



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.passapply.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

An LOV must be displayed several times in your form; therefore, good performance when displaying LOV is essential. In a When-New-Form-instance trigger, you want to save the ID of the LOV in a global variable so that you can use it in any code to display the LOV.

Which built-in would you use to get the ID of the LOV?

- A. SHOW_LOV
- B. FIND_LOV
- C. GET_ITEM_PROPERTY
- D. GET_LOV_PROPERTY
- E. GET_APPLICATION_PROPERTY

Correct Answer: B

Referencing Objects by Internal ID Finding the object ID:

```
lov_id := FIND_LOV(\\my_lov\\')
```

QUESTION 2

When users enter address information, you want them to be able to select the state from a static list of values. You have not used a list of states before, and there is no database table that contains state information.

What is the first step in creating such a list of values as quickly as possible?

- A. invoke the LOV wizard.
- B. Create a new record group that is based on a SQL query.
- C. Create a new static record group.
- D. Create a list item instead; a list of values is not appropriate for a static list.

Correct Answer: A

How to create LOV in Oracle forms?

List of Values(LOV) are used either when a selected list is too long and hence would not be appropriate for a drop down, but needs a search form to select the value.

Steps to create LOV functionality:

*

Click on the LOV icon in the object navigator and choose a manual/automatic wizard. This will create a record group.

*



Open a search form in the Dialog Page

*

Select values inside this form.

*

Return selected values to the original page.

QUESTION 3

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 4

You are implementing a JavaBean in a form, the bean has no visible component on the form when invoked, the bean displays an input text where users enter a zip code, the bean has a single method that returns a short weather forecast for that zip code as a character value.

How can you retrieve that value so that you can display it to the user?

- A. Use FBEAN.ENABLE_EVENT to register a listener for the bean event. Obtain the value of SYSTEM.CUSTOM_ITEM_EVENT in a When-Custom item Event trigger, and then use that value as an argument to the MESSAGE built in.
- B. Use FBEAN.REGISTER_BEAN as an argument to the MESSAGE built in to invoke the bean's method and return the value as a message displayed to the user.
- C. Use FBEAN.REGISTER_BEAN to register the bean, so that when the user enters a zip code into the bean's input text, the value is automatically displayed in the bean area item.
- D. Use FBEAN.ENABLE_EVENT to register a listener for the bean event. Obtain the value of SYSTEM.CUSTOM_EVENT_PARAMETER in a When-Custom-item-Event trigger, and then use that value as an argument to the MESSAGE built-in.

Correct Answer: D

When a user interacts with a JavaBean at run time, it usually causes an event to occur. You can use FBEAN.ENABLE_EVENT to register a listener for the event, so that when the event occurs Forms will fire the When-Custom-Item-Event trigger. In this trigger, you can code a response to the event. The :SYSTEM.CUSTOM_ITEM_EVENT and :SYSTEM.CUSTOM_EVENT_PARAMETERS variables contain the name of the event and information the bean is sending to the form.

QUESTION 5

The Orders form has four blocks. The Orders and Order_items block are on the CV_Order content canvas; the inventories block items are on the CV_inventories content canvas; and Control block buttons are on the CV_Buttons toolbar canvas. All buttons have mouse Navigate set to No.

The Order_Items block is a detail of Orders. The inventories block is a detail of Order_Items, showing the stock of the selected product.

There is a button in the Control block with a When-Button-Pressed trigger:



```
IF GET_CANVAS_PROPERTY(:SYSTEM.cursor_item, item_canvas) = '\\CV_ORDER\\' THEN GO_BLOCK  
(\\inventories\\)  
ELSE  
GO_BLOCK(\\orders\\);  
END IF;
```

When you run the form and click the button, navigation does not occur, and the form displays the runtime error "FRM-41053: Cannot find Canvas: invalid ID." What should you do to correct this problem?

- A. Change the sequence of blocks in the Object Navigator
- B. Change the Mouse Navigator property of the button to yes
- C. in the first line of code, change the built-in to GET_ITEM_PROPERTY
- D. in the first line of code, change the system variable to: SYSTEM.CURSOR_CANVAS.
- E. in the first line of code, change the CV_ORDER to lowercase
- F. Change the argument to the GO_BLOCK built-ins to uppercase

Correct Answer: D

Note: GET_CANVAS_PROPERTY built-in Description Returns the given canvas property for the given canvas. . Syntax
FUNCTION GET_CANVAS_PROPERTY(canvas_id Canvas ,property NUMBER);FUNCTION
GET_CANVAS_PROPERTY(canvas_name VARCHAR2 ,property NUMBER)

[Latest 1Z0-151 Dumps](#)

[1Z0-151 PDF Dumps](#)

[1Z0-151 Study Guide](#)