

1Z0-151^{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.pass2lead.com/1z0-151.html>

100% Passing Guarantee
100% Money Back Assurance

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

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



QUESTION 1

Which statement is always true about using OPEN_FORM to open multiple forms in an application?

- A. Database transactions are always continued across forms.
- B. Code is always shared among forms.
- C. A calling form must pass data to a called form.
- D. A global record group is always created.
- E. Only A and B are true.
- F. Only B and C are true.
- G. Only C and D are true.
- H. None of the statements above are true.

Correct Answer: H

OPEN_FORM Opens the indicated form. Use OPEN_FORM to create multiple-form applications, that is, applications that open more than one form at the same time. OPEN_FORM can be instructed for the two forms to share the same database session and/or library data.

Note: CALL_FORM Runs an indicated form while keeping the parent form active. Oracle Forms runs the called form with the same Run form preferences as the parent form. When the called form is exited Oracle Forms processing resumes in the calling form at the point from which you initiated the call to CALL_FORM. CALL_FORM can be instructed to hide or not hide the calling form, replace or not replace the menu, open the new form in query only mode, share or not share library data.

NEW_FORM Exits the current form and enters the indicated form. The calling form is terminated as the parent form. If the calling form had been called by a higher form, Oracle Forms keeps the higher call active and treats it as a call to the new form.

Incorrect answers:

A: session_mode:

NO_SESSION (The default.): Specifies that the opened form should share the same database session as the current form.

SESSION: Specifies that a new, separate database session should be created for the opened form.

C: data_mode

NO_SHARE_LIBRARY_DATA (The default.): At runtime, Oracle Forms will not share data between forms that have identical libraries attached (at design time). SHARE_LIBRARY_DATA: At runtime, Oracle Forms will share data between forms that have identical libraries attached (at design time).

Reference: Difference between call_form, new_form and open_form

QUESTION 2

Which three are valid ways to populate a display item?

- A. initial Value property
- B. User input
- C. When-Button-Pressed trigger
- D. Calculation
- E. Executing a query

Correct Answer: ACD

C: You can use the attachable PL/SQL libraries to implement a flexible message function for messages that are displayed programmatically by the built-in routines MESSAGE or CHANGE_ALERT_MESSAGE, or by assigning a message to a display item from a trigger or procedure.

D: Creating Calculated Items

Change the Item Type to Display Item.

Set the Calculation Mode property to Formula.

Set the Formula property to the required formula.

Incorrect answers:

B: Display items and text items are fairly similar and share many of the same properties. The biggest difference between the two is that a user can navigate to a text item and change its value. This is not possible with a display item. As its name implies, it merely displays information.

Note: Display items do not allow any user interaction - they merely display data and never accept cursor focus.

Use a display item for the following situations:

*

Null-canvas fields

*

Context fields

*

Fields that act as titles or prompts

QUESTION 3

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.

QUESTION 4

You are planning the alerts that are needed for your Human Resources application. You wish to display the following in alerts:

*

A message to inform the user about being at the just record

*

A warning about a potential conflict with the data just entered

*

A message to display a validation error to the user

*

A warning that the salary is out of range and that asks whether the user wants to correct it

You want the note symbol () to appear on alerts that display only informative messages, the



warning symbol to appear on messages where you will allow the user to continue despite some data problem, and the alarm boll symbol to appear where the user will not be allowed to continue without correcting the situation that caused the alert to be displayed.

You want to define the minimum number of alerts possible and customize them at run time.

Which alerts should you define?

- A. One Note style alert and caution style alert
- B. One Note style alert and one Stop style alert
- C. On Caution style alert and one stop style alert
- D. On Note style alert, one Caution style alert, and one Stop style alert
- E. Two Caution style alerts and one Stop style alert

Correct Answer: D

In this scenario we need three different kind of alerts.

How to Create an Alert

Like other objects you create at design-time, alerts are created from the Object Navigator.

1.

Select the Alerts node in the Navigator, and then select Create.

2.

Define the properties of the alert in the Property Palette. Here are the properties that are specific to an alert. This is an abridged list.

Property	Description
Name	Name for this object
Title	Alert title
Alert Style	Defines the symbol that accompanies message: Stop, Caution, or Note
Button1, Button2, Button3	Labels for each of the three possible buttons (Null indicates that the button is not required.)
Default Alert Button	Button 1, Button 2, or Button 3
Message	Message that will appear in the alert (maximum 200 characters)

Note: Alerts are an alternative method for communicating with the operator. Because they display in a modal window, alerts provide an effective way of drawing attention and forcing the operator to answer the message before processing can continue.

Note 2: Potentially, you can create an alert for every separate alert message that you need to display, but this is usually unnecessary.

You can define a message for an alert at run time, before it is displayed to the operator. This means that a single alert can be used for displaying many messages, providing that the available buttons are suitable for responding to each of these messages.

Create an alert for each combination of:

*

Alert style required

*

Set of available buttons (and labels) for operator response For example, an application might require one Note-style alert with a single button (OK) for acknowledgment, one Caution alert with a similar button, and two Stop alerts that each provide a different combination of buttons for a reply. You can then assign a message to the appropriate alert before its display, through the SET_ALERT_PROPERTY built-in procedure.

Reference: Oracle Forms Student Guide, How to Create an Alert

QUESTION 5

You have been assigned to maintain a Forms application that was designed by a developer who has left the company.

The CV_Tools canvas contains several buttons. There is limited space on the canvas, but the first button is not quite wide enough to display its complete label. In the Layout Editor, you widen the button slightly and find that the width of all the buttons increases so that they now overlap each other.

What could have caused this problem?

- A. The buttons inherit their properties from the same property class.
- B. The buttons were created by dragging them from an object group.
- C. The buttons were created by using a Smart Class.
- D. All other buttons were subclassed from the first button.
- E. All other buttons were copied from the first button.
- F. The first button was subclassed from another button.
- G. The first button was copied from another button.

Correct Answer: D

D (not F): With subclassing you can make an exact copy, and then alter the properties of some objects if desired. If you change the parent class, the changes also apply to the properties of the subclassed object that you have not altered. However, any properties that you override remain overridden. This provides a powerful object inheritance model.

Incorrect answers:

A: A property class is a named object that contains a list of properties and their settings. You can use property classes to increase productivity and improve network performance.

Here are some of the uses of a property class:

*

It increases productivity by setting standard or frequently used values for common properties and associates them with several form builder objects.

*

It defines standard properties not just for one particular object but also for several at a time. This results in increased productivity, because it eliminates the time spent on setting identical properties for several objects.

*

It improves network performance by increasing the efficiency of message diffing.

C: A SmartClass is a special member of an Object Library. It can be used to easily subclass existing objects in a form using the SmartClass option from the right mouse button popup menu. E, G: When you copy an object, a separate, unique version of that object is created in the target module. Any objects owned by the copied object also get copied. Changes made to a copied object in the source module do not affect the copied object in the target module.

[1Z0-151 PDF Dumps](#)

[1Z0-151 VCE Dumps](#)

[1Z0-151 Study Guide](#)