



# C\_ABAPD\_2309<sup>Q&As</sup>

SAP Certified Associate - Back-End Developer - ABAP Cloud

**Pass SAP C\_ABAPD\_2309 Exam with 100%  
Guarantee**

Free Download Real Questions & Answers **PDF** and **VCE** file from:

[https://www.pass4itsure.com/c\\_abapd\\_2309.html](https://www.pass4itsure.com/c_abapd_2309.html)

100% Passing Guarantee  
100% Money Back Assurance

Following Questions and Answers are all new published by SAP Official  
Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers



**QUESTION 1**

Given the following Core Data Service View Entity Data Definition:

1 @AccessControl.authorizationCheck: #NOT\_REQUIRED

2 DEFINE VIEW ENTITY demo\_flight\_info\_join

3AS SELECT

4FROM scarr AS a

5LEFT OUTER JOIN scounter AS c

6LEFT OUTER JOIN sairport AS p

7ON p.id = c.airport

8ON a.carrid = c.carrid

9{

10a.carridAS carrier\_id,

11p.idAS airport\_id,

12c.countnumAS counter\_number

13}

In what order will the join statements be executed?

- A. scarr will be joined with scounter first and the result will be joined with sairport.
- B. sairport will be joined to scounter first and the result will be joined with scarr.
- C. scarr will be joined with sairport first and the result will be joined with scounter.
- D. scounter will be joined to sairport first and the result will be joined with scarr.

Correct Answer: A

The order in which the join statements will be executed is: scarr will be joined with scounter first and the result will be joined with sairport. This is because the join statements are nested from left to right, meaning that the leftmost data source is joined with the next data source, and the result is joined with the next data source, and so on. The join condition for each pair of data sources is specified by the ON clause that follows the data source name. The join type for each pair of data sources is specified by the join operator that precedes the data source name. In this case, the join operator is LEFT OUTER JOIN, which means that all the rows from the left data source are included in the result, and only the matching rows from the right data source are included. If there is no matching row from the right data source, the corresponding fields are filled with initial values<sup>1</sup>. Therefore, the join statements will be executed as follows: First, scarr AS a will be joined with scounter AS c using the join condition a.carrid = c.carrid. This means that all the rows from scarr will be included in the result, and only the rows from scounter that have the same value for the carrid field will be included. If there is no matching row from scounter, the countnum field will be filled with an initial value. Second, the result of the first join will be joined with sairport AS p using the join condition p.id = c.airport. This means that all the rows from the first join will be included in the result, and only the rows from sairport that have the same value for the id



field as the airport field from the first join will be included. If there is no matching row from sairport, the id field will be filled with an initial value. References:1: Join - ABAP Keyword Documentation

---

## QUESTION 2

In an Access Control Object, which clauses are used? Note: There are 3 correct answers to this question.

- A. Where (to specify the access conditions)
- B. Crant (to identify the data source)
- C. Return code (to assign the return code of the authority check)
- D. Define role (to specify the role name)
- E. Revoke (to remove access to the data source)

Correct Answer: ADE

An Access Control Object (ACO) is a CDS annotation that defines the access control rules for a CDS view entity. An ACO consists of one or more clauses that specify the role name, the data source, the access conditions, and the return code of the authority check<sup>12</sup>. Some of the clauses that are used in an ACO are: Where (to specify the access conditions): This clause is used to define the logical expression that determines whether a user has access to the data source or not. The expression can use the fields of the data source, the parameters of the CDS view entity, or the predefined variables \$user and \$session. The expression can also use the functions check\_authorization and check\_role to perform additional authority checks<sup>12</sup>. Define role (to specify the role name): This clause is used to assign a name to the role that is defined by the ACO. The role name must be unique within the namespace of the CDS view entity and must not contain any special characters. The role name can be used to reference the ACO in other annotations, such as @AccessControl.authorizationCheck or @AccessControl.grant<sup>12</sup>. Revoke (to remove access to the data source): This clause is used to explicitly deny access to the data source for a user who meets the conditions of the where clause. The revoke clause overrides any grant clause that might grant access to the same user. The revoke clause can be used to implement the principle of least privilege or to enforce data segregation<sup>12</sup>. You cannot do any of the following: Grant (to identify the data source): This is not a valid clause in an ACO. The grant clause is a separate annotation that is used to grant access to a CDS view entity or a data source for a user who has a specific role. The grantclause can reference an ACO by its role name to apply the access conditions defined by the ACO<sup>12</sup>. Return code (to assign the return code of the authority check): This is not a valid clause in an ACO. The return code of the authority check is a predefined variable that is set by the system after performing the access control check. The return code can be used in the where clause of the ACO to specify different access conditions based on the outcome of the check<sup>12</sup>. References: 1: Access Control Objects - ABAP Keyword Documentation - SAP Online Help 2: Access Control in Core Data Services (CDS) | SAP Help Portal

---

## QUESTION 3

Class super has subclass sub. Which rules are valid for the sub constructor? Note: There are 2 correct answers to this question.

- A. The method signature can be changed.
- B. Import parameters can only be evaluated after calling the constructor of super.
- C. The constructor of super must be called before using any components of your own instance.
- D. Events of your own instance cannot be raised before the registration of a handler in super.



Correct Answer: AC

The sub constructor is the instance constructor of the subclass sub that inherits from the superclass super. The sub constructor has some rules that it must follow when it is defined and implemented<sup>12</sup>. Some of the valid rules are:

The method signature can be changed: This is true. The sub constructor can have a different method signature than the super constructor, which means that it can have different input parameters, output parameters, or exceptions. However,

the sub constructor must still call the super constructor with appropriate actual parameters that match its interface<sup>12</sup>.

The constructor of super must be called before using any components of your own instance: This is true. The sub constructor must ensure that the super constructor is called explicitly using super->constructor before accessing any instance

components of its own class, such as attributes or methods. This is because the super constructor initializes the inherited components of the subclass and sets the self-reference me-> to the current instance<sup>12</sup>.

You cannot do any of the following:

Import parameters can only be evaluated after calling the constructor of super:

This is false. The sub constructor can evaluate its own import parameters before calling the constructor of super, as long as it does not access any instance components of its own class. For example, the sub constructor can use its import

parameters to calculate some values or check some conditions that are needed for calling the super constructor<sup>12</sup>.

Events of your own instance cannot be raised before the registration of a handler in super: This is false. The sub constructor can raise events of its own instance before calling the constructor of super, as long as it does not access any

instance components of its own class. For example, the sub constructor can raise an event to notify the consumers of the subclass about some status or error that occurred during the initialization of the subclass<sup>12</sup>.

References: 1: Inheritance and Constructors - ABAP Keyword Documentation - SAP Online Help 2: Using Static and Instance constructor methods | SAP Blogs

---

#### QUESTION 4

In the assignment, data (gv\_result) = 1/8. What will be the data type of gv\_result?

- A. OTYPE I
- B. TYPE DEFLOAT 16
- C. TYPE P DECIMALS 3
- D. TYPE P DECIMALS 2

Correct Answer: B

The data type of gv\_result in the assignment data (gv\_result) = 1/8 will be TYPE DECFLOAT 16. This is because the assignment operator (=) in ABAP performs an implicit type conversion from the source type to the target type, according to



the following rules12:

If the target type is specified explicitly, the source value is converted to the target type.

If the target type is not specified explicitly, the source type is used as the target type, unless the source type is a literal or an expression, in which case the target type is determined by the following priority order: DECFLOAT34, DECFLOAT16,

P, F, I, C, N, X, STRING, XSTRING.

In this case, the target type is not specified explicitly, and the source type is an expression (1/8). Therefore, the target type is determined by the priority order, and the first matching type is DECFLOAT16, which is a decimal floating point type

with 16 digits of precision12. References: 1: ABAP Assignment Rules - ABAP Keyword Documentation - SAP Online Help 2: ABAP Data Types - ABAP Keyword Documentation - SAP Online Help

## QUESTION 5

Which field is defined incorrectly?

```
1 DEFINE TABLE demo_table
2 |
3 KEY field1 : NUMERIC(30) abap-int(12);
4 KEY field2 : abap-char(12);
5 @quantity:quantity:unit@Resource : 'demo_table:field4'
6 field3 : abap-quant(2);
7 field4 : abap-int(12);
8 |
```

(Sorry, we do not have a more clear image. If we have a better resource for the image, we will update this one immediately.)

- A. field1
- B. field3
- C. field4
- D. field2

Correct Answer: C

The field4 is defined incorrectly in the ABAP code snippet. The reason is that the data type c (character) cannot have a decimal places specification. The decimal places specification is only valid for the data types p (packed number) and f



(floating point number)1. Therefore, the field4 definition should either omit the decimal places specification or change the data type to p or f. References: 1: Data Types and Data Objects - ABAP Keyword Documentation - SAP Online Help

[Latest C\\_ABAPD\\_2309 Dumps](#)

[C\\_ABAPD\\_2309 Practice Test](#)

[C\\_ABAPD\\_2309 Study Guide](#)