



**Free Questions for C\_ABAPD\_2309 by [braindumpscollection](#)**

**Shared by [Castillo](#) on [15-04-2024](#)**

**For More Free Questions and Preparation Resources**

**[Check the Links on Last Page](#)**

# Question 1

---

**Question Type:** MultipleChoice

---

Refer to the Exhibit.

Given the following Core Data Services View Entity Data Definition:

```
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 DEFINE VIEW ENTITY demo_cds_assoc_element
3   AS SELECT FROM scarr
4   ASSOCIATION OF ONE TO MANY demo_cds_assoc_spfli AS _spfli
5   ON scarr.carrid = _spfli.carrid
6   {
7     KEY carrid,
8   ?
9     carrname
10  }
```

The "demo\_ods\_assoc\_spfli" data source referenced in line #4 contains a field "connid" which you would like to expose in the element list.

Which of the following statements would do this if inserted on line #8?

## Options:

---

- A- demo\_ods\_assoc\_spfli.connid,
- B- demo\_ods\_assoc\_spfli-connid/
- C- spfli-connid,
- D- \_spfli.connid/

## Answer:

---

A

## Explanation:

---

The statement that can be used to expose the field "connid" of the data source "demo\_ods\_assoc\_spfli" in the element list is A. demo\_ods\_assoc\_spfli.connid,. This statement uses the dot notation to access the field "connid" of the data source "demo\_ods\_assoc\_spfli", which is an association defined on line #4. The association "demo\_ods\_assoc\_spfli" links the data source "demo\_ods" with the table "spfli" using the field "carrid".The statement also ends with a comma to separate it from the next element in the list12.

You cannot do any of the following:

B) demo\_ods\_assoc\_spfli-connid/: This statement uses the wrong syntax to access the field "connid" of the data source "demo\_ods\_assoc\_spfli". The dash notation is used to access the components of a structure or a table, not the fields of a data source.The statement also ends with a slash, which is not a valid separator for the element list12.

C) spfli-connid,: This statement uses the wrong data source name to access the field "connid". The data source name should be "demo\_ods\_assoc\_spfli", not "spfli".The statement also uses the wrong syntax to access the field "connid", as explained above12.

D) \_spfli.connid/: This statement uses the wrong data source name and the wrong separator to access the field "connid". The data source name should be "demo\_ods\_assoc\_spfli", not "\_spfli".The statement also ends with a slash, which is not a valid separator for the element list12.

## Question 2

---

### Question Type: MultipleChoice

---

As a consultant you are posed the following question from a client who is using SAP S/4HANA Cloud, public edition and also SAP BTP, ABAP environment.

"We are currently using an SAP Fiori app based on SAP Fiori elements that analyzes open orders. We have determined that it should be extended via a new button on the UI which will perform an on-the-fly calculation and display the result in a quick popup for the enduser. We have been informed by SAP that all underlying stack layers for the SAP Fiori app have been extensibility enabled."

Based on this which of the following extension types would you recommend to the customer to add

the new button?

**Options:**

---

**A-** RAP BO Behavior Extension

**B-** SAP HANA database table extension

**C-** RAP BO Node Extension

**D-** Business Service Extension

**Answer:**

---

C

## Question 3

---

**Question Type: MultipleChoice**

---

Which statement can you use to change the contents of a row of data in an internal table?

### Options:

---

- A- Append table
- B- Modify table
- C- Insert table
- D- Update table

### Answer:

---

B

### Explanation:

---

The statement that can be used to change the contents of a row of data in an internal table is MODIFY table. The MODIFY table statement can be used to change the contents of one or more rows of an internal table, either by specifying the table index, the table key, or a condition. The MODIFY table statement can also be used to change the contents of a database table, by specifying the table name and a work area or an internal table. The MODIFY table statement can use the TRANSPORTING addition to specify which fields should be changed, and the WHERE addition to specify which rows should be changed.

The other statements are not suitable for changing the contents of a row of data in an internal table, as they have different purposes and effects. These statements are:

APPEND table: This statement can be used to add a new row of data to the end of an internal table, either by specifying a work area or an inline declaration. The APPEND table statement does not change the existing rows of the internal table, but only increases the

number of rows by one.

**INSERT table:** This statement can be used to insert a new row of data into an internal table, either by specifying the table index, the table key, or a sorted position. The INSERT table statement does not change the existing rows of the internal table, but only shifts them to make room for the new row. The INSERT table statement can also be used to insert a new row of data into a database table, by specifying the table name and a work area or an inline declaration.

**UPDATE table:** This statement can be used to update the contents of a database table, by specifying the table name and a work area or an internal table. The UPDATE table statement can use the SET addition to specify which fields should be updated, and the WHERE addition to specify which rows should be updated. The UPDATE table statement does not affect the internal table, but only the corresponding database table.

## Question 4

---

**Question Type:** MultipleChoice

---

You want to provide a short description of the data definition for developers that will be attached to the database view

```
1 @HeaderItem() headerItem() {
2     //
3     @DrawableRes() drawable() {
4         @Color() color() {
5             @Color() color() {
6                 @Color() color() {
7                     @Color() color() {
8                         @Color() color() {
9                             @Color() color()
10                    }
11                }
12            }
13        }
14    }
15 }
16 }
17 }
18 }
19 }
20 }
21 }
22 }
23 }
24 }
25 }
26 }
27 }
28 }
29 }
30 }
```

Which of the following annotations would do this if you inserted it on line #27

**Options:**

---

- A- @UI headerinto description label
- B- @UI.badge.title.label
- C- @EndUserText.quickInfo
- D- @EndUserText label

**Answer:**

---

D



## Explanation:

---

The annotation that can be used to provide a short description of the data definition for developers that will be attached to the database view is the `@EndUserText.label` annotation. This annotation is used to specify a text label for the data definition that can be displayed in the development tools or in the documentation. The annotation can be inserted on line #27 in the code snippet provided in the question12. For example:

The following code snippet uses the `@EndUserText.label` annotation to provide a short description of the data definition for the CDS view `ZCDS_VIEW`:

```
@AbapCatalog.sqlViewName: 'ZCDS_VIEW' @AbapCatalog.compiler.compareFilter: true @AbapCatalog.preserveKey: true
@AccessControl.authorizationCheck: #CHECK @EndUserText.label: 'CDS view for flight data' 'short description for developers define
view ZCDS_VIEW as select from sflight { key carrid, key connid, key fldate, seatsmax, seatsocc }
```

You cannot do any of the following:

`@UI.headerInfo.description.label`: This annotation is used to specify a text label for the description field of the header information of a UI element. This annotation is not relevant for the data definition of a database view12.

`@UI.badge.title.label`: This annotation is used to specify a text label for the title field of a badge UI element. This annotation is not relevant for the data definition of a database view12.

`@EndUserText.quickInfo`: This annotation is used to specify a quick information text for the data definition that can be displayed as a tooltip in the development tools or in the documentation. This annotation is not the same as a short description or a label for the data definition12.

## Question 5

---

**Question Type:** MultipleChoice

---

Which of the following are valid sort operations for internal tables? Note: There are 3 correct answers to this question.

### Options:

---

**A-** Sort a standard table using  
SORT itab ASCENDING.  
Sort a sorted table using

**B-** SORT itab BY field1 ASCENDING field2 DESCENDING.  
Sort a standard table using

**C-** SORT itab BY field1 field2.  
Sort a standard table using

**D-** SORT itab.  
Sort a sorted table using  
E.  
SORT itab DESCENDING.

### Answer:

---

A, C, D

# Question 6

---

Question Type: MultipleChoice

---

## Exhibit

```
1 @AccessControl.authorizationcheck: NOT_REQUIRED
2 DEFINE VIEW ZVTW_ZVTW7777_ZWZ_PARM_VTM_VTM_HY
3 SELECT FROM ZVTW7777_ZWZ_PARM_VTM_VTM_HY
4 p_date =>@p_date
5 TO SELECT FROM
6 @flight
7 {
8 KEY coveid,
9 KEY coveid,
10 KEY flight,
11 prior,
12 seatname,
13 seatname
14 }
15 @HERE #date => @parameters.p_date;
```

Which of the following ABAP SQL snippets are syntactically correct ways to provide a value for the parameter on line #4? Note: There are 2 correct answers to this question

Options:

---

**A-** ...SELECT \* FROM deno\_cds\_param\_view\_entity (p\_date = @ (cl\_abap\_context\_info->get\_system\_date ()))...

**B-** ...SELECT \* FROM deno\_cds\_param\_view\_entity (p\_date - '20230101')... )

**C-** ...SELECT \* FROM demo\_cds\_param\_view\_entity (p\_date: 20238181')... )

**D-** ...SELECT \* FROM demo\_cds\_param\_view entity (p\_date: \$session.system\_date)...

**Answer:**

---

A, B

## Question 7

---

**Question Type:** MultipleChoice

---

/DMO/I\_Connection is a CDS view.

What variable type is connection full based on the following code? DATA connection full TYPE

/DMD/I\_Connection.

**Options:**

---

- A- Simple variable
- B- Structure
- C- Internal Table

**Answer:**

---

B

**Explanation:**

---

Based on the following code, the variable type of connection\_full is a structure. A structure is a complex data type that consists of a group of related data objects, called components, that have their own data types and names. A structure can be defined using the TYPES statement or based on an existing structure type, such as a CDS view entity or a CDS DDIC-based view. In this case, the variable connection\_full is declared using the TYPE addition, which means that it has the same structure type as the CDS view entity /DMO/I\_Connection. The CDS view entity /DMO/I\_Connection is a data model view that defines a data model based on the database table /DMO/Connection. The CDS view entity /DMO/I\_Connection has the following components: carrid, connid, airpfrom, airpto, distance, and fltime. Therefore, the variable connection\_full has the same components as the CDS view entity /DMO/I\_Connection, and each component has the same data type and length as the corresponding field in the database table /DMO/Connection.

## Question 8

---

**Question Type: MultipleChoice**

---

In ABAP SQL, which of the following can be assigned an alias? Note: There are 2 correct answers to this question.

### Options:

---

- A- order criterion (from order by clause)
- B- field (from field list)
- C- database table
- D- group criterion (from group by clause)

### Answer:

---

B, C

### Explanation:

---

In ABAP SQL, an alias is a temporary name that can be assigned to a field or a database table in a query. An alias can be used to make the query more readable, to avoid name conflicts, or to access fields or tables with long names. An alias is created with the AS keyword and is only valid for the duration of the query<sup>1</sup>.

The following are examples of how to assign an alias to a field or a database table in ABAP SQL:

B) field (from field list): A field is a column of a table or a view that contains data of a certain type. A field can be assigned an alias in the field list of a SELECT statement, which specifies the fields that are selected from the data source. For example, the following query assigns the alias name to the field carrname of the table scarr:

```
SELECT carrid, carrname AS name FROM scarr.
```

The alias name can be used instead of carrname in other clauses of the query, such as WHERE, GROUP BY, ORDER BY, and so on<sup>2</sup>.

C) database table: A database table is a collection of data that is organized in rows and columns. A database table can be assigned an alias in the FROM clause of a SELECT statement, which specifies the data source that is selected from. For example, the following query assigns the alias c to the table scarr:

```
SELECT c.carrid, c.carrname FROM scarr AS c.
```

The alias c can be used instead of scarr in other clauses of the query, such as WHERE, JOIN, GROUP BY, ORDER BY, and so on<sup>3</sup>.

The following are not valid for assigning an alias in ABAP SQL:

A) order criterion (from order by clause): An order criterion is a field or an expression that is used to sort the result set of a query in ascending or descending order. An order criterion cannot be assigned an alias in the ORDER BY clause of a SELECT statement, because the alias is not visible in this clause. The alias can only be used in the clauses that follow the clause where it is defined<sup>1</sup>.

D) group criterion (from group by clause): A group criterion is a field or an expression that is used to group the result set of a query into subsets that share the same values. A group criterion cannot be assigned an alias in the GROUP BY clause of a SELECT statement, because the alias is not visible in this clause. The alias can only be used in the clauses that follow the clause where it is defined<sup>1</sup>.

## Question 9

---

Question Type: MultipleChoice

---

Refer to the Exhibit.

```
1  Given this code,  
2  DATA: go_super TYPE REF TO lcl_super,  
3         go_sub1  TYPE REF TO lcl_sub1.  
4  ...  
5  go_sub1 = CAST #( go_super ).  
6  go_sub1->sub1_meth1(...).  
7
```

with which predicate condition can you ensure that the CAST will work?

### Options:

---

- A- IS SUPPLIED
- B- IS NOT INITIAL
- C- IS INSTANCE OF
- D- IS BOUND

### Answer:

---



C

## **Explanation:**

---

The predicate condition that can be used to ensure that the CAST will work is IS INSTANCE OF. The IS INSTANCE OF predicate condition checks whether the operand is an instance of the specified class or interface. This is useful when you want to perform a downcast, which is a conversion from a more general type to a more specific type. A downcast can fail if the operand is not an instance of the target type, and this can cause a runtime error. Therefore, you can use the IS INSTANCE OF predicate condition to check whether the downcast is possible before using the CAST operator<sup>12</sup>. For example:

The following code snippet uses the IS INSTANCE OF predicate condition to check whether the variable g\_super is an instance of the class lcl\_super. If it is, the CAST will work and the variable g\_sub1 will be assigned the value of g\_super.

```
DATA: g_super TYPE REF TO lcl_super, g_sub1 TYPE REF TO lcl_sub1. IF g_super IS INSTANCE OF lcl_super. g_sub1 = CAST #(
g_super ). g_sub1->method( ... ). ENDIF.
```

You cannot do any of the following:

**IS SUPPLIED:** The IS SUPPLIED predicate condition checks whether an optional parameter of a method or a function module has been supplied by the caller. This is useful when you want to handle different cases depending on whether the parameter has a value or not. However, this predicate condition has nothing to do with the CAST operator or the type of the operand<sup>12</sup>.

**IS NOT INITIAL:** The IS NOT INITIAL predicate condition checks whether the operand has a non-initial value. This is useful when you want to check whether the operand has been assigned a value or not. However, this predicate condition does not guarantee that the CAST will work, because the operand may have a value but not be an instance of the target type<sup>12</sup>.



## Options:

---

- A- SELECT mat FROM Material...
- B- SELECT mat FROM demo\_sales\_cds\_so\_i\_ve...
- C- SELECT mat FROM demo\_sales\_so\_i...
- D- SELECT mat FROM demo sales cds material ve...

## Answer:

---

B

## Explanation:

---

Using ABAP SQL, the select statement that selects the mat field on line #17 is:

```
SELECT mat FROM demo_sales_cds_so_i_ve...
```

This statement selects the mat field from the CDS view demo\_sales\_cds\_so\_i\_ve, which is defined on line #1. The CDS view demo\_sales\_cds\_so\_i\_ve is a projection view that projects the fields of the CDS view demo\_sales\_cds\_so\_i, which is defined on line #2. The CDS view demo\_sales\_cds\_so\_i is a join view that joins the fields of the database table demo\_sales\_so\_i, which is defined on line #3, and the CDS view demo\_sales\_cds\_material\_ve, which is defined on line #4. The CDS view demo\_sales\_cds\_material\_ve is a value help view that provides value help for the material field of the database table demo\_sales\_so\_i. The mat field is an alias for the material field of the database table demo\_sales\_so\_i, which is defined on line #91.

The other options are not valid because:

A) `SELECT mat FROM Material...` is not valid because `Material` is not a valid data source in the given code. There is no CDS view or database table named `Material`.

C) `SELECT mat FROM demo_sales_so_i...` is not valid because `demo_sales_so_i` is not a valid data source in the given code. There is no CDS view named `demo_sales_so_i`, only a database table. To access a database table, the keyword `TABLE` must be used, such as `SELECT mat FROM TABLE demo_sales_so_i...`

D) `SELECT mat FROM demo sales cds material ve...` is not valid because `demo sales cds material ve` is not a valid data source in the given code. There is no CDS view or database table named `demo sales cds material ve`. The correct name of the CDS view is `demo_sales_cds_material_ve`, with underscores instead of spaces.

## Question 11

---

**Question Type:** MultipleChoice

---

For what kind of applications would you consider using on-stack developer extensions? Note: There are 2 correct answers to this question.

**Options:**

---

- A-** Applications that provide APIs for side by side SAP BTP apps
- B-** Applications that access SAP S/4HANA data using complex SQL
- C-** Applications that integrate data from several different systems
- D-** Applications that run separate from SAP S/4HANA

### **Answer:**

---

A, B

### **Explanation:**

---

On-stack developer extensibility is a type of extensibility that allows you to create development projects directly on the SAP S/4HANA Cloud technology stack. It gives you the opportunity to develop cloud-ready and upgrade-stable custom ABAP applications and services inside the SAP S/4HANA Cloud, public edition system. You can use the ABAP Development Tools in Eclipse to create and deploy your on-stack extensions. On-stack developer extensibility is suitable for the following kinds of applications:

Applications that provide APIs for side by side SAP BTP apps. On-stack developer extensibility allows you to create OData services or RESTful APIs based on CDS view entities or projection views. These services or APIs can expose SAP S/4HANA data and logic to other applications that run on the SAP Business Technology Platform (SAP BTP) or other platforms. This way, you can create a loosely coupled integration between your SAP S/4HANA system and your side by side SAP BTP apps.

Applications that access SAP S/4HANA data using complex SQL. On-stack developer extensibility allows you to use ABAP SQL to access SAP S/4HANA data using complex queries, such as joins, aggregations, filters, parameters, and code pushdown techniques. You can also use ABAP SQL to perform data manipulation operations, such as insert, update, delete, and upsert. This way, you can

create applications that require advanced data processing and analysis on SAP S/4HANA data.

The other kinds of applications are not suitable for on-stack developer extensibility, as they have different requirements and challenges. These kinds of applications are:

Applications that integrate data from several different systems. On-stack developer extensibility is not meant for creating applications that integrate data from multiple sources, such as other SAP systems, third-party systems, or cloud services. This is because on-stack developer extensibility does not support remote access or data replication, and it may cause performance or security issues. For this kind of applications, you should use side by side extensibility, which allows you to create applications that run on the SAP BTP and communicate with the SAP S/4HANA system via public APIs or events.

Applications that run separate from SAP S/4HANA. On-stack developer extensibility is not meant for creating applications that run independently from the SAP S/4HANA system, such as standalone apps, microservices, or web apps. This is because on-stack developer extensibility requires a tight coupling with the SAP S/4HANA system, and it may limit the scalability, flexibility, and portability of the applications. For this kind of applications, you should use side by side extensibility, which allows you to create applications that run on the SAP BTP and leverage the cloud-native features and services of the platform.

## Question 12

---

**Question Type:** MultipleChoice

---

When processing an internal table with the statement LOOP AT itab... ENDLOOP, what system variable contains the current row number?

**Options:**

---

A- sy-index

B- sy-subrc

C- sy-linno

D- sy-tabix

**Answer:**

---

D

**Explanation:**

---

When processing an internal table with the statement LOOP AT itab... ENDLOOP, the system variable that contains the current row number is sy-tabix. The sy-tabix variable is a predefined field of the system structure sy that holds the index or the row number of the current line in an internal table loop. The sy-tabix variable is initialized with the value 1 for the first loop pass and is incremented by 1 for each subsequent loop pass. The sy-tabix variable can be used to access or modify the current line of the internal table using the index access12.

**To Get Premium Files for C\_ABAPD\_2309 Visit**

**[https://www.p2pexams.com/products/c\\_abapd\\_2309](https://www.p2pexams.com/products/c_abapd_2309)**

**For More Free Questions Visit**

**<https://www.p2pexams.com/sap/pdf/c-abapd-2309>**

