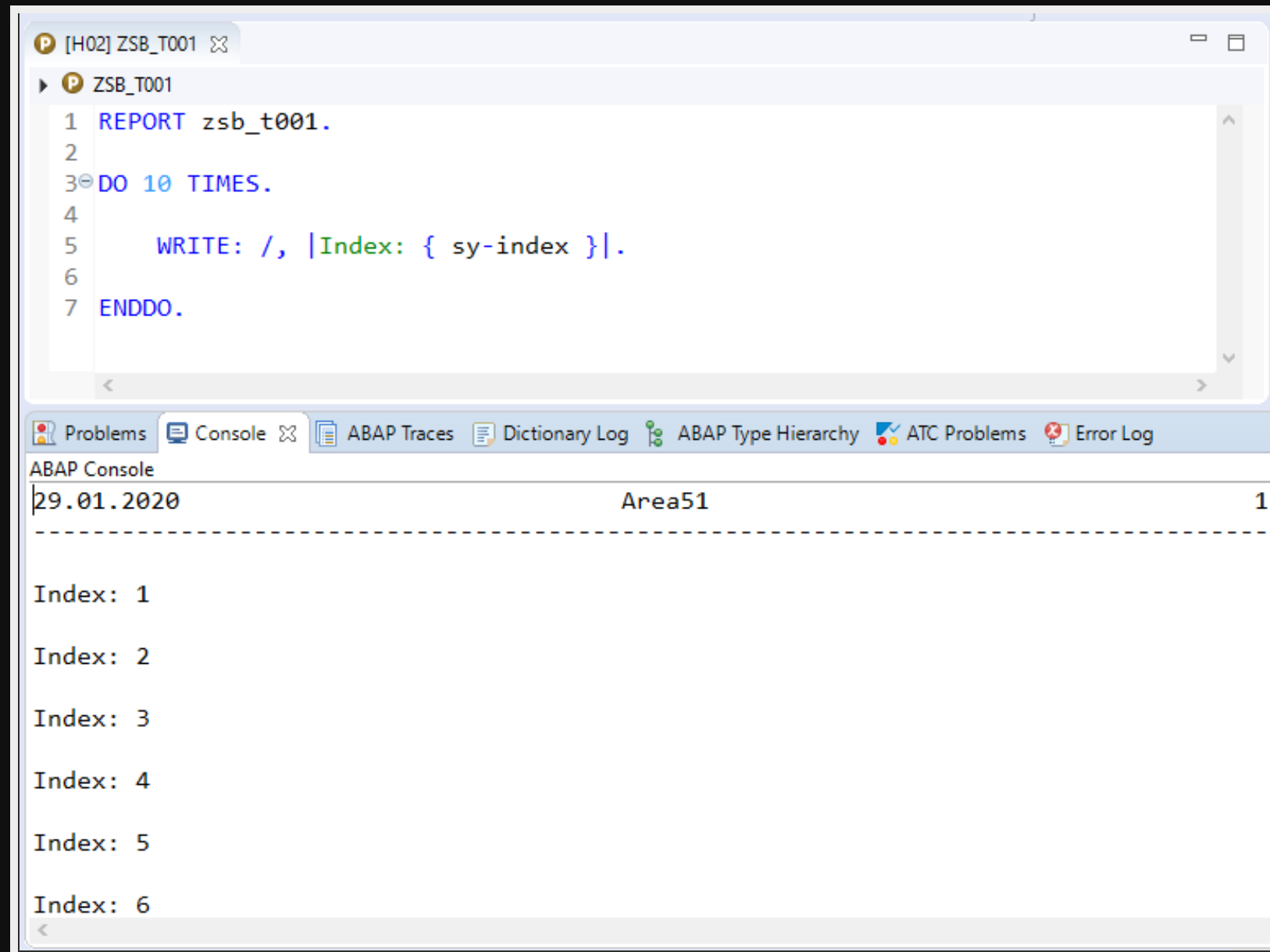
Read-Only Smart Insert 107: 24: 2505 H02, 100, gicomsap18, BAUER, EN

Search Development Objects

CTRL + SHIFT + A



F9: Run Report in Console

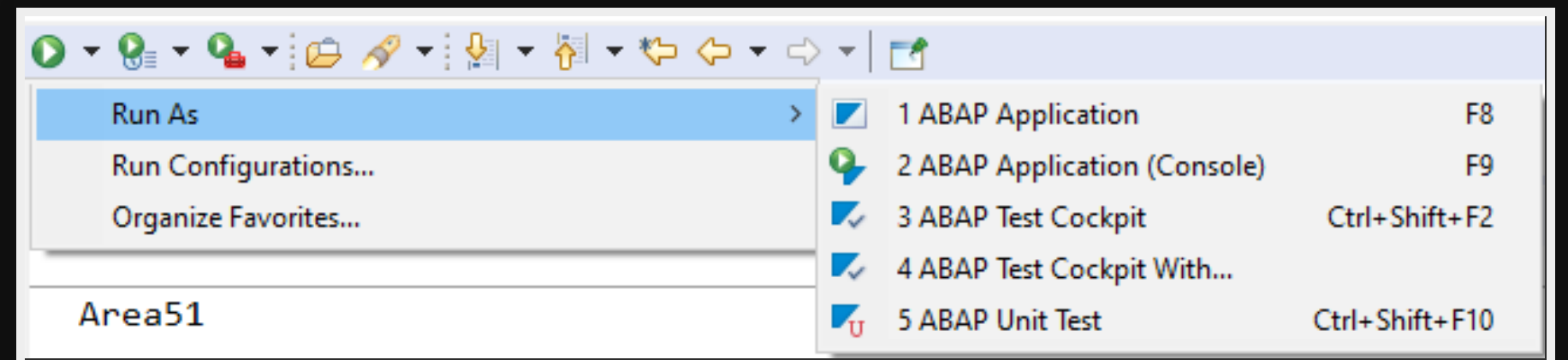


The screenshot shows the SAP IDE interface. The top editor window displays the ABAP code for report ZSB_T001:

```
1 REPORT zsb_t001.  
2  
3 DO 10 TIMES.  
4  
5     WRITE: /, |Index: { sy-index }|.  
6  
7 ENDDO.
```

Below the editor is the 'ABAP Console' window. It shows the execution date '29.01.2020' and the user 'Area51'. The output of the report is displayed as follows:

```
Index: 1  
Index: 2  
Index: 3  
Index: 4  
Index: 5  
Index: 6
```



Debug Settings

Properties for H02_100_bauer_en

type filter text

- > Resource
- ▼ ABAP Development
 - ABAP Test Cockpit
 - Debug**
 - > Editors
 - Project Explorer
 - SQL Monitor
 - Builders
 - Coverage
 - Project Facets
 - Project Natures
 - Project References
 - Refactoring History
 - Run/Debug Settings
 - Semantic Resource
- > Task Repository
 - Task Tags
- > Validation
 - WikiText

Debug

[Configure Workspace Settings...](#)

☒ Enable debugger

Breakpoint activation is effective for

☐ This project only

☒ Logon user (BAUER)

☐ User:

☐ Terminal ID

Enables debugging of any request for the logon user. This includes embedded SAP GUI and ABAP Unit requests as well as external requests like HTTP and RFC.

Advanced Settings

☐ Allow debugging of tool requests

☐ Show data aging temperature

Debug information

Terminal ID: A7D7927ABC2F4B7F907B10D2E6134EEA

IDE ID: 65653FD700C230356A3241438EC07A23

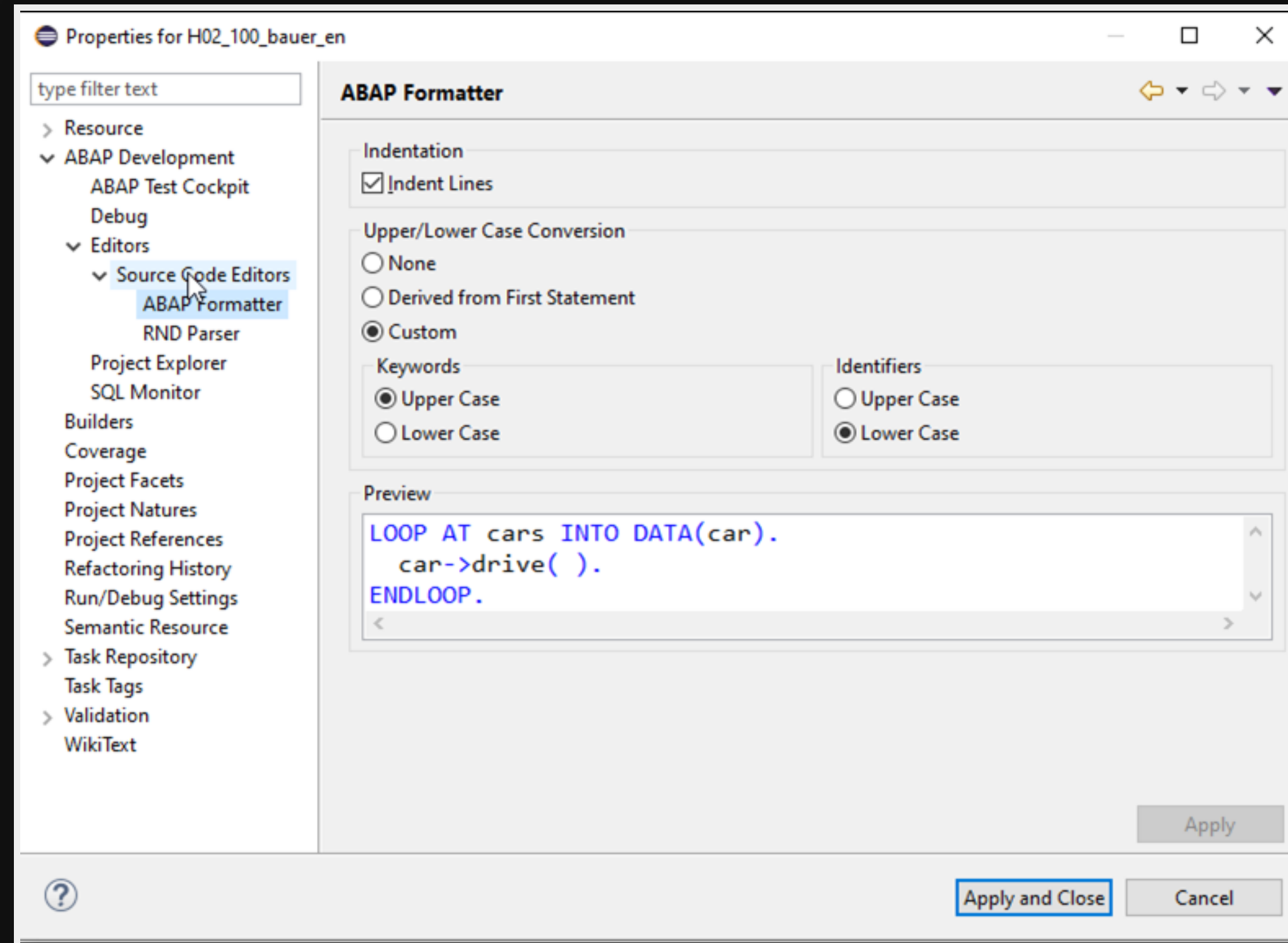
AMDP Debugger

Compile in Debug Mode:

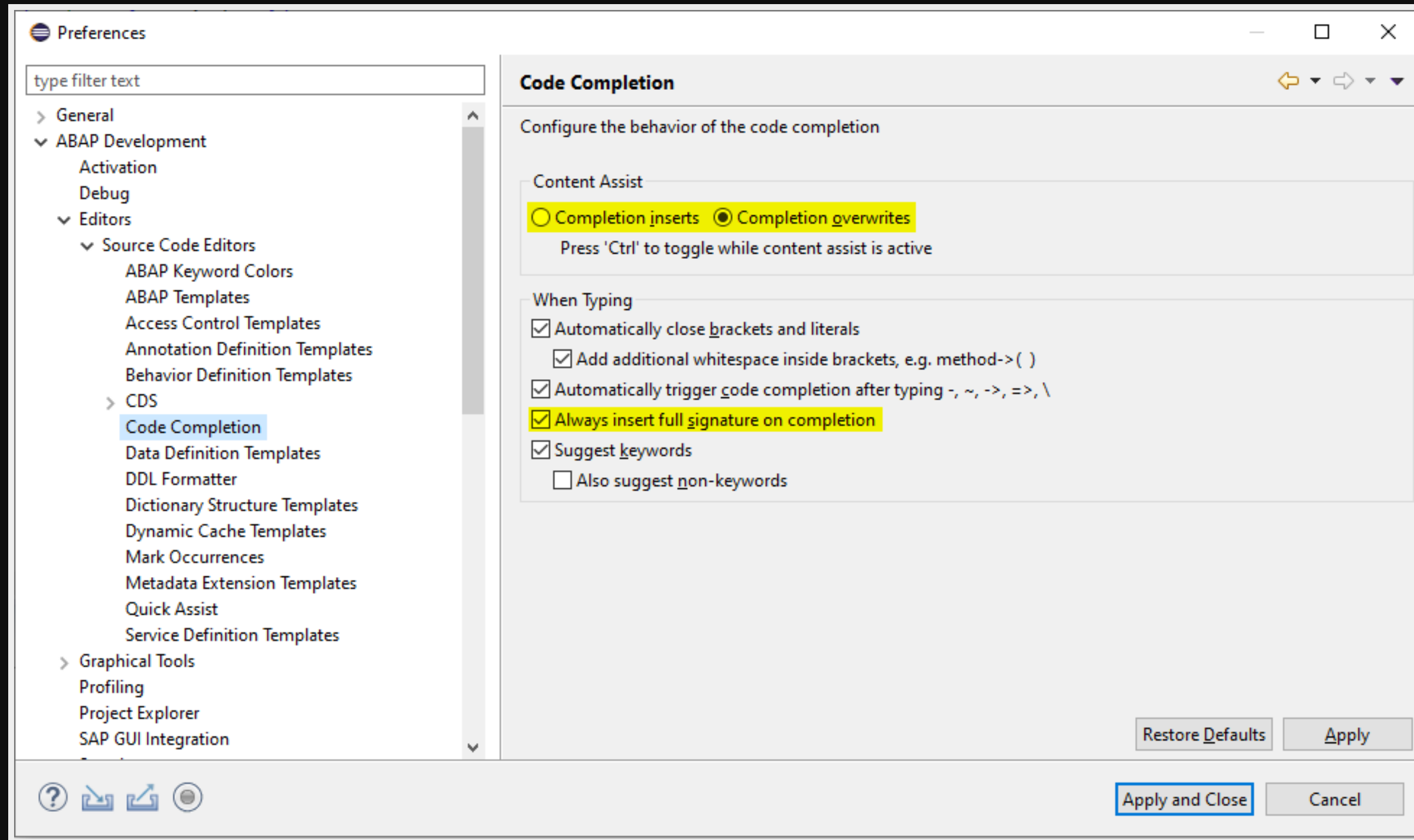
Restore Defaults Apply

? Apply and Close Cancel

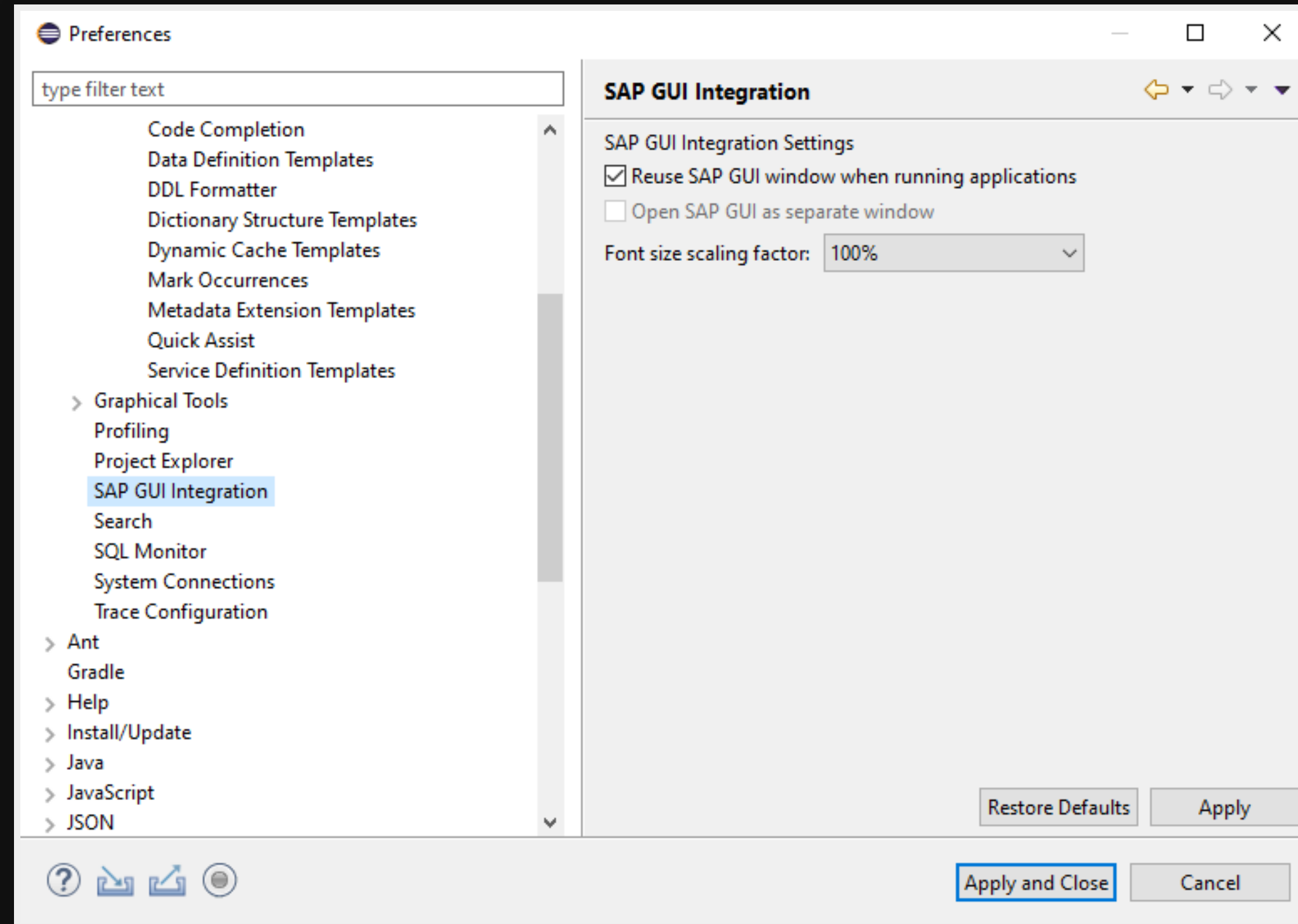
ABAP Formatter



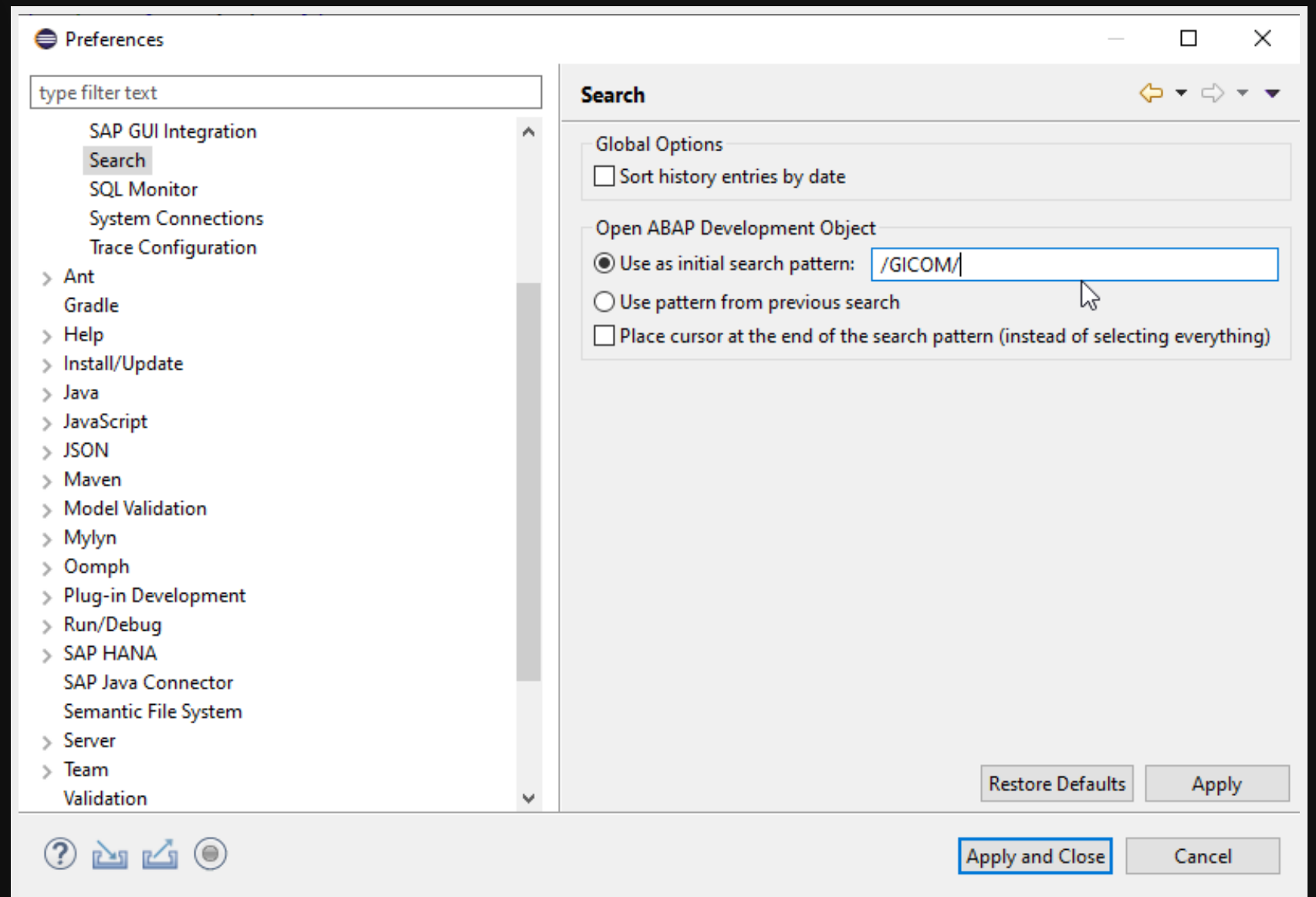
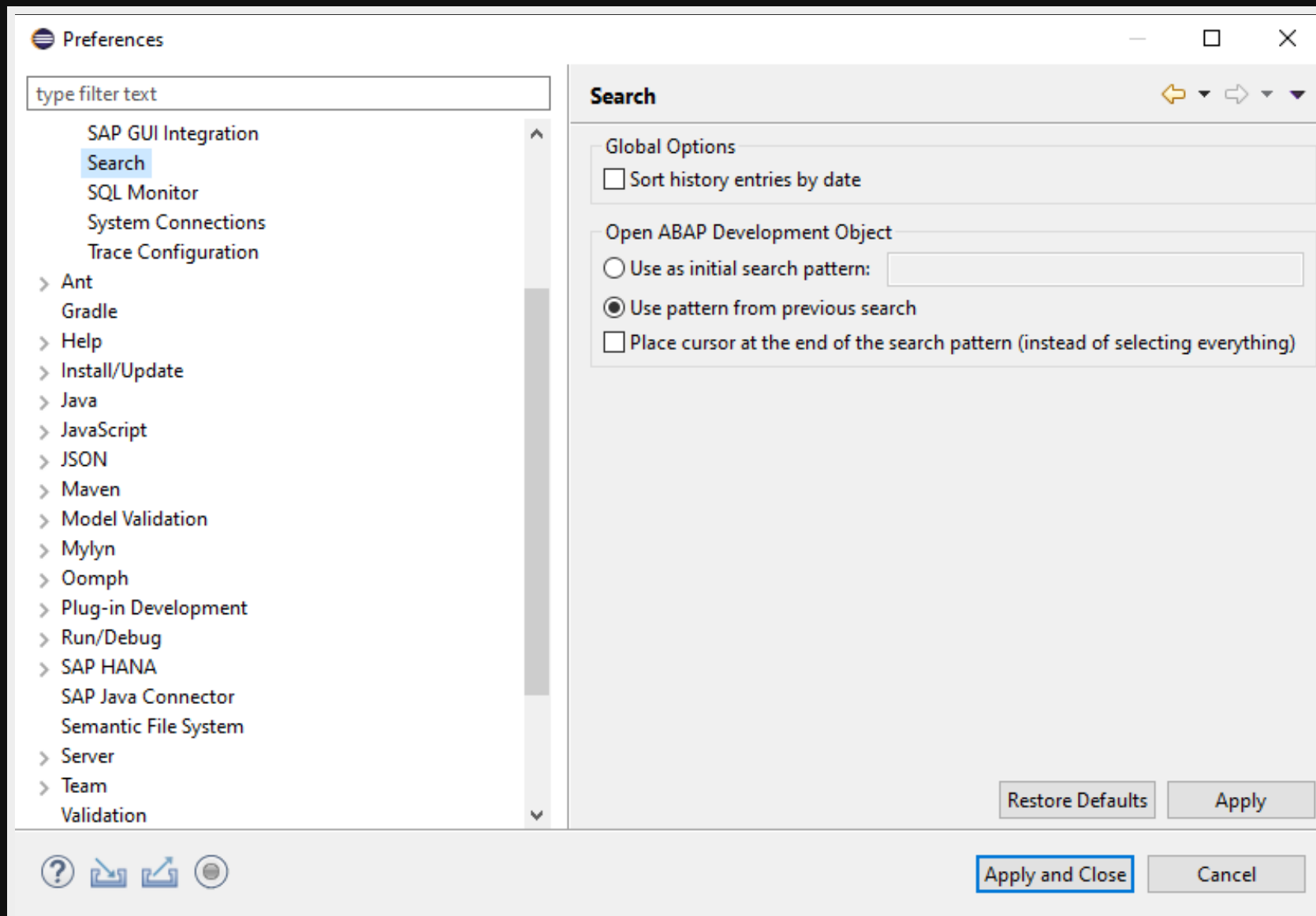
Code Completion



SAP GUI Integration

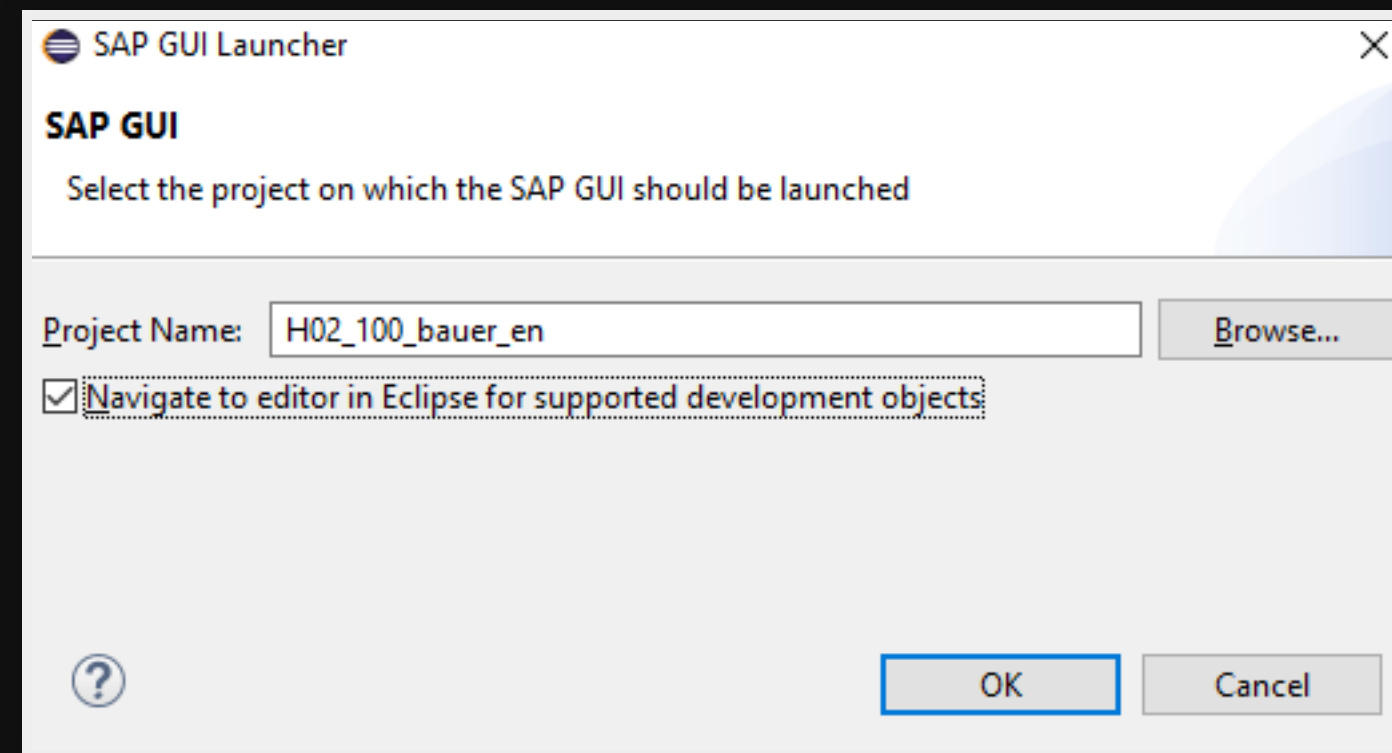


Search



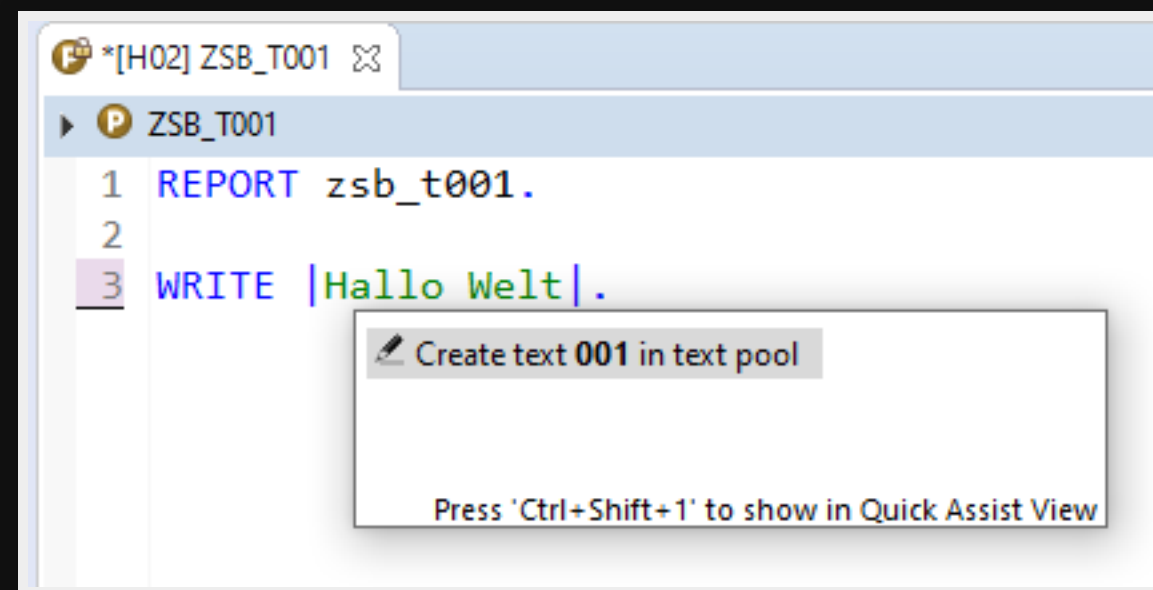
Navigate to Eclipse from SAP GUI

CTRL + 6

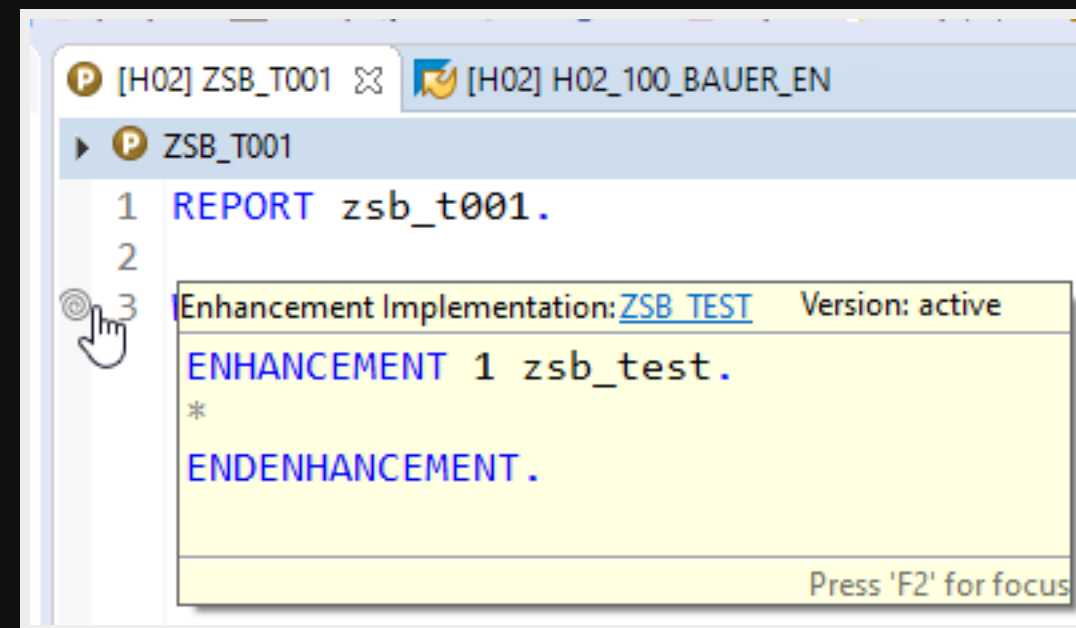


Text Pool

CTRL + 1

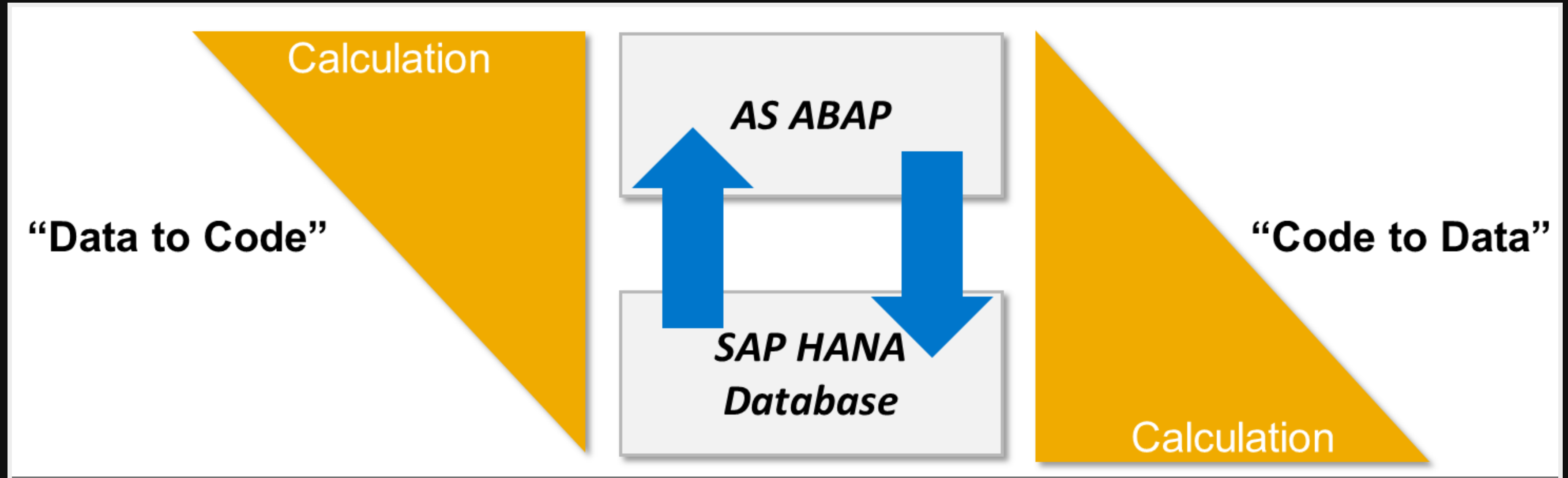


Enhancements



Outlook

Code Push Down



Code Push Down

- CDS Views
- ABAP Managed Database Procedures (AMDP)s

CDS Views

```
@AbapCatalog.sqlViewName: 'ZSB_V'  
@AbapCatalog.compiler.compareFilter: true  
@AccessControl.authorizationCheck: #CHECK  
define view ZSB_CDS  
  as select from  
  mara as material  
  association to makt as _material_text on  
    _material_text.matnr = material.matnr and  
    _material_text.spras = $session.system_language  
  
  {  
    material.matnr as material_number,  
    material.meins as unit_of_measure,  
  
    _material_text  
  }  
}
```

CDS Path Expressions

```
SELECT
    material_number,
    unit_of_measure,
    \_material_text-maktx
FROM
    zsb_cds
INTO TABLE
    @DATA(lt_).
```

AMDP

```
CLASS zsb_cl_amdp_test DEFINITION
  PUBLIC
  FINAL
  CREATE PUBLIC .

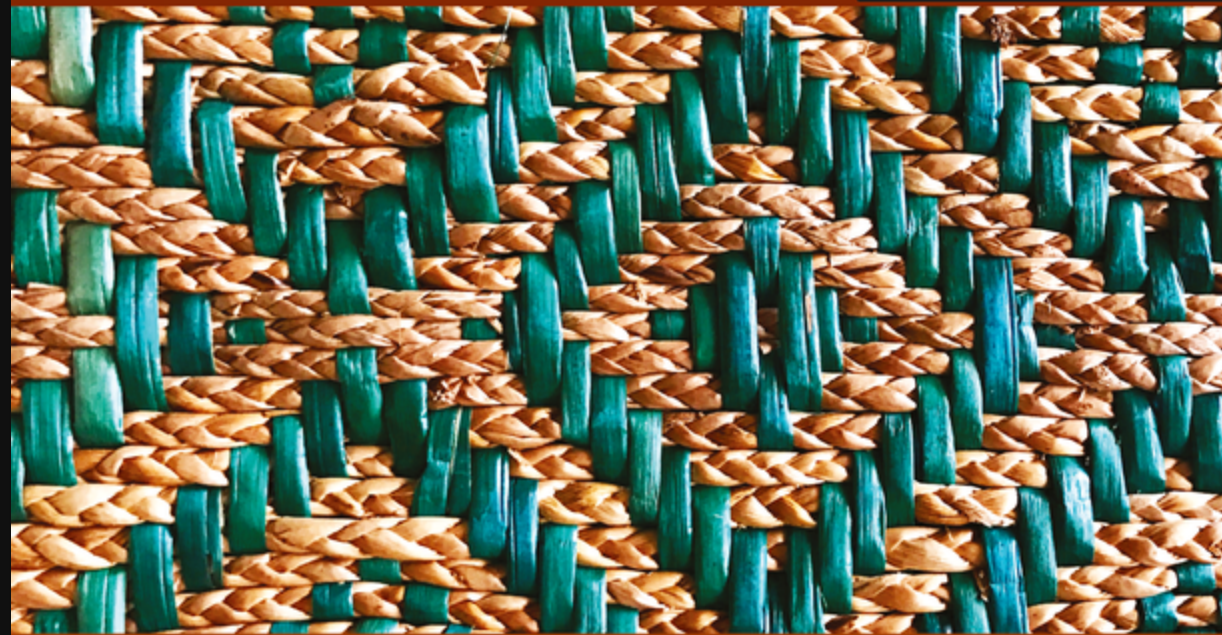
  PUBLIC SECTION.
    INTERFACES if_amdp_marker_hdb.

    TYPES: tt_mara TYPE STANDARD TABLE OF mara.

    CLASS-METHODS: read_product
      IMPORTING
        VALUE(iv_matnr)      TYPE mara-matnr
      EXPORTING
        VALUE(et_material) TYPE tt_mara
      DATETIME
```

RESTful ABAP Programming Model (RAP)

- Development of applications based on CDS views
- Successor of BOPF, but with less code generation
- Automatic creation of SAPUI5 application possible (at runtime)
- Importing for ABAP Cloud



ABAP® RESTful Programming Model

ABAP® Development for SAP S/4HANA®

- › Use the ABAP RESTful programming model for SAP S/4HANA
- › Develop applications with core data services, business objects, SAP Fiori, and more
- › Implement automated testing and configure continuous integration

Stefan Haas
Bince Mathew

