1Z0-151$^{\text{Q\&As}}$

Oracle Fusion Middleware 11g: Build Applications with Oracle Forms

Pass Oracle 1Z0-151 Exam with 100% Guarantee

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

https://www.leads4pass.com/1z0-151.html

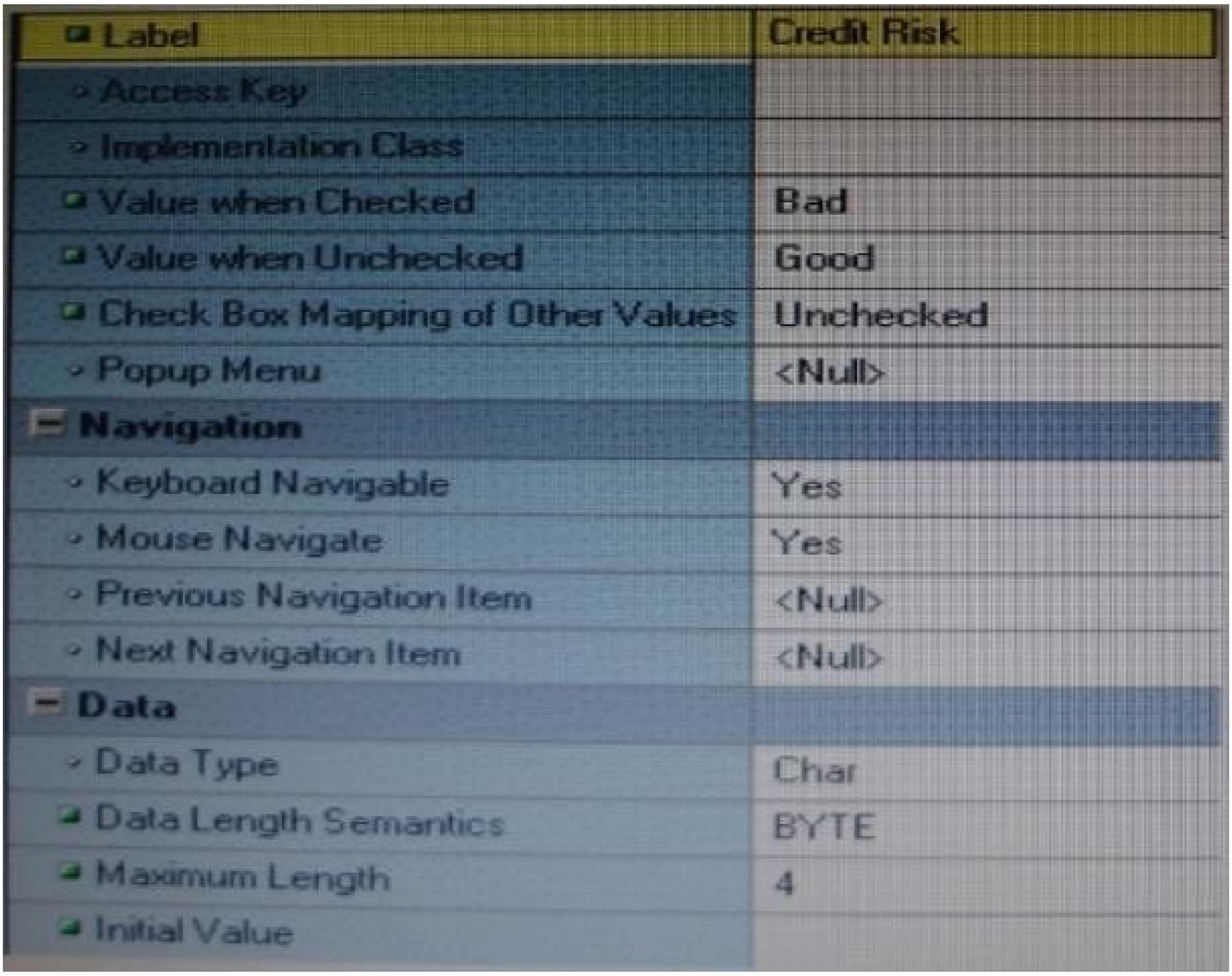
100% Passing Guarantee
100% Money Back Assurance

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

QUESTION 1

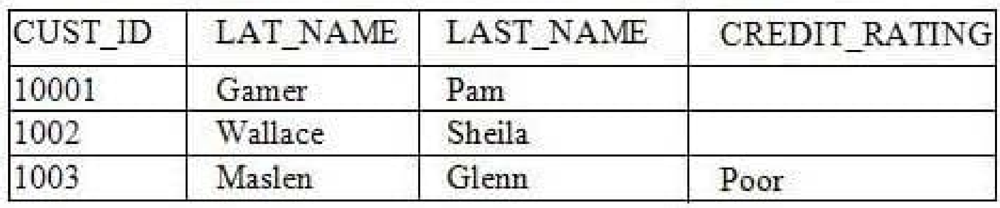
The Credit_Rating item in the Customers block of an Order Entry form must be restricted to three possible values: Good, Poor, or Null (the latter value indicates that the credit rating has not been determined). Order entry clerks must be able to update this item to any of the preceding three values.

You want to change the item to a check box labeled "Credit Risk" that should be selected if the customer has poor credit, the check box should not be selected if the customer\\'s credit rating is good or undetermined.



Label	Credit Risk
Access Key	
Implementation Class	
Value when Checked	Bad
Value when Unchecked	Good
Check Box Mapping of Other Values	Unchecked
Popup Menu	<Null>
Navigation	
Keyboard Navigable	Yes
Mouse Navigate	Yes
Previous Navigation Item	<Null>
Next Navigation Item	<Null>
Data	
Data Type	Char
Data Length Semantics	BYTE
Maximum Length	4
Initial Value	

You change the item type to Checkbox, set other properties as shown in the Exhibit, and then run your form and insert three records: two with good credit and the check box deselected, and one with poor credit and the check box selected. You commit the data and query the records, with the following result set:



CUST_ID	LAT_NAME	LAST_NAME	CREDIT_RATING
10001	Gamer	Pam	
1002	Wallace	Sheila	
1003	Maslen	Glenn	Poor

The first two records show an undetermined credit rating, although your intention was to set the value to Good for these customers. What change must you make in the properties of the Credit_Ratinq item to enable values of Good, Poor, and Null to be entered?

A. Change the initial Value property to Good.

B. Change the Check Box Mapping of Other Values property to Not Allowed.

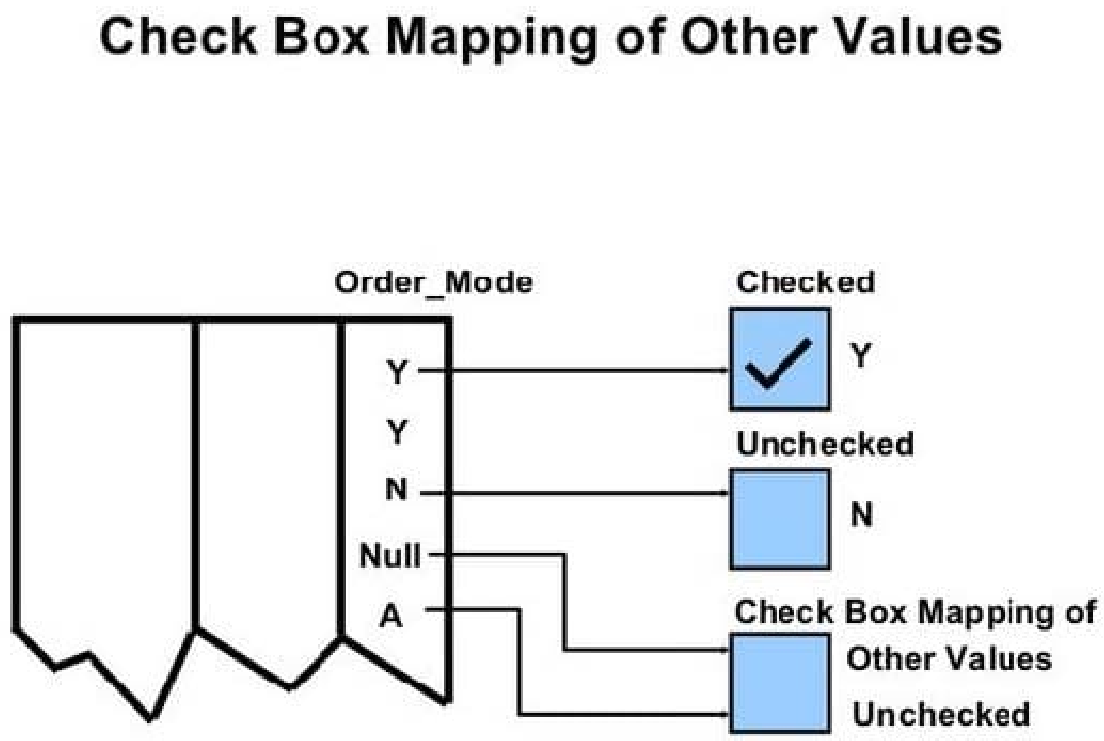
C. Change the initial Value property to Good and the Value When Unchecked property to Null.

D. Change the initial Value property to Good and the Chock Box Mapping of Other Valued property to Null.

E. Change the initial Value property to Good and the Check Box Mapping of Other Values property to checked.

F. Change the item type. It is not appropriate to use a check box to enable entry and update of more than two values in an item.

Correct Answer: E

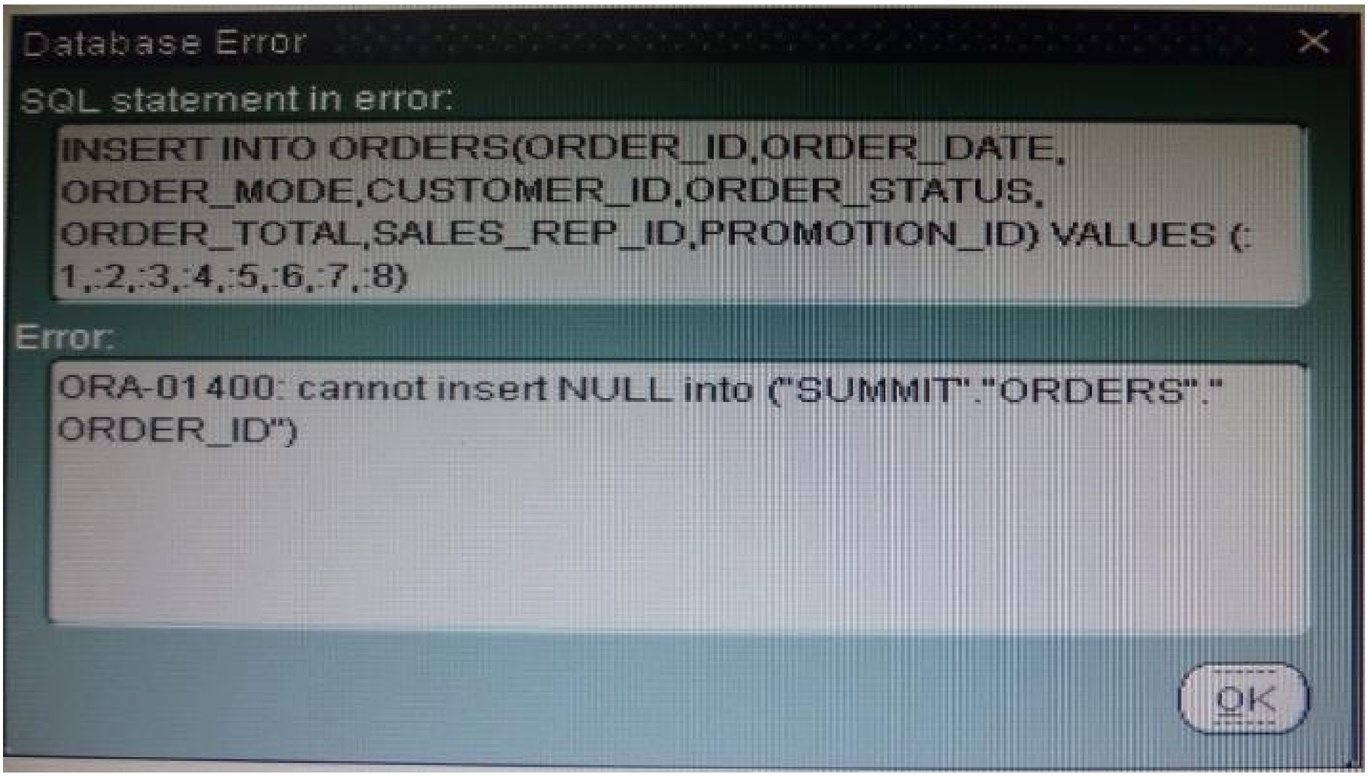


Incorrect answers:

F: A check box is not limited to two values.

QUESTION 2

View the Exhibit.



```
INSERT INTO ORDERS(ORDER_ID,ORDER_DATE,
ORDER_MODE,CUSTOMER_ID,ORDER_STATUS,
ORDER_TOTAL,SALES_REP_ID,PROMOTION_ID) VALUES (:
1,:2,:3,:4,:5,:6,:7,:8)
```

```
ORA-01400: cannot insert NULL into ("SUMMIT"."ORDERS"."
ORDER_ID")
```

You have placed the following code in a Post insert trigger on the Orders block:

SELECT orders_seq.NEXTVAL INTO :orders.order_id FORM SYS.dual;

You have also set Item properties for the Order_Id Item in the form so that users cannot directly enter an Order ID.

As the SUMMIT user, you run the form to test it. Yon are able to insert a record in the block, but when you click Save, you have a database error as shown in the Exhibit.

What is the probable cause of this error?

A. The code should be in Pre_insert trigger instead

B. The code should be in a database trigger instead

C. You should change the code to select the sequence number from SUMMIT.dual instead.

D. You should create the sequence in the database, because the runtime error is an indication that is does not exist.

E. You should assign a temporary value to the Order_Id item before saving the form.

Correct Answer: E

QUESTION 3

Yon have created a list item for the Credit__Rating field in the Customers form, the Finance department usually determines a range of scores that pertain to the customer, the values in the list are Excellent, Good, Fair, and Poor, corresponding to the ranges provided by Finance. The default value Fair, because that is the range that applies to most customers.

Sometimes, Finance can provide an exact credit score; so users must be able to enter the exact numerical value if it is known. Often clerks enter the customer data before the performance of a credit check, so they must be able to enter a blank value if the credit rating is not yet determined.

Users want to be able to select a blank value from the list, but the list does not display a blank line for them to select. How can you meet this requirement without changing the default value of the list item?

A. Add a blank line to the list item when you define the static values.

B. Set Mapping of other Values for the Credit_Rating Item to NULL.

C. Set the Required property of the Credit_Rating item to No.

D. Delete the NOT NULL constraint for the Credit_Rating Column in the database.

E. You cannot change this type of list item to display a blank value; users must delete the default value records if the credit rating is not known.

Correct Answer: A

QUESTION 4

To troubleshoot a problem with a form, you have added a call to the MESSAGE () built-in at the beginning of the When-Validate-Item trigger of the Customer_Id then in the Orders Block of the Orders form. The message simply states that the trigger is firing.

You run the form, make a change in Customer_Id, and then tab out of the item but the message does not appear. What are two possible causes for this problem?

A. The form is in Enter-Query mode.

B. The item is using an LOV for validation.

C. The validation unit of the form needs to be changed.

D. The MESSAGE () built-in is not allowed in validation triggers.

E. There is a syntax error in the call to the MESSAGE() built-in.

F. Validation for the Customer_Id item failed.

Correct Answer: AB

A: WHEN-VALIDATE-ITEM does not fire in ENTER-QUERY mode.

B: LOVs in ENTER-QUERY Mode

LOVs in ENTER-QUERY mode should be used sparingly, as Query Find is the preferred method for a user to locate records.

You should only code them where they dramatically improve the usability of ENTER-QUERY mode, and you expect this mode to be used regularly despite Query Find.

An LOV in ENTER-QUERY mode should display all values that the user can query, not just currently valid values.

EXAMPLE: An LOV for vendors in a purchase order form in enter-query mode shows all vendors that could ever be placed on a PO, not just the set of vendors that currently are allowed to be placed on a PO.

Do not reuse the entry LOV in ENTER_QUERY mode unless it provides the correct set of data for both modes.

Important: WHEN-VALIDATE-ITEM does not fire in ENTER-QUERY mode. Therefore, you cannot depend on the WHEN-VALIDATE-ITEM trigger to clear hidden fields when selecting from an ENTER-QUERY LOV.

Note: Validation occurs when you press enter, when you navigate away from the item, or when you save your block to the database.

Note 2: When-Validate-Item trigger Description Fires during the Validate the Item process. Specifically, it fires as the last part of item validation foritems with the New or Changed validation status. Definition Level form, block, or item Legal Commands SELECT statements, unrestricted built-ins Enter Query Mode no Usage Notes

*

Use a When-Validate-Item trigger to supplement Form Builder default item validation processing.

*

It is possible to write a When-Validate-Item trigger that changes the value of an item that FormBuilder is validating. If validation succeeds, Form Builder marks the changed item as Valid anddoes not re-validate it. While this behavior is necessary to avoid validation loops, it does make itpossible for your application to commit an invalid value to the database.

*

The setting you choose for the Defer Required Enforcement property can affect the When- Validate-Item trigger. See Defer_Required_Enforcement for details. On Failure If fired as part of validation initiated by navigation, navigation fails, and the focus remains on theoriginal item. Fires In Validate the Item

QUESTION 5

You have a form that called a database stored procedure. You do not want processing to continue, so

immediately after the call to the stored procedure, you add the following code:

IF NOT FORM_SUCCESS THEN

MESSAGE (\\'Stored Procedure failure\\');

RAISE FORM_TRIGGER_FAILURE;

END IF;

You test the code and input some data that intentionally causes the stored procedure to fail. However, the

message that you defined does not appear. What are two possible reasons for this?

A. You must handle database errors in an exception clause of the trigger.

B. You must test for FORM_FAILURE when testing for failure of a stored procedure.

C. You must test for FORM_FATAL when testing for failure of a stored procedure.

D. FORM_SUCCESS tests for failure of Forms built-ins, not stored procedures.

E. You cannot test for failure of a stored procedure, because database errors are transparent to Forms.

Correct Answer: AD

A: You should use DBMS_ERROR_CODE and DBMS_ERROR_TEXT in an ON-ERROR trigger to trap errors propagated from Oracle server from Forms.

Note: FORM_SUCCESS and FORM_TRIGGER_FAILURE

Either the FORM_SUCCESS built-in or the FORM_TRIGGER_FAILURE exception must be used to handle all Forms errors. FORM_SUCCESS is a Boolean function that evaluates to TRUE or FALSE depending on the success or failure of a Forms built-in. FORM_TRIGGER_FAILURE is an exception that has to be raised to stop further processing whenever an error occurs or FORM_SUCCESS evaluates to FALSE. The following sections describe their use in error-message handling.