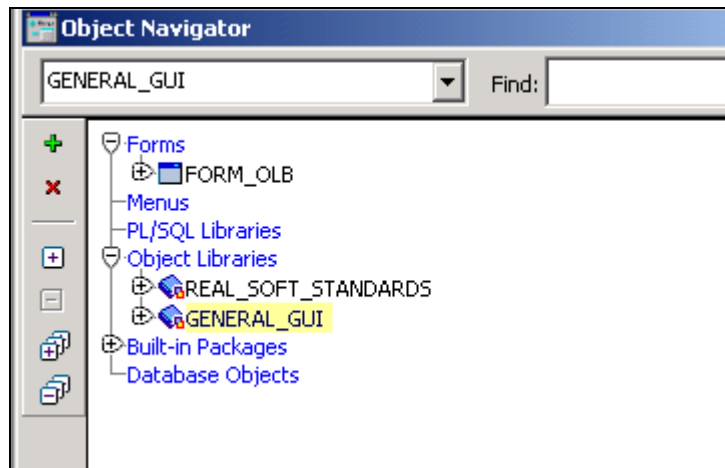


CHAPTER 14

What is The Object Library?

The object library became part of Oracle Forms Builder starting with FORMS V5.0 . It appears in the navigator as a module by itself.



In any development environment, you will always have standards to which you want your developers to adhere as well as common objects that can be reused throughout the development effort.

The Object Library provides an easy method of reusing objects and enforcing standards across the entire development organization.

You can use the Object Library to create, store, maintain, and distribute standard and reusable objects.

There are several advantages to using object libraries to develop applications:

- 1.Object libraries are automatically re-opened when you startup Form Builder, making your reusable objects immediately accessible.
- 2.You can associate multiple object libraries with an application. For example, you can create an object library specifically for corporate standards, and you can create an object library to satisfy project-specific requirements.

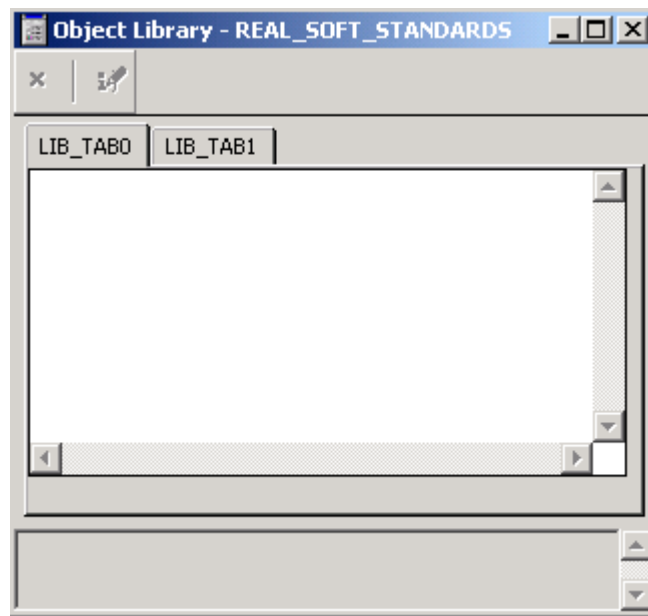
3. Object libraries feature “SmartClasses” objects that you define as being the standard. You use SmartClasses to convert objects to standard objects.

In essence then, Object libraries store a collection of Oracle Forms objects much like PL/SQL libraries store Oracle Forms Functions and Procedures.

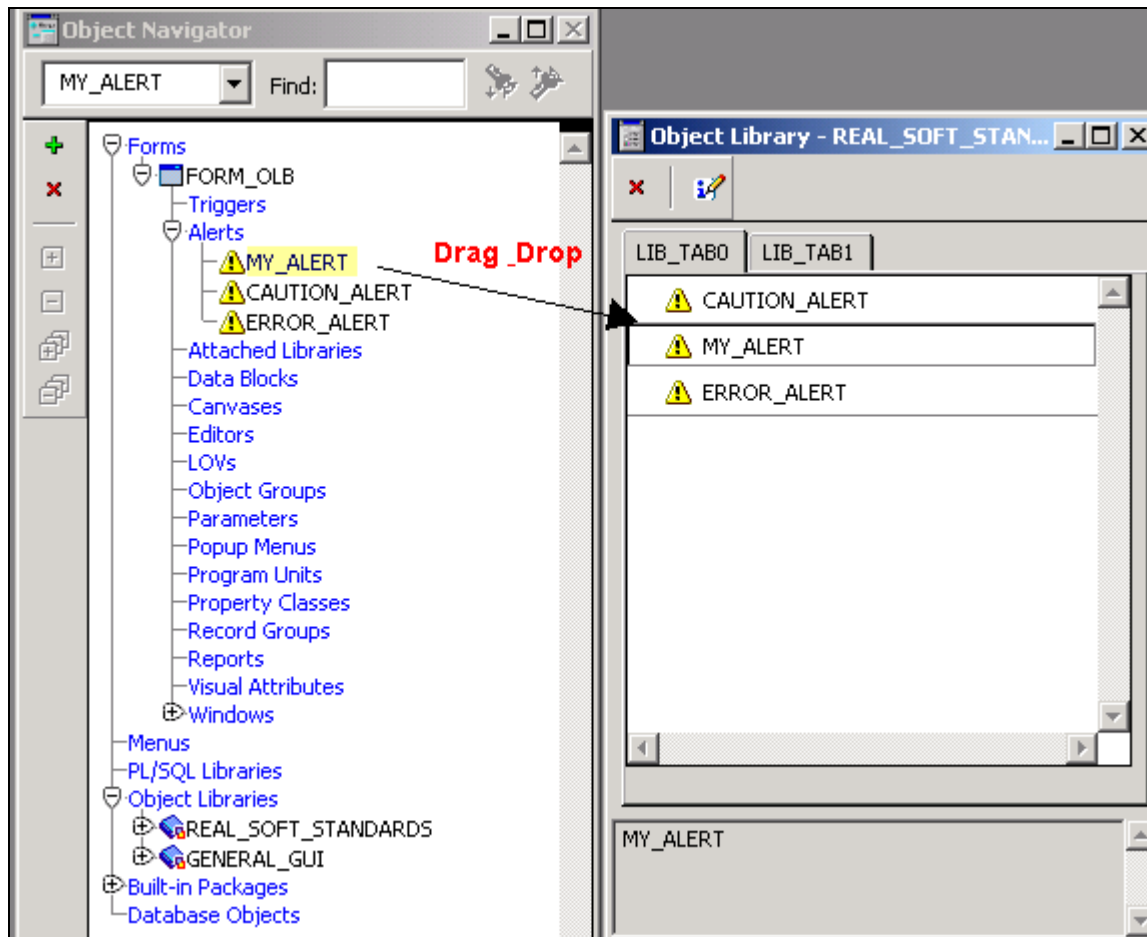
How to Create Object Libraries?

Assume that you have created three Alerts that you expect to reuse many times in your application. An object library will serve in storing these objects and making them available to other forms.

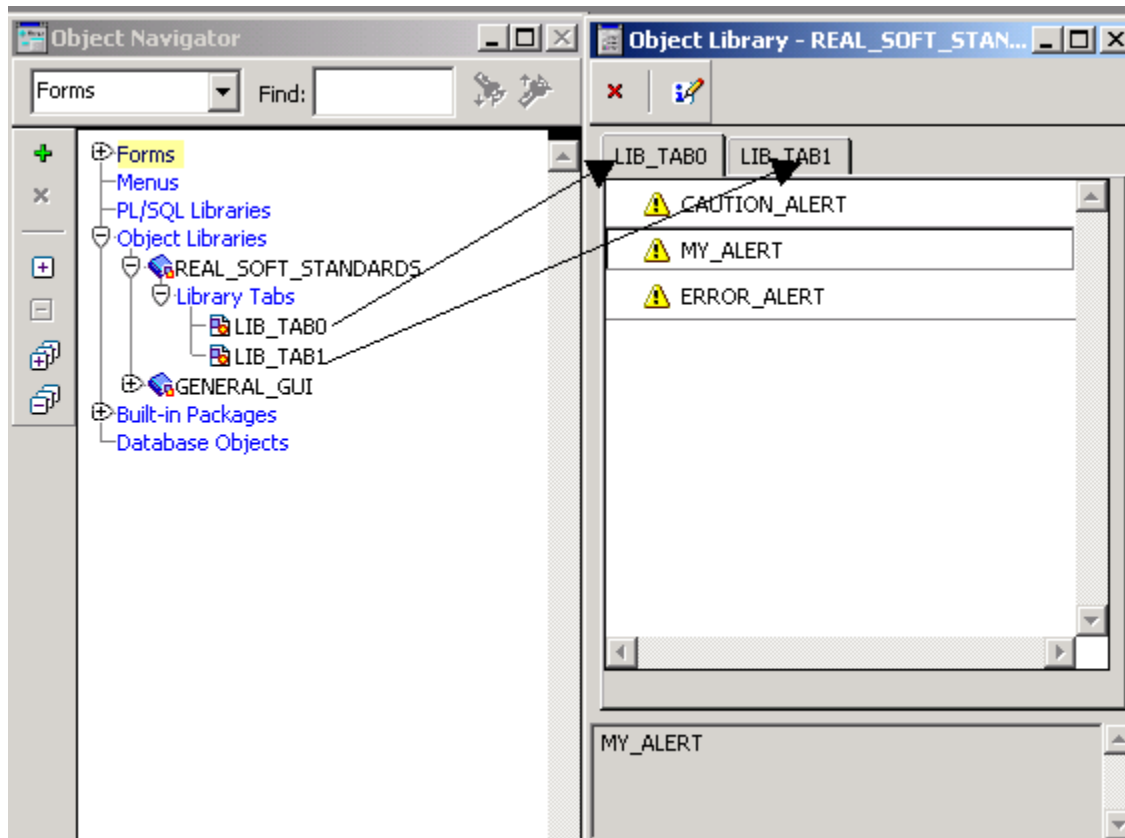
-To build a new object library use **Tools→Object Library**, or alternatively, select the Object libraries node in the object navigator and double click, Oracle Forms Builder will display the following window:



- Drag the Alerts from the Object Navigator to the newly created object library as shown below:

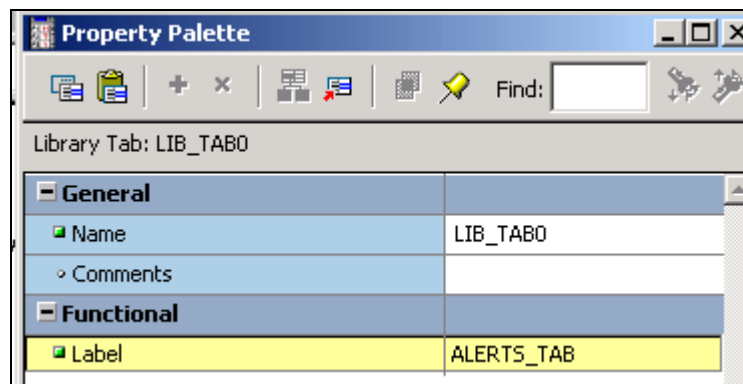


- The Object Navigator will have the following hierarchy.



Note that the new object library was created with two tabs (LIB_TAB0 and LIB_TAB1). Each one of these two tabs has its own properties. It is required that you change the first tab title to ALERTS. What you need to do is to activate the property palette of the first tab (Select it and press F4, or use **Tools** → **Property palette**).

- After the property palette appears, change the TITLE property to ALERT.

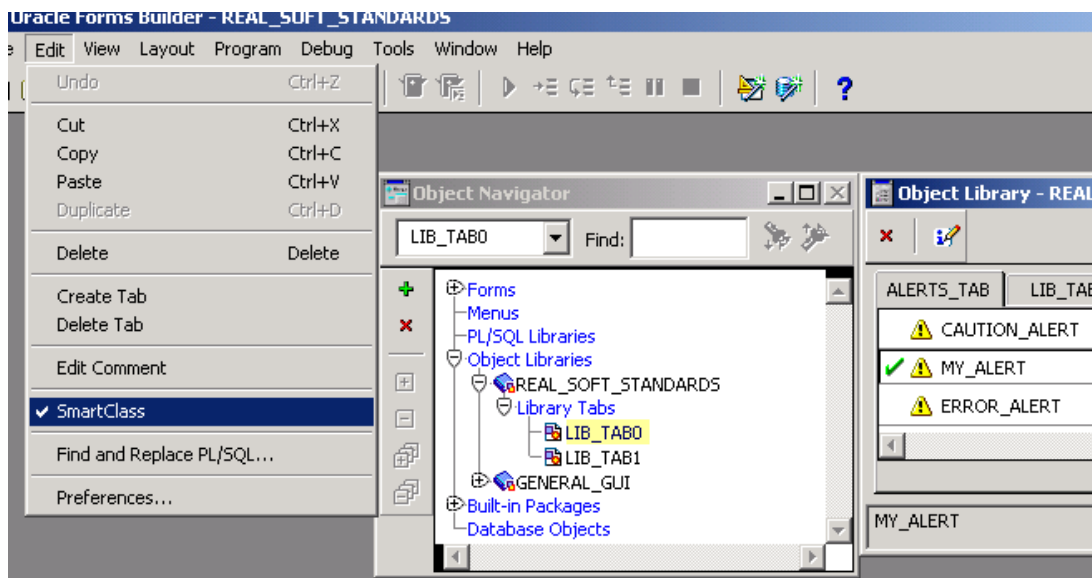


What Are SmartClasses?

A SmartClass is an object that you create within the Object Library like any ordinary object. What is special about a SmartClass is that once you define an object as a SmartClass, you can create new objects based on the SmartClass by dragging the desired SmartClass to the Object Navigator, or you can apply the SmartClass to an existing object by choosing SmartClasses from the popup menu.

How to define an object as a smartclass?

- 1- Invoke the Object library by using **Tools→Object Library**.
- 2- In the object Library, click the desired object. Assume that you want to define the STD_ALERT Object as a smartclass. Click the ALERT Library Tab and click the STD_ALERT Object.
- 3- Choose **Edit→Smartclass**. The STD_ALERT will then be immediately checked with a green check sign as shown below:



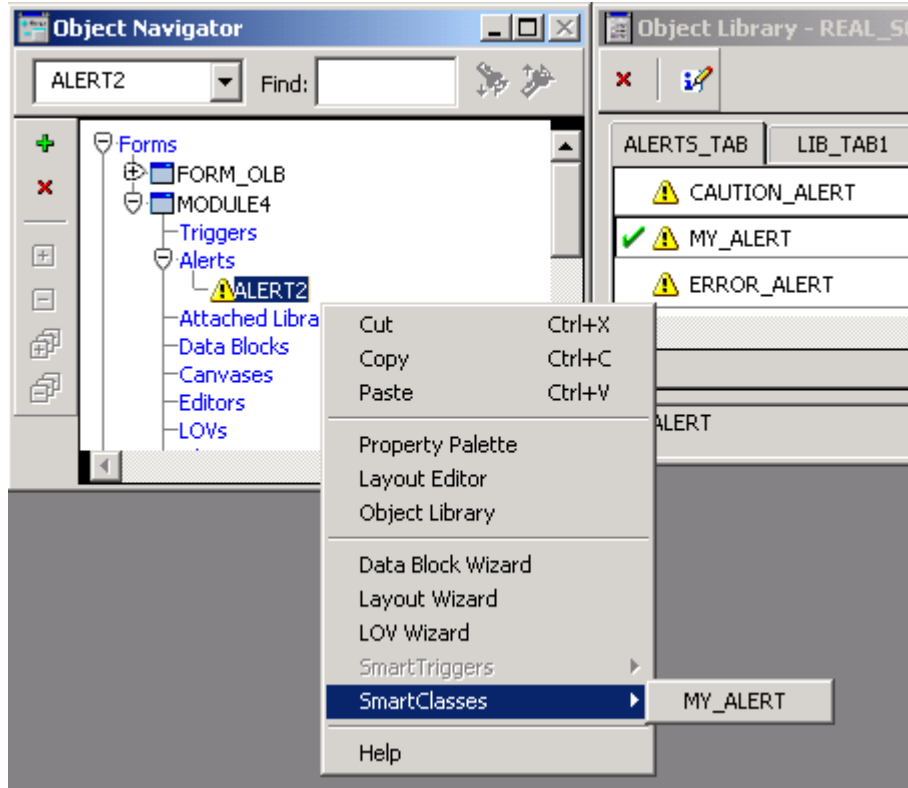
How to create an object based on a SmartClass?

- 1-In the Object Library, click the desired SmartClass.
- 2-Drag the SmartClass to the desired module and node.

How to Modify an existing Object based on a SmartClass?

- 1- Selected the desired object in the Object Navigator.

- 2- Click the right mouse button to display the popup menu that appears below:



When you choose the SmartClasses option in the popup menu, all define SmartClass will appear. You can select STD_ALERT.

How Does a SmartClass differ from other Objects?

SmartClasses differ from the other objects in the Object Library in that any object that is SmartClassed is always a subclassed object.

By default, when you drag an object (that is not a SmartClass) to a form, the object created is a copy, not a subclassed object.

Other objects in the Object Library can only be used to create new object instances, never to modify an existing object. In contrast, SmartClasses can be applied to existing objects.

Oracle Ships demo Object Libraries with the Developer/2000 Rel 2.0, 6, 6i and 9i. In this section, we will demonstrate how to display a CALANDER based on The *STANDARDS* object library

Please follow the following steps

This section is incomplete, please ignore the rest

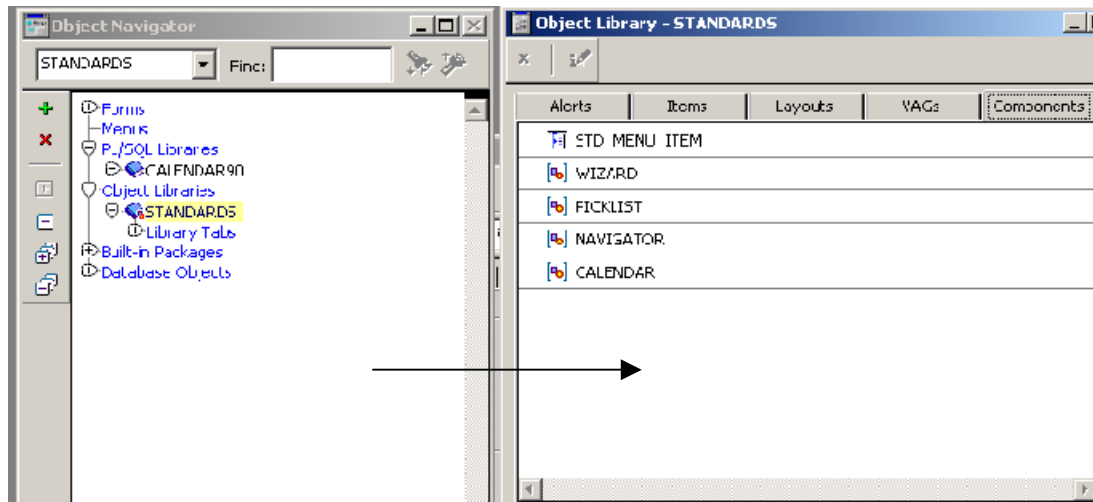
http://otn.oracle.com/sample_code/products/forms/content.html

- 1- Create a default form based on EMP table
- 2- In the Object Navigator, Open the stndrd20.olb object Library. It can be found in the following path:

“c:\orawin95\tools\devdem20\demo\formsstndrd20.olb”.

- 3- Click on the STANDARDS object Library

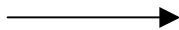
4-

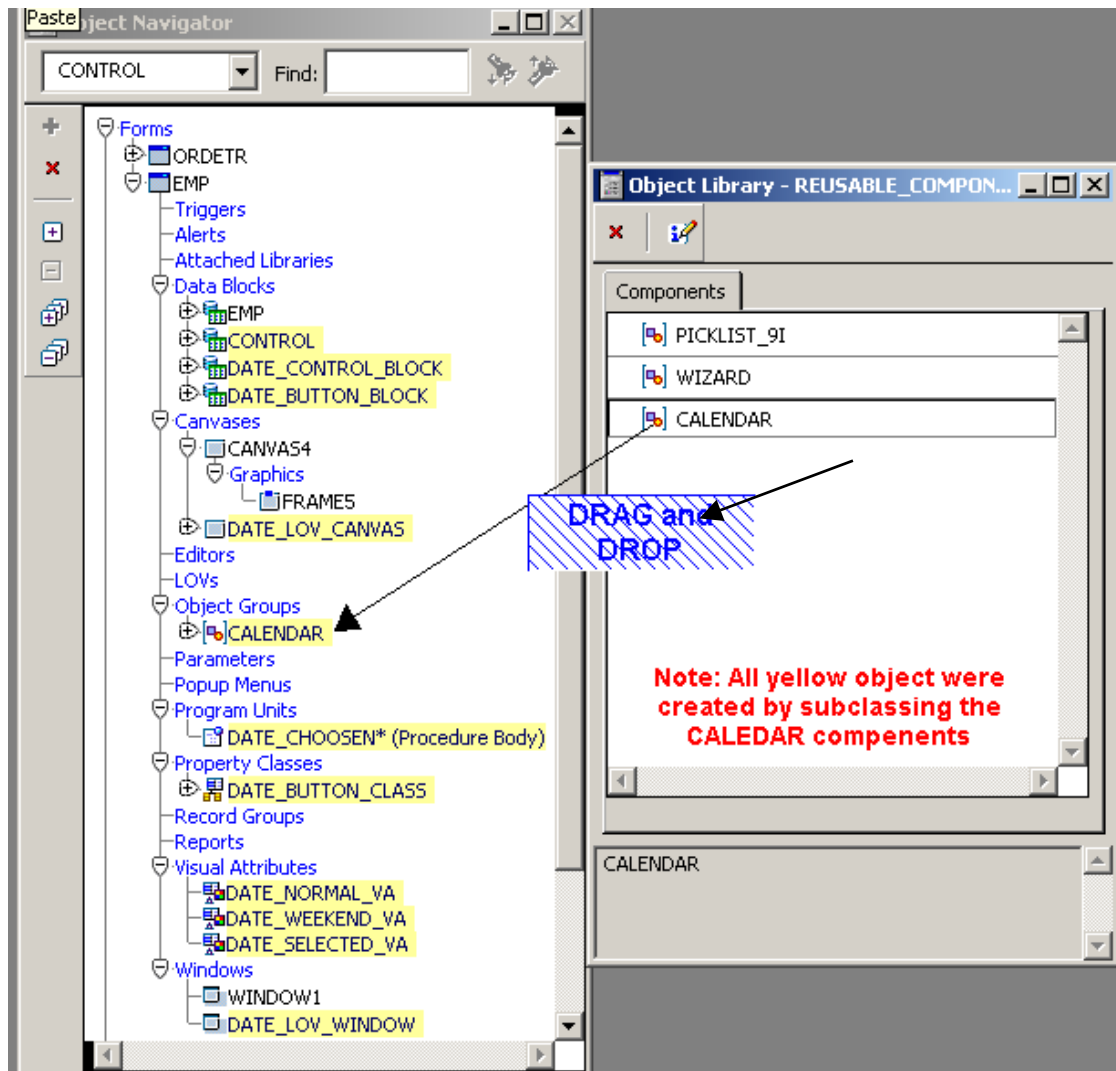


- 5- Then choose the components TAB and the object library windows appears as shown above (The window on the right)

- 6- Choose on the CALENDAR component. This object is of type Object Groups.

- 7- Drag the Calendar object from the Object Library to the Object Groups in the Object Navigator of your new form as shown below:





8- Attach a Library called Calendar.pll to your form Create a trigger.

9- Create a *When-mouse-double-click* on the HIREDATE item with the following PL/SQL code:

```

date_lov.get_date(sysdate,      -- initial date
                  'emp.hiredate', -- return block.item
                  240,          -- window x position
                  60,           -- window y position
                  'Start Date',  -- window title
                  'OK',         -- ok button label
                  'Cancel',     -- cancel button label
                  TRUE,         -- highlight weekend days
                  FALSE,        -- autoconfirm selection
                  FALSE);      -- autoskip after selection

```

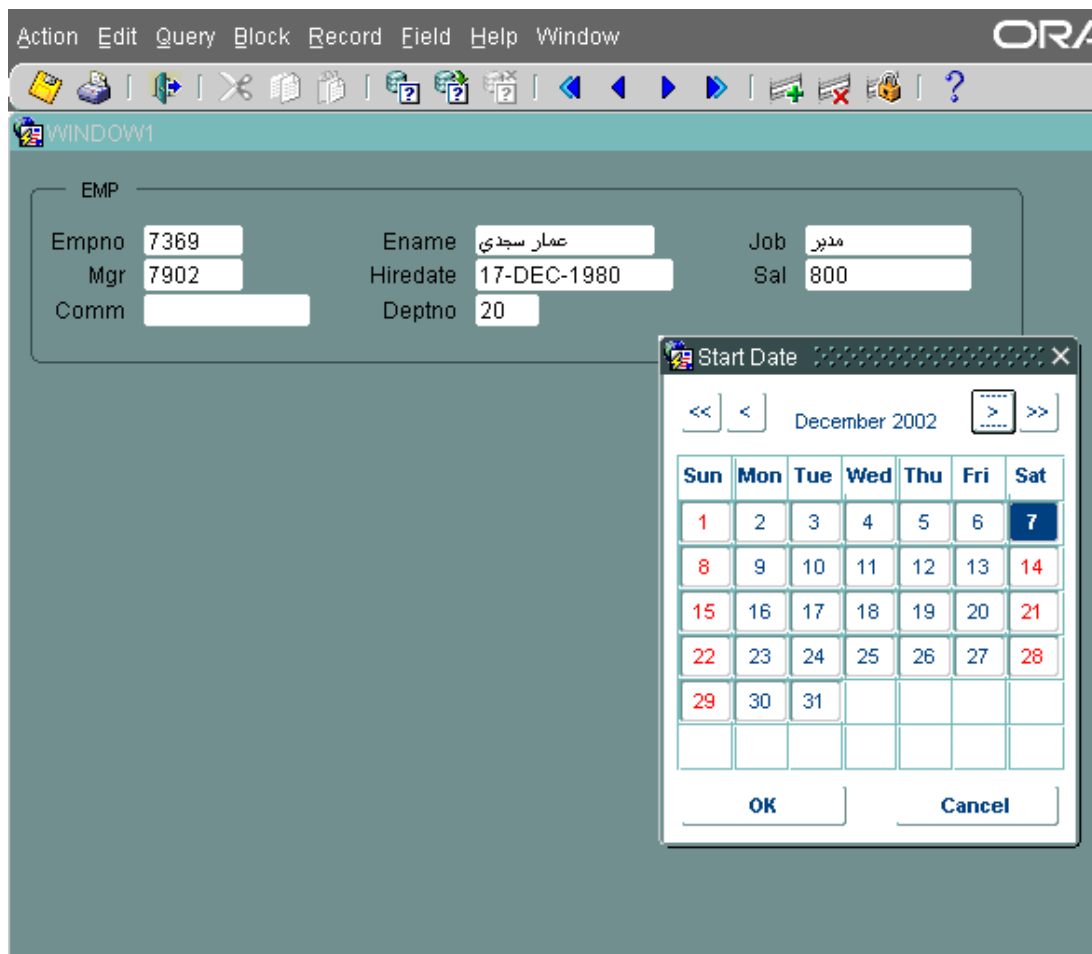

The `date_lov.get_date` is a PL/SQL library call

10. Create a new procedure in the *PROGRAM UNIT* object in the object Navigator and call it **Date_choosen**. Write the following PL/SQL code for the procedure. (This could be already part of the calendar component)

```
PROCEDURE date_choosen IS
BEGIN
    copy(to_char(date_lov.current_lov_date, 'dd-mon-yyyy'),
date_lov.date_lov_return_item);
    go_item(date_lov.date_lov_return_item);
    if date_lov.lov_auto_skip = TRUE then
        next_item;
    end if;
END;
```

11-Compile and Run the form

12- Enter a new record. When you reach the Hiredate item, double click and the following Window appears:



This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.