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
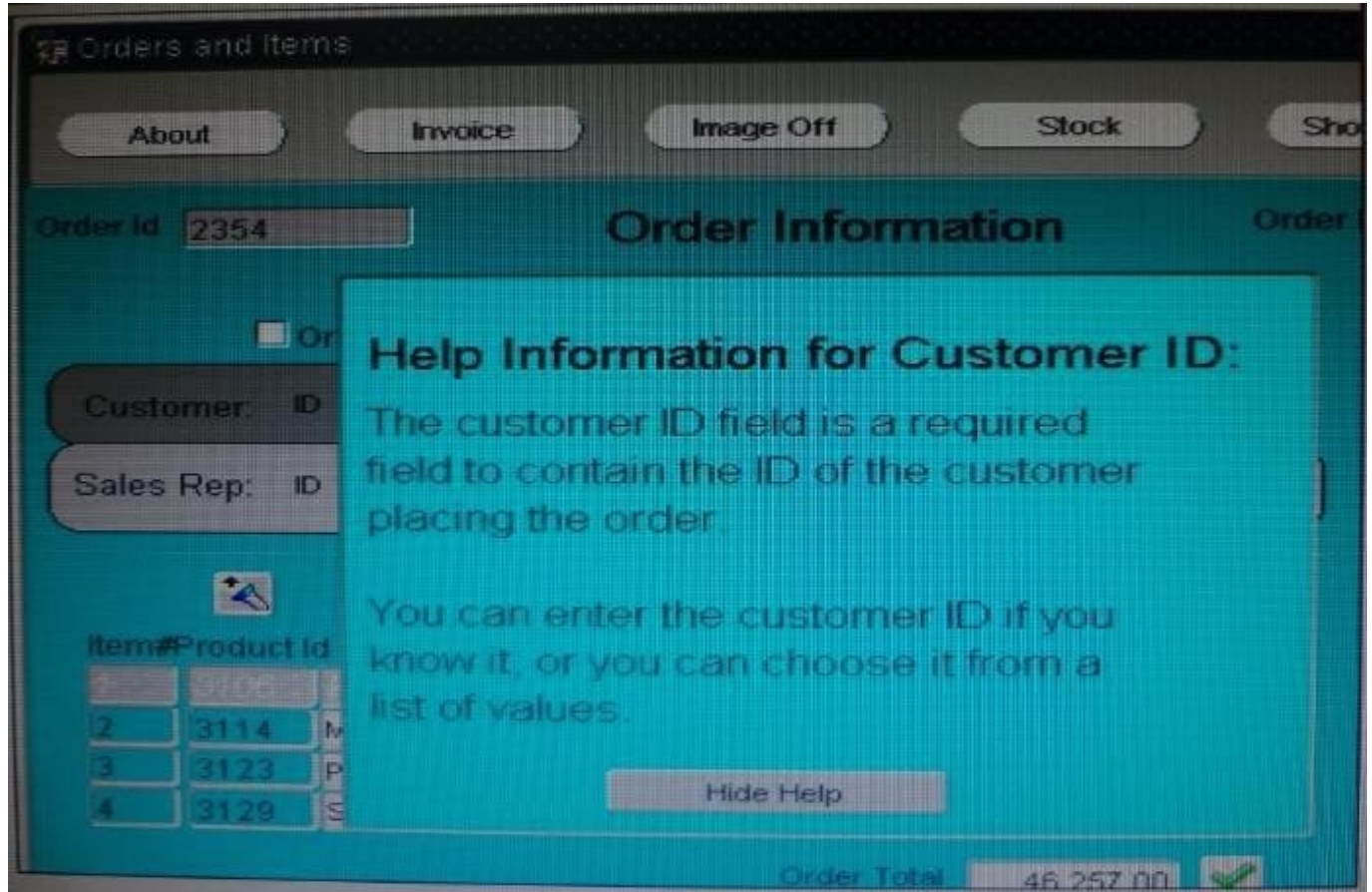
⚙ **Instant Download** After Purchase

⚙ **100% Money Back** Guarantee

⚙ **365 Days** Free Update

⚙ **800,000+** Satisfied Customers

QUESTION 1

View the Exhibit.

The orders form contains two canvases. Orders_CV displays one order and all of its order items. Help-CV displays context-sensitive help.

When users invoke the help screen for the Customer_Id item, the help information obscures the Custormer_Id item, as shown in the Exhibit. Users would like to be able to see both the item and its help information simultaneously.

How can you move the help information to the right so that the Customer_Id item is visible?

A. increase the Viewport X Position on the Help_CV canvas.

B. increase the Viewport X Position on Canvas on the Orders_CV canvas.

C. Decrease the Width on the Help_CV canvas.

D. in the Layout Editor for the Orders_CV canvas, select View > Stacked Views, and then select the Help_CV canvas. Drag the Help_CV canvas to the right of the Customer_Id item.

Correct Answer: A

Note: Viewport X Position on Canvas, Viewport Y Position on Canvas property Description Specifies the location of the view\'s upper left corner relative to the upper left corner of the canvas. The size and location of the viewport define the view ; that is, the part of the canvas that is actually visible in the window to which the canvas is assigned. Applies to

canvas

QUESTION 2

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

The form uses different property classes to standardize the appearance of objects in the form. For example, all buttons should be of the same width.

The CV_Tools canvas contains several buttons. The Print invoice button is not quite wide enough to display its complete label.

In forms Builder, you open the Button_PC property class and change its Width property to a higher number. All the buttons become larger except the Print invoice button, which remains its original size. What could have caused this problem?

A. The Print invoice button was copied from, rather than subclassed from, the Button_PC property class.

B. The subclass information property for the Print invoice button is blank.

C. The Width property of the Print invoice button is an inherited property.

D. The Width property of the Button_PC property class is a variant property.

E. The X Position of the print invoice button is a variant property.

F. The X Position Property is not included in the Button_PC property

Correct Answer: A

QUESTION 3

The Orders form has the following triggers defined:

1.

Post-Text-Item on Customer_Id

2.

Pre-Text-Item on Sales_Rep_Id

3.

When-New-Item-instance on Sales_Rep_Id

The form\\'s Validation Unit property is set to Record.

When the user navigates from Customer_Id to Sales_Rep_Id, which triggers fire and what is the sequence?

A. 1, 2, and 3

B. 1, 3, and 2

C. 1 and 2 only

D. 2 and 3 only

E. 3 only

Correct Answer: A

All three triggers fire when you navigate

QUESTION 4

Which four types of variables must have their names preceded by a colon when they are referenced in trigger code?

A. PL/SQL variables

B. Form items

C. global variables

D. system variables

E. parameters

Correct Answer: ABCD

D: Example:

On calling the

raiseEvent method, a trigger named WHEN-CUSTOM-JAVASCRIPT-EVENT is fired on the server side.

declare

event_val varchar2(300):= :system.javascript_event_value; begin

if (:system.javascript_event_name=\\'show\\') then

handleShowEvent(event_val);

elsif(:system.javascript_event_name=\\'grab\\') then

handleGrabEvent(event_val);

else

null;

end if;

end;

QUESTION 5

You are using a PL/SQL program unit in the Orders form to display an image for the selected product in

the Order Items block. The code is called from several different triggers in the form.

The code (with line numbers added) is:

1.

PROCEDURE get_image IS

2.

Product_image_id ITEM := FIND_ITEM (\\'control.product_image\\');

3.

Filename VARCHAR2(250);

4.

BEGIN

5.

Filename := TO_CHAR(:order_items.product_id) || \\'.jpg\\';

6.

READ_IMAGE_FILE (filename, \\'jpeg\\', product_image_id);

7.

END;

Management wants all codes to be in PL/SQL libraries to facilitate reuse. You create a PL/SQL library and

drag the program unit from the orders form to the Program Units node of the library.

You then delete the program unit from the Orders form.

What three things must you do to compile the PL/SQL library and to use the code in the Orders form?

A. Change line 5 to: filename := to_char(name_in \\'order_items.product_id\\')) || \\'.jpg\\';.

B. insert the RELEASE keyword between lines 1 and 2.

C. Delete the BEGIN and END statements (lines 4 and 7).

D. Change product_id to an in parameter and filename to an OUT parameter.

E. Save the library.

F. Generate the a .plx file.

G. Attach the library to the orders form.

H. Change the way the program unit is called in the Orders form.

Correct Answer: EGH

E: Save and compile the library.

G: select the Attached Libraries node and click Create.

In the Attach Library dialog box, click Browse to navigate to the saved library and open it, then answer Yes

to the alert that asks if you want to remove the path.

H: Modify the to simply call the procedure from the attached library: