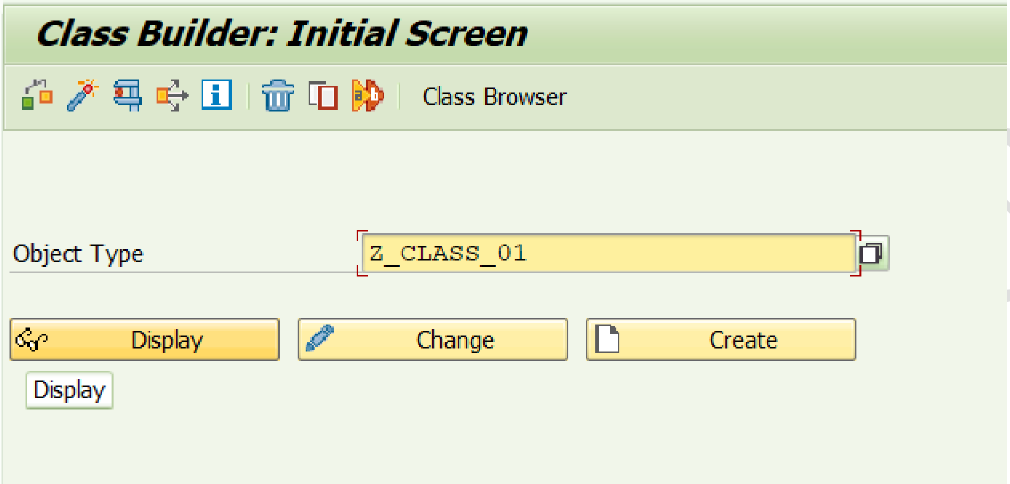
Requirement: To display the sales order details when a sales document number is inserted in the given field using a class.

Solution:

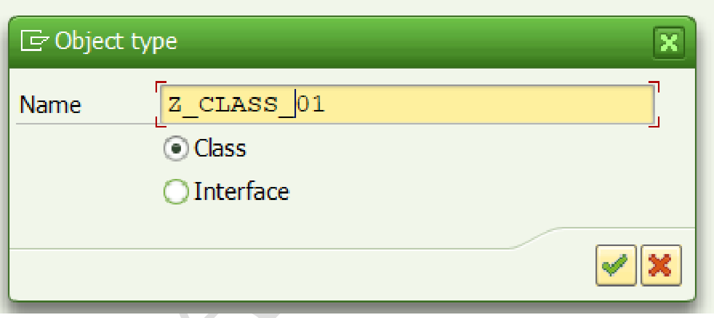
Step 1.1: Create a class through class builder t-code (SE24).



Class Builder: Initial Screen

Object Type Z_CLASS_01

Step 1.2: Choose class from the following.



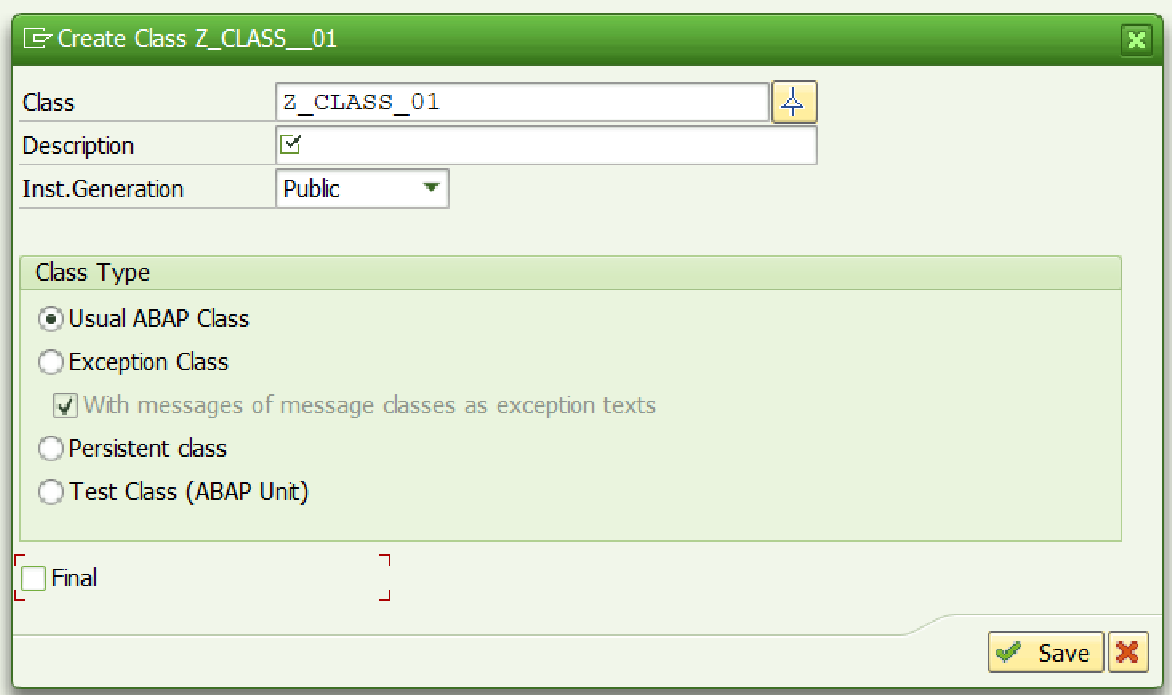
Object type

Name Z_CLASS_01

● Class

○ Interface

Step 1.3: Choose class types as Usual ABAP Class and uncheck final .



Create Class Z_CLASS__01

Class Z_CLASS_01

Description ☑

Inst.Generation Public

Class Type

● Usual ABAP Class

○ Exception Class

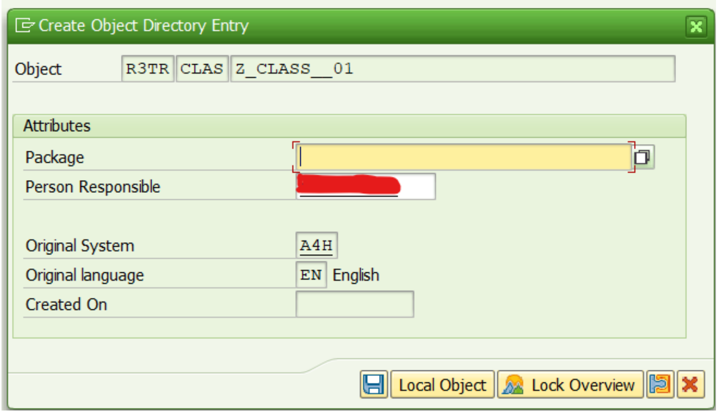
☑ With messages of message classes as exception texts

○ Persistent class

○ Test Class (ABAP Unit)

□ Final

Step 1.4: Save it as local.



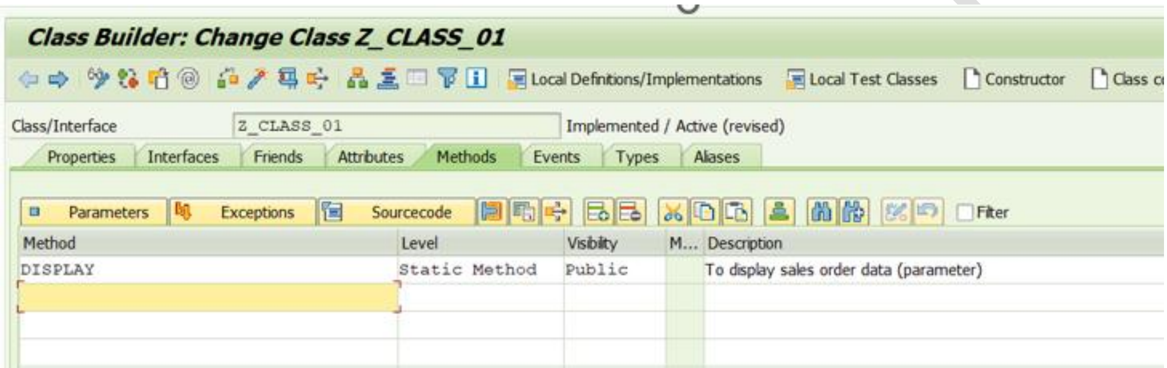
Create Object Directory Entry

Object: R3TR CLAS Z_CLASS__01

Attributes

Package	
Person Responsible	[redacted]
Original System	A4H
Original language	EN English
Created On	

Step 1.5: Create a method, choose level as static/instance, select visibility type and write a meaningful description.

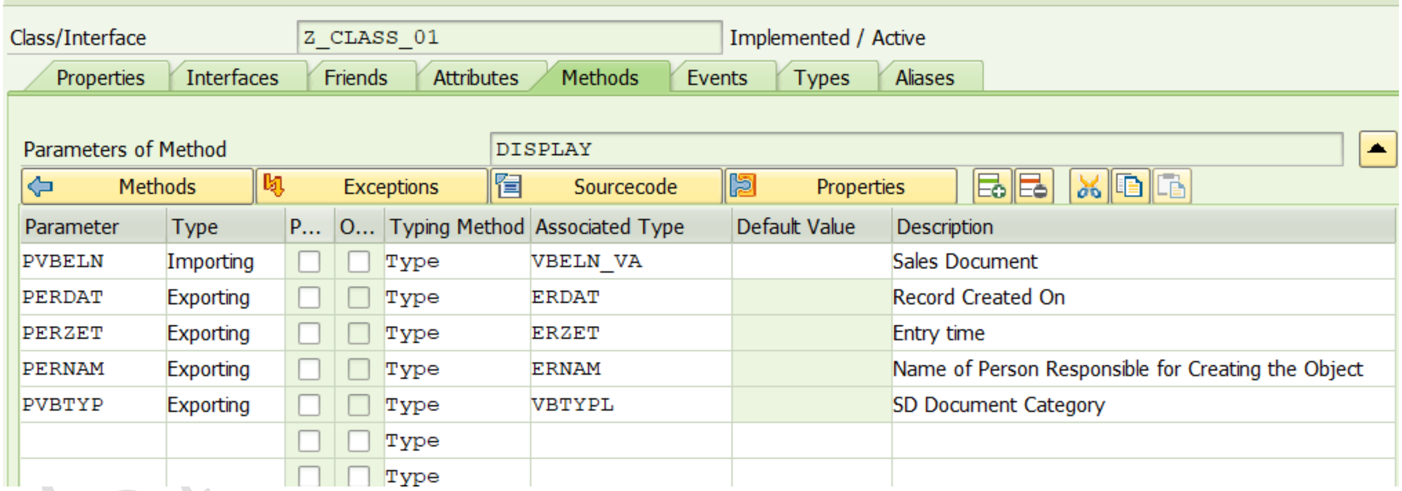


Class Builder: Change Class Z_CLASS_01

Class/Interface: Z_CLASS_01 Implemented / Active (revised)

Method	Level	Visibility	M...	Description
DISPLAY	Static Method	Public		To display sales order data (parameter)

Step 1.6: Select importing and exporting parameters.

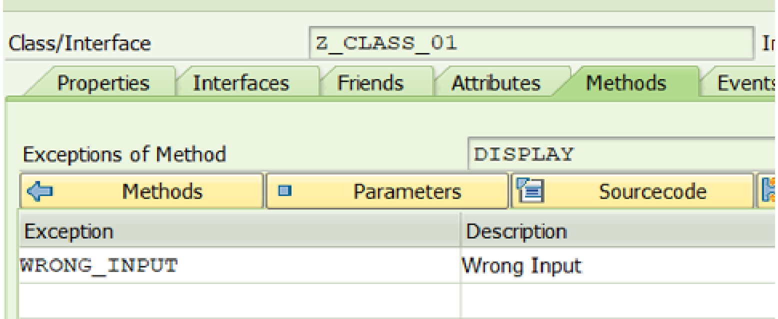


Class/Interface: Z_CLASS_01 Implemented / Active

Parameters of Method: DISPLAY

Parameter	Type	P...	O...	Typing Method	Associated Type	Default Value	Description
PVBELN	Importing	□	□	Type	VBELN_VA		Sales Document
PERDAT	Exporting	□	□	Type	ERDAT		Record Created On
PERZET	Exporting	□	□	Type	ERZET		Entry time
PERNAM	Exporting	□	□	Type	ERNAM		Name of Person Responsible for Creating the Object
PVBTYP	Exporting	□	□	Type	VBTYPL		SD Document Category
		□	□	Type			
		□	□	Type			

Step 1.7: Select exceptions button and write required number of exceptions.



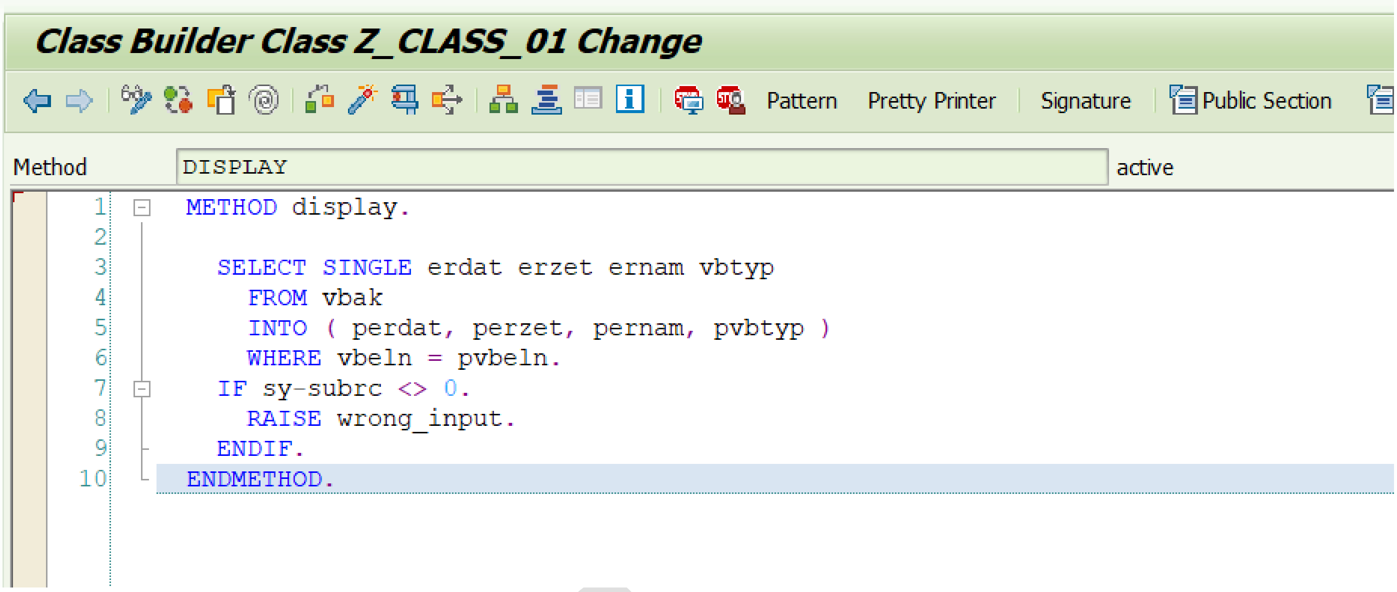
Class/Interface: Z_CLASS_01

Exceptions of Method: DISPLAY

Exception	Description
WRONG_INPUT	Wrong Input

Step 1.8: Write a source code to the method for class.

```
SELECT SINGLE erdat erzet ernam vbtyp
   FROM vbak
   INTO ( perdat, perzet, pernam, pvbtyp )
   WHERE vbeln = pvbeln.
  IF sy-subrc <> 0.
   RAISE wrong_input.
  ENDIF.
```



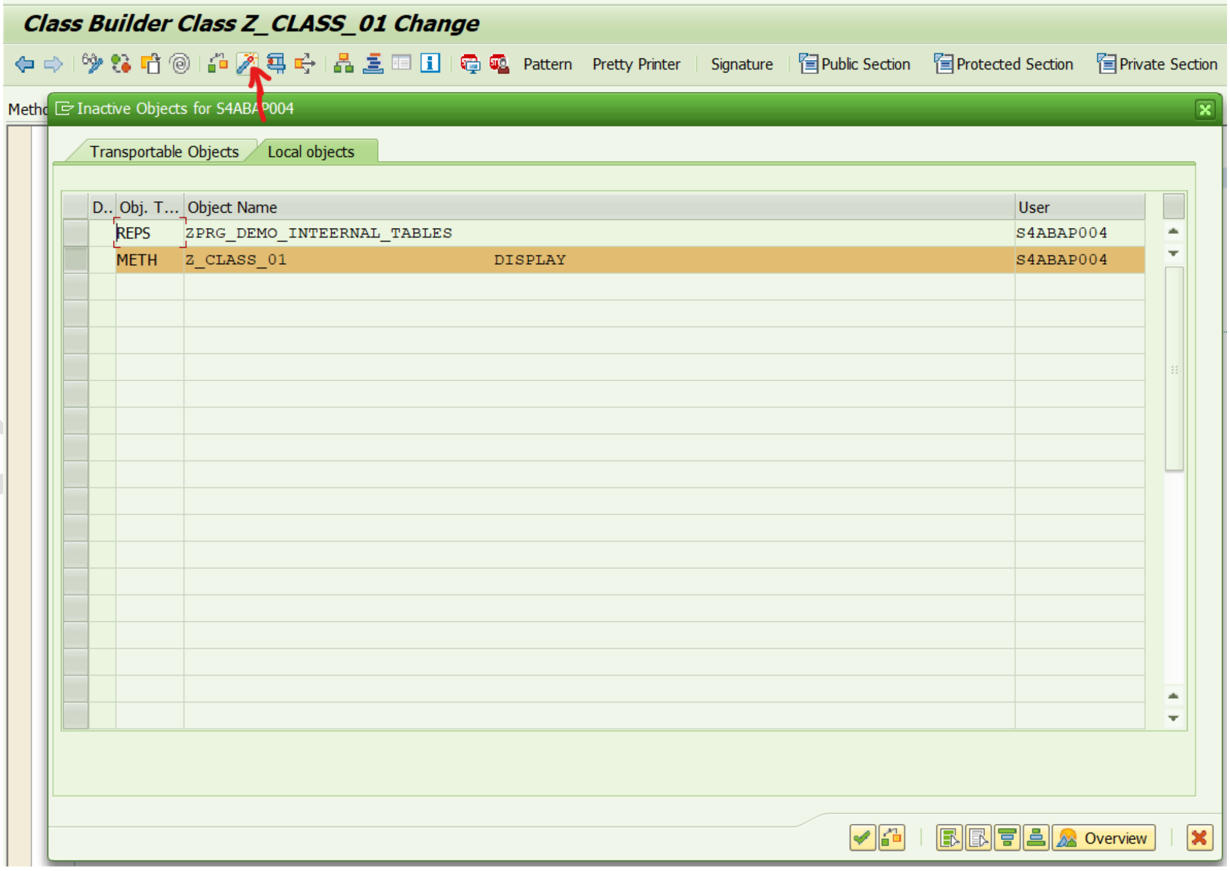
Class Builder Class Z_CLASS_01 Change

Method DISPLAY active

```
METHOD display.

  SELECT SINGLE erdat erzet ernam vbtyp
    FROM vbak
    INTO ( perdat, perzet, pernam, pvbtyp )
    WHERE vbeln = pvbeln.
  IF sy-subrc <> 0.
    RAISE wrong_input.
  ENDIF.
ENDMETHOD.
```

Step 1.9: Check if there is any error, then save and activate.

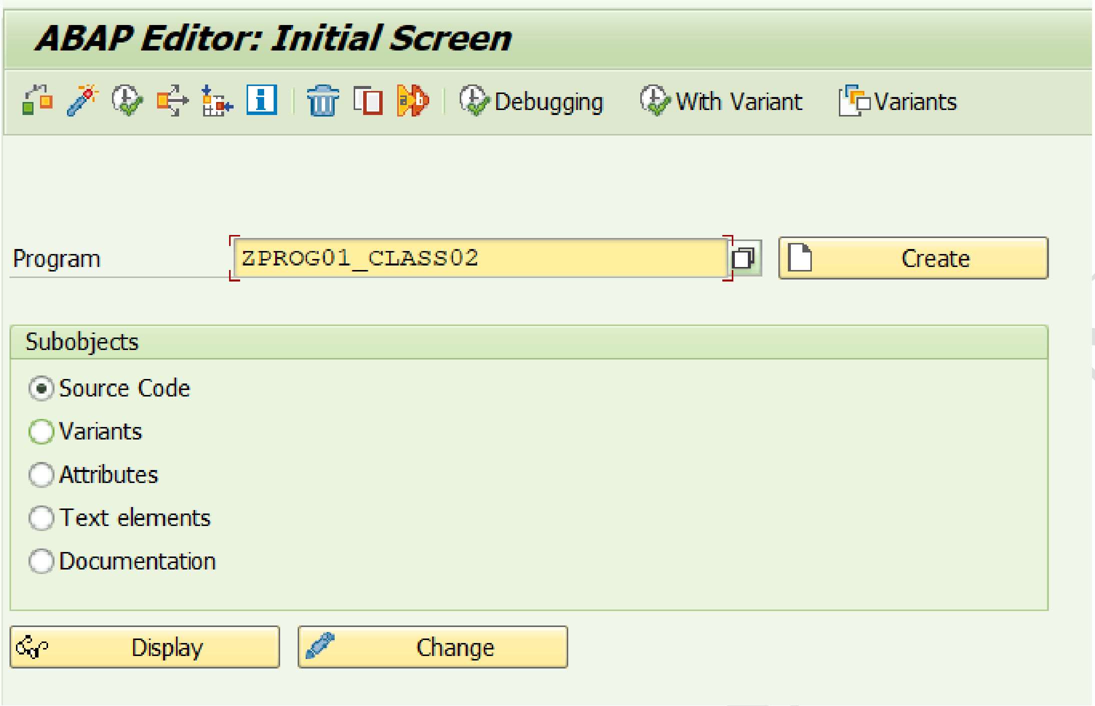


Class Builder Class Z_CLASS_01 Change

Inactive Objects for S4ABAP004

D..	Obj. T...	Object Name		User
	REPS	ZPRG_DEMO_INTEERNAL_TABLES		S4ABAP004
	METH	Z_CLASS_01	DISPLAY	S4ABAP004

Step 2.1: Create a Report through ABAP Editor T-code (SE38).



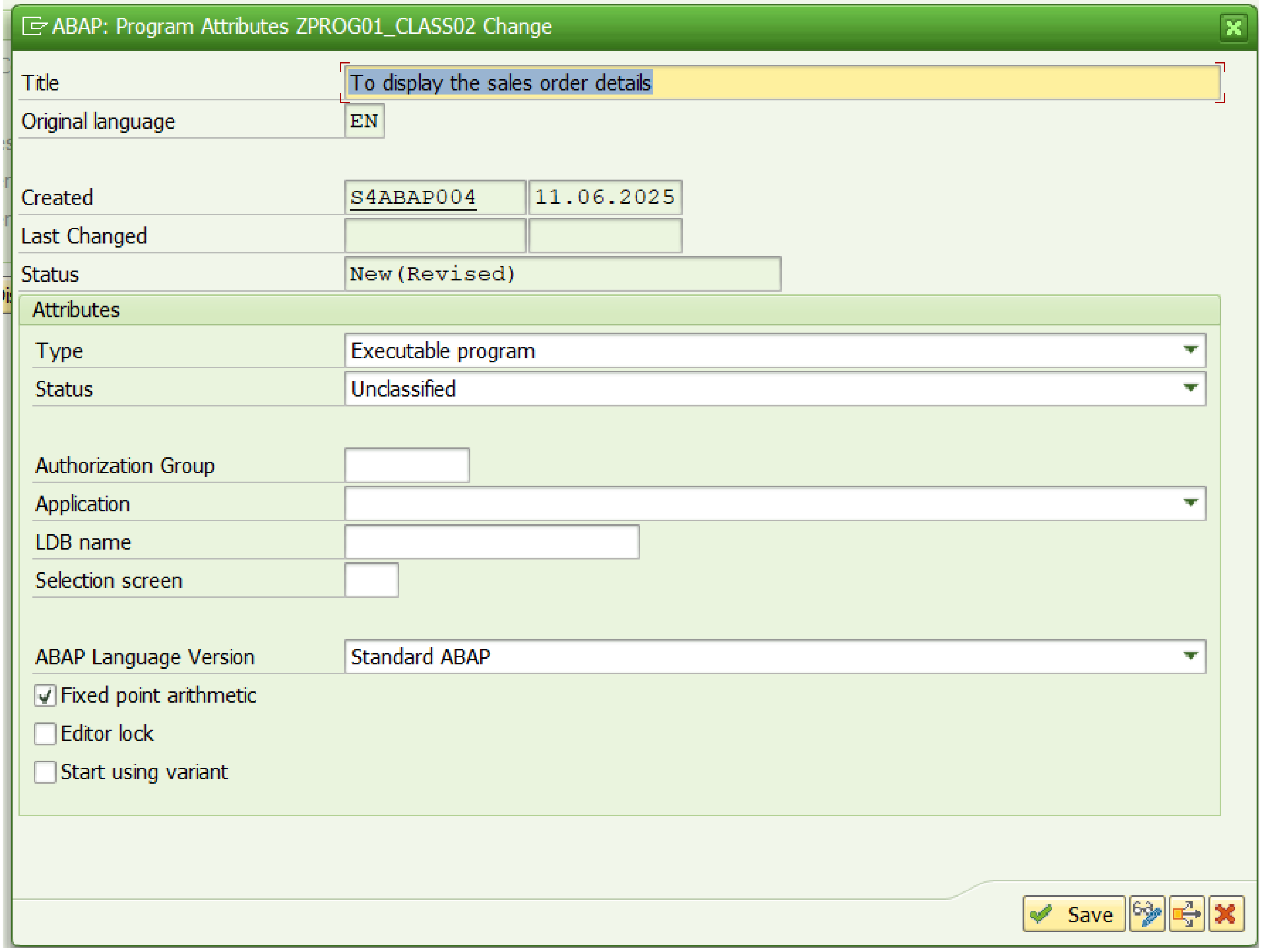
ABAP Editor: Initial Screen

Program: ZPROG01_CLASS02

Subobjects

- ● Source Code
- ○ Variants
- ○ Attributes
- ○ Text elements
- ○ Documentation

Step 2.2: Give a meaningful name and choose type as executable program. Select save button.



ABAP: Program Attributes ZPROG01_CLASS02 Change

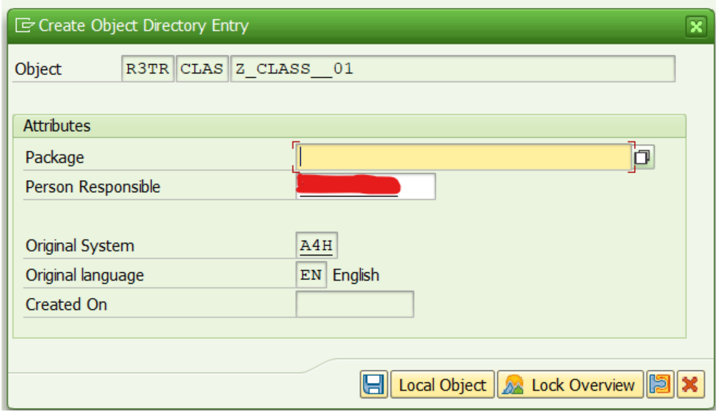
Title	To display the sales order details	
Original language	EN	
Created	S4ABAP004	11.06.2025
Last Changed		
Status	New(Revised)	

Attributes

Type	Executable program
Status	Unclassified
Authorization Group	
Application	
LDB name	
Selection screen	
ABAP Language Version	Standard ABAP

- ☑ Fixed point arithmetic
- □ Editor lock
- □ Start using variant

Step 2.3: Save it as local.



Create Object Directory Entry

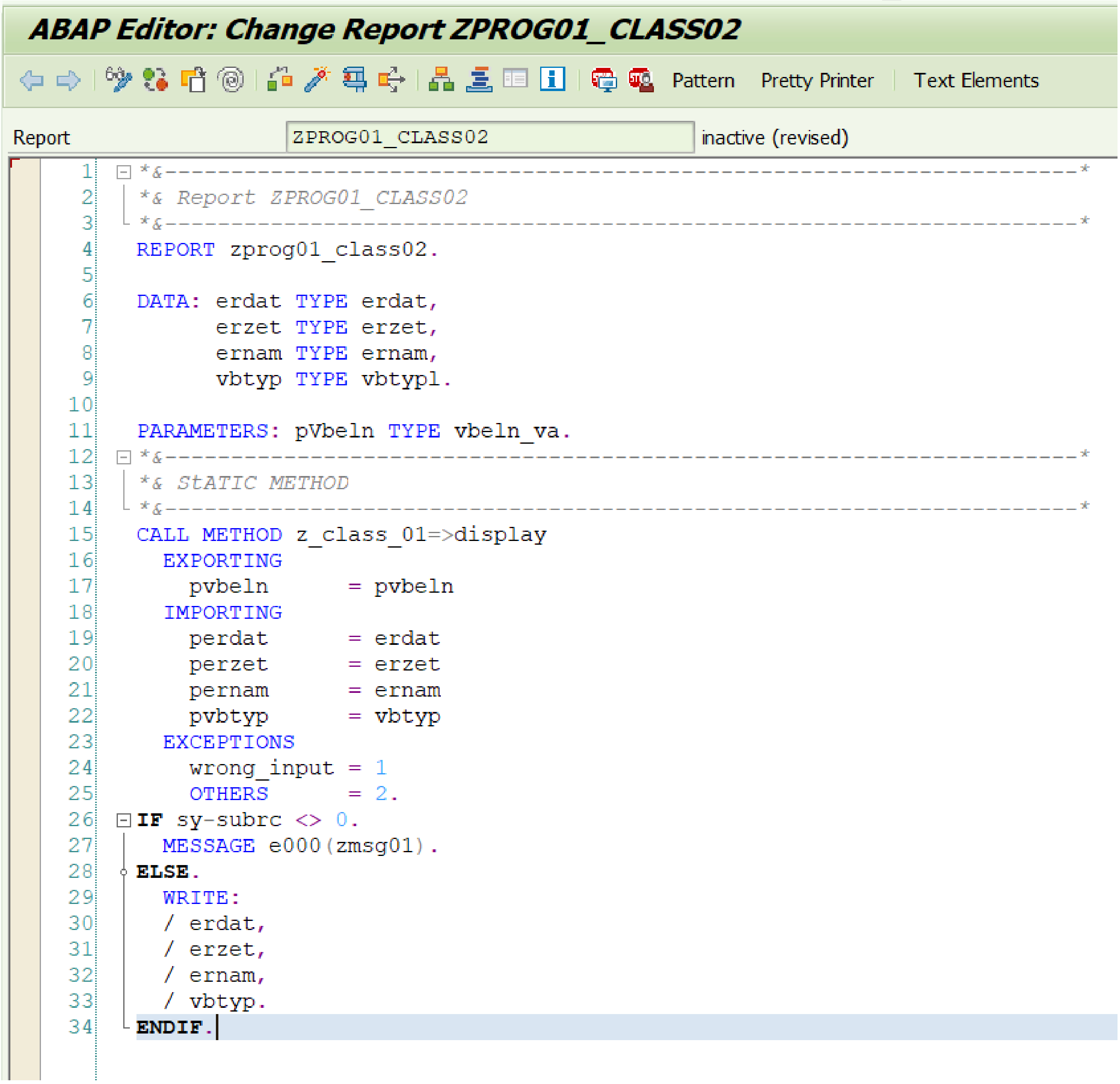
Object	R3TR	CLAS	Z_CLASS__01

Attributes

Package	
Person Responsible	[redacted]
Original System	A4H
Original language	EN English
Created On	

Step 2.4: Write a code to call the function to display the sales document number.

STATIC METHOD



ABAP Editor: Change Report ZPROG01_CLASS02

Report ZPROG01_CLASS02 inactive (revised)

```
*&---------------------------------------------------------------------*
*& Report ZPROG01_CLASS02
*&---------------------------------------------------------------------*
REPORT zprog01_class02.

DATA: erdat TYPE erdat,
      erzet TYPE erzet,
      ernam TYPE ernam,
      vbtyp TYPE vbtypl.

PARAMETERS: pVbeln TYPE vbeln_va.
*&---------------------------------------------------------------------*
*& StATIC METHOD
*&---------------------------------------------------------------------*
CALL METHOD z_class_01=>display
  EXPORTING
    pvbeln      = pvbeln
  IMPORTING
    perdat      = erdat
    perzet      = erzet
    pernam      = ernam
    pvbtyp      = vbtyp
  EXCEPTIONS
    wrong_input = 1
    OTHERS      = 2.
IF sy-subrc <> 0.
  MESSAGE e000(zmsg01).
ELSE.
  WRITE:
  / erdat,
  / erzet,
  / ernam,
  / vbtyp.
ENDIF.
```

```
REPORT zprog01_class02.

DATA: erdat TYPE erdat,
      erzet TYPE erzet,
      ernam TYPE ernam,
      vbtyp TYPE vbtypl.

PARAMETERS: pVbeln TYPE vbeln_va.
*&---------------------------------------------------------------------*
*& StATIC METHOD
*&---------------------------------------------------------------------*
CALL METHOD z_class_01=>display
  EXPORTING
    pvbeln      = pvbeln
  IMPORTING
    perdat      = erdat
    perzet      = erzet
    pernam      = ernam
    pvbtyp      = vbtyp
  EXCEPTIONS
    wrong_input = 1
    OTHERS      = 2.
IF sy-subrc <> 0.
  MESSAGE e000(zmsg01).
ELSE.
  WRITE:
  / erdat,
  / erzet,
  / ernam,
  / vbtyp.
ENDIF.
```

INSTANCE METHOD

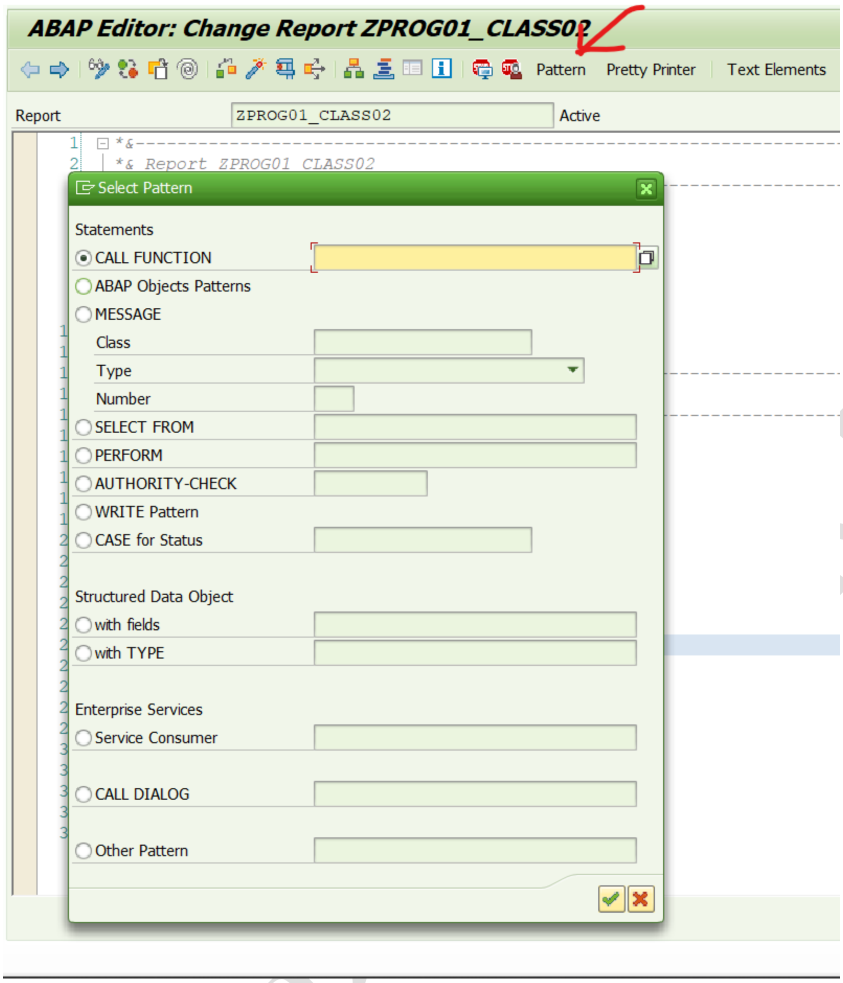
```
REPORT zprog01_class02.


DATA: erdat TYPE erdat,
      erzet TYPE erzet,
      ernam TYPE ernam,
      vbtyp TYPE vbtypl.

PARAMETERS: pVbeln TYPE vbeln_va.
*---------------------------------------------------------------
* Instance method
*---------------------------------------------------------------
*DATA: lo_object TYPE REF TO z_class_01.
*CREATE OBJECT lo_object.
*CALL METHOD lo_object->display1
*  EXPORTING
*    pvbeln      = pvbeln
*  IMPORTING
*    perdat      = erdat
*    perzet      = erzet
*    pernam      = ernam
*    pvbtyp      = vbtyp
*  EXCEPTIONS
*    wrong_input = 1
*    OTHERS      = 2.
*IF sy-subrc <> 0.
*  MESSAGE e000(zmsg01).
*  ELSE.
*    WRITE:
*    / erdat,
*    / erzet,
*    / ernam,
*    / vbtyp.
*ENDIF.
```

Step 2.5: Check if there is any error, then save and activate.

Note:- Call these class by pressing PATTERN button, click on the radio button ABAP OBJECT PATTERNS


ABAP Editor: Change Report ZPROG01_CLASS02

Report ZPROG01_CLASS02 Active

```
*&---------------------------------------------------------------
*& Report ZPROG01_CLASS02
```

Select Pattern

Statements

- ● CALL FUNCTION
- ○ ABAP Objects Patterns
- ○ MESSAGE
  - Class
  - Type
  - Number
- ○ SELECT FROM
- ○ PERFORM
- ○ AUTHORITY-CHECK
- ○ WRITE Pattern
- ○ CASE for Status

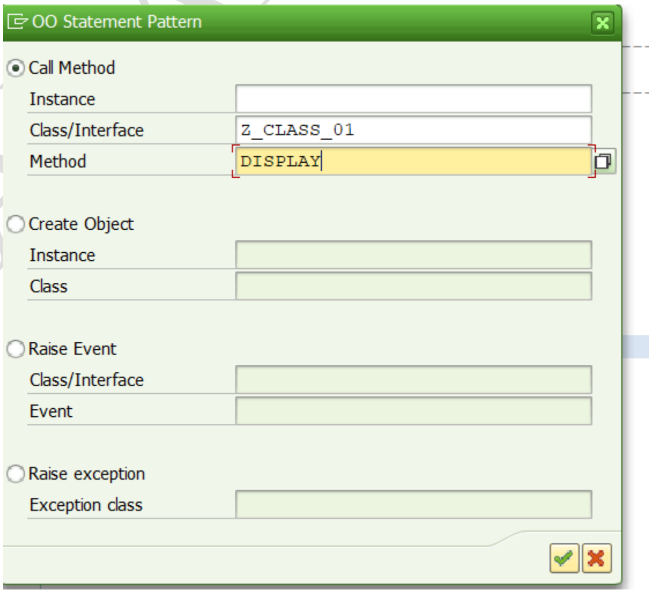
Structured Data Object

- ○ with fields
- ○ with TYPE

Enterprise Services

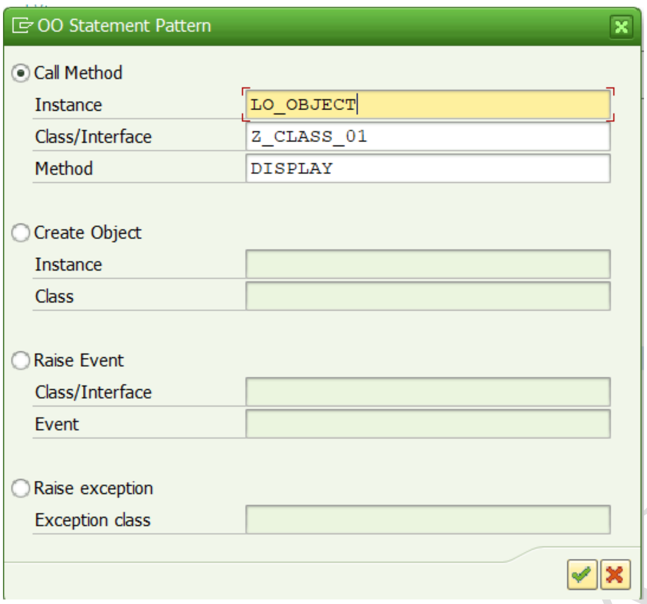
- ○ Service Consumer
- ○ CALL DIALOG
- ○ Other Pattern

FOR STATIC METHOD


OO Statement Pattern

- ● Call Method
  - Instance:
  - Class/Interface: Z_CLASS_01
  - Method: DISPLAY
- ○ Create Object
  - Instance
  - Class
- ○ Raise Event
  - Class/Interface
  - Event
- ○ Raise exception
  - Exception class

FOR INSTANCE METHOD



OO Statement Pattern

● Call Method

Instance: LO_OBJECT

Class/Interface: Z_CLASS_01

Method: DISPLAY

○ Create Object

Instance

Class

○ Raise Event

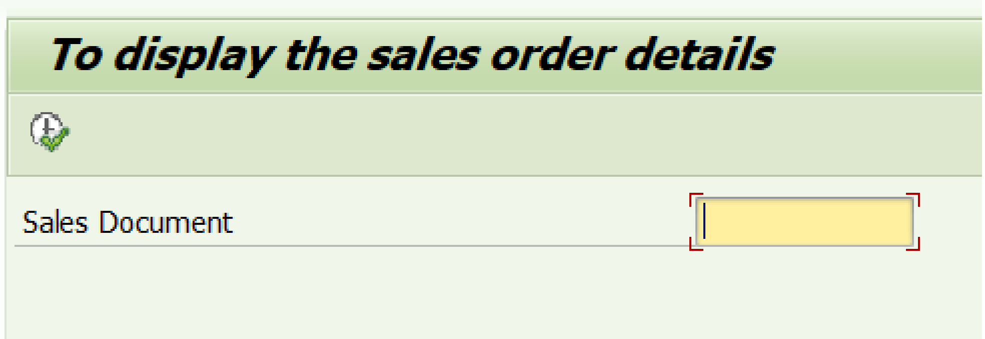
Class/Interface

Event

○ Raise exception

Exception class

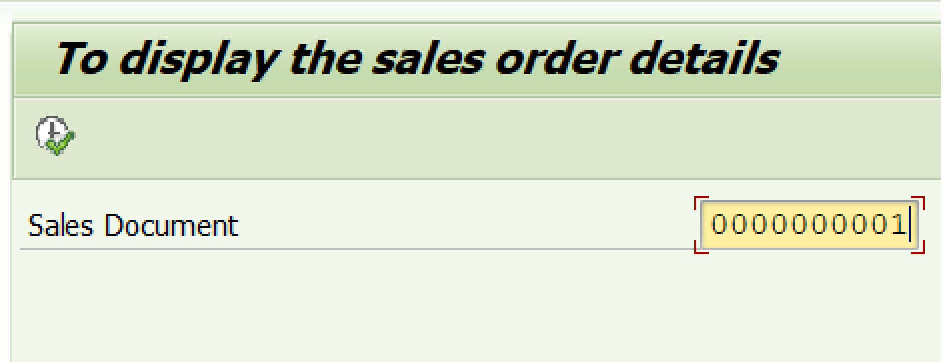
Step 2.6: Execute the code to check output.



To display the sales order details

Sales Document

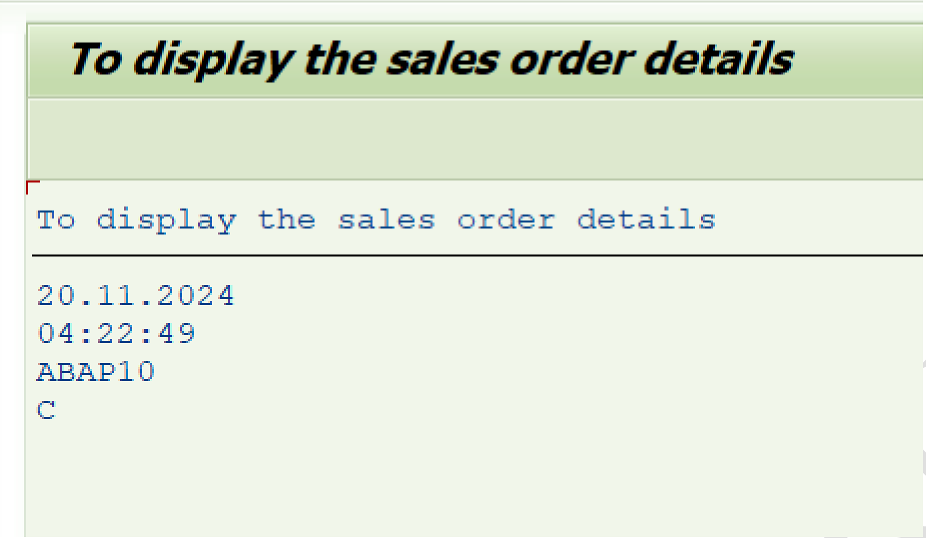
Inset Document Number



To display the sales order details

Sales Document: 0000000001

OUTPUT



To display the sales order details

```
To display the sales order details

20.11.2024
04:22:49
ABAP10
C
```