R12 Oracle EBS Suite: Introduction - IBM Graduate Program

Student Guide – Volume 2

D81890GC10

Edition 1.0

June 2013

ORACLE

Contents *(Application Object Library (AOL) - IBM Graduate Program)*

1

Application Object Library

Function Security

Objectives

After Completion of this lesson you should be able to to following:

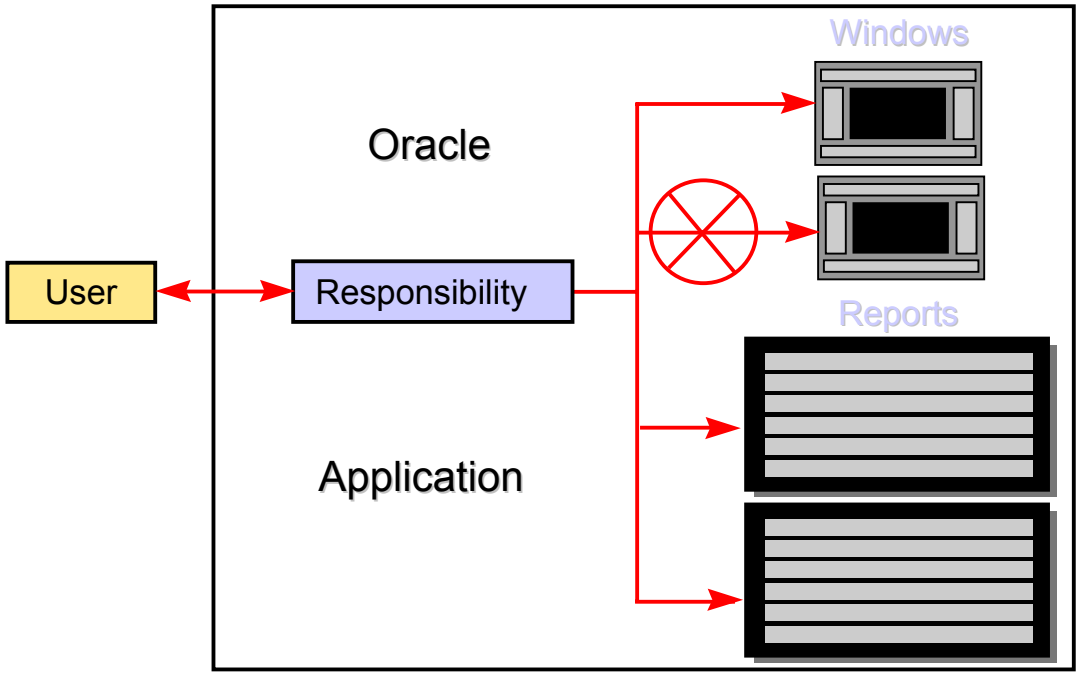
- Define an application user and assign responsibilities
- Use and create responsibilities. Customize application privileges for individual users or categories of users
- Restrict access to menu items and functions by responsibility
- Define custom menus

ORACLE

Lesson Aim

Function Security restricts user access to individual menus of functions, such as forms, HTML pages, or widgets (such as buttons in a form) within an application. Function Security allows you to define a user and assign the user one or more responsibilities, where each responsibility has a menu associated with it. Function Security by itself restricts access to various functions, but it does not restrict access to the data a user can see or what actions a user can perform on that data. This lesson discusses defining users, responsibilities, and menus.

Managing Application Security: Overview



Managing Application Security: Overview

In an Oracle Application, the system administrator manages security by creating user signons and assigning them to one or more responsibilities. Users then have access to all the functionality associated with that responsibility.

User Security

You authorize a user to sign on to Oracle Applications by defining an application user. You then assign one or more responsibilities to the new user.

Responsibility Security

A responsibility is a collection of authorizations that allow access to:

- A specific application or applications
- A set of books
- A restricted list of windows, functions, and reports

Each user has one or more responsibilities, and several users can share the same responsibility.

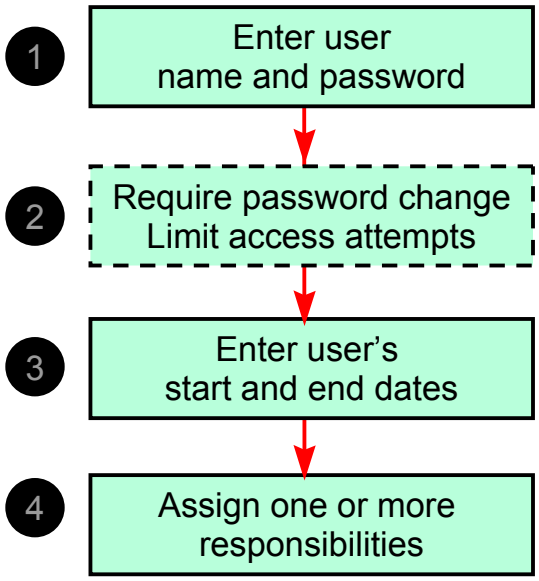
Managing Application Security: Overview (continued)

A system administrator can assign users any of the standard responsibilities provided with Oracle Applications or create new custom responsibilities as needed.

Self-Service Applications Security

Oracle Self-Service Web Applications use columns, rows, and values in database tables to define what information users can access. Table columns represent “attributes” that can be assigned to a responsibility. These attributes are defined in the Web Application Dictionary.

Defining New Application User



Steps 1, 3, and 4 are required

ORACLE

Defining a New User

(N) Security > User > Define

Note: All Navigation paths, unless otherwise noted, are from the System Administrator Responsibility.

Although defining user accounts may be the last task you complete in setting up function security for your installation, we will cover this task first so that you can complete the following sections by logging into Oracle Applications with your own user account.

Define an authorized user of Oracle Applications by specifying a username and password. Grant application privileges by assigning one or more responsibilities to the user. The user will be able to access functions and reports via the assigned responsibilities. Responsibilities will be covered later in this chapter.

Note that in previous releases of Oracle Applications, user passwords were treated as case insensitive. Now, Oracle Applications user passwords can optionally be treated as case sensitive. Case-sensitivity is controlled by the site-level profile option Signon Password Case.

Custom Applications

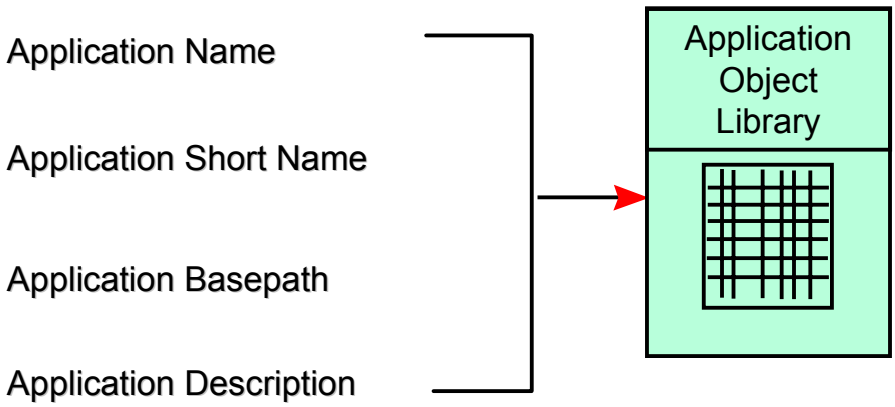
An Application is composed of Components such as forms menus and Concurrent Programs

Custom Applications

An application is composed of components such as forms, menus, and concurrent programs.

An application serves as a unique context for these components. The next three slides discuss necessary steps in implementing security for a custom application.

Defining an Application



Defining an Application

(N) Application > Register

You can protect custom functions, forms, reports, and programs from being lost during upgrades by registering them under a custom application.

In the Applications window you will supply the following information:

- Application – Enter a user-friendly name that will appear in lists seen by the user.
- Short Name – Oracle Applications uses this short name to identify forms, menus concurrent programs, and other components of your application.
- Basepath – Enter the base path where the forms, reports, and program files are located.

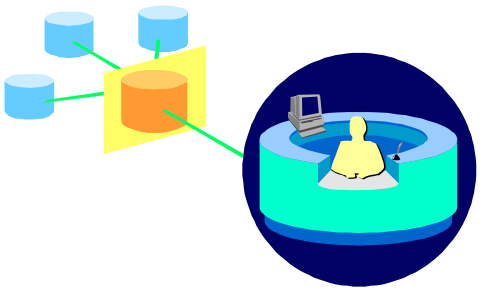
Make sure your base path is unique to prevent other applications from writing to the same directory.

For a complete explanation of the fields in this form see:

(Help) Applied Technology > Oracle Applications System Administration > Configuration > Applications DBA > Applications Window

Registering an oracle ID

You can give access to an Oracle account (Oracle ID) by working with an Oracle Database Administrator



Ask DBA to create a new oracle username and password that connects to an existing oracle account(Oracle ID)

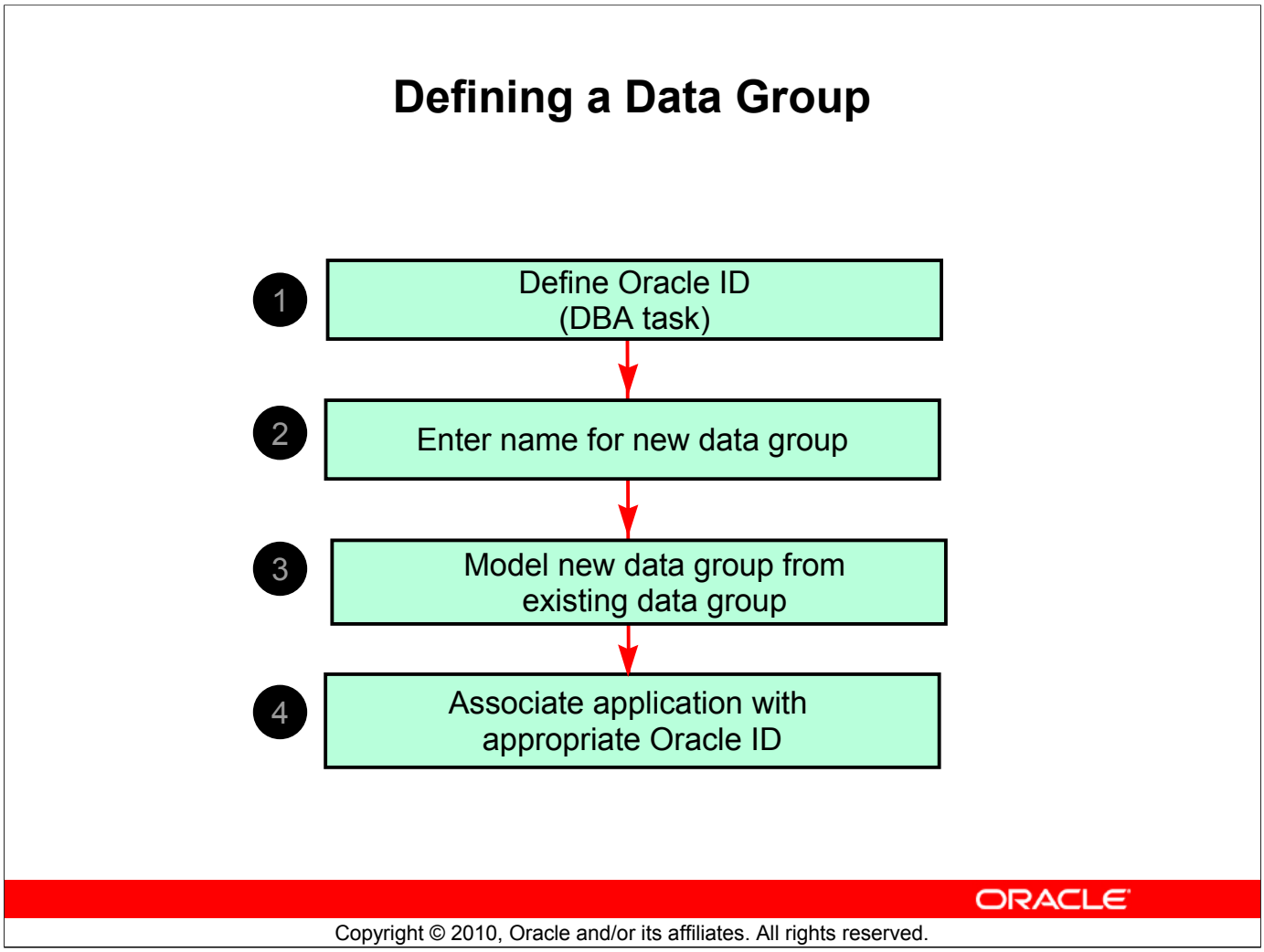
Registering an Oracle ID

(N) Security > ORACLE > Register

You need to register an Oracle ID if you create a custom application using Oracle Application Object Library.

Note that:

- Only database administrators can create Oracle accounts.
- Only create a new Oracle ID for a custom extension to Oracle Applications.
- The install group designates which data group the application is associated with. For applications that span all data groups, the install group is 0. Install groups numbered 1 or greater are associated with one specific data group. Your database administrator can supply the correct install group number.



Defining Data Groups

By defining a data group, you can determine which Oracle account (Oracle ID) an application's windows, reports, or concurrent programs connect to. Use data groups to grant application database account privileges to a responsibility and the requests that it submits

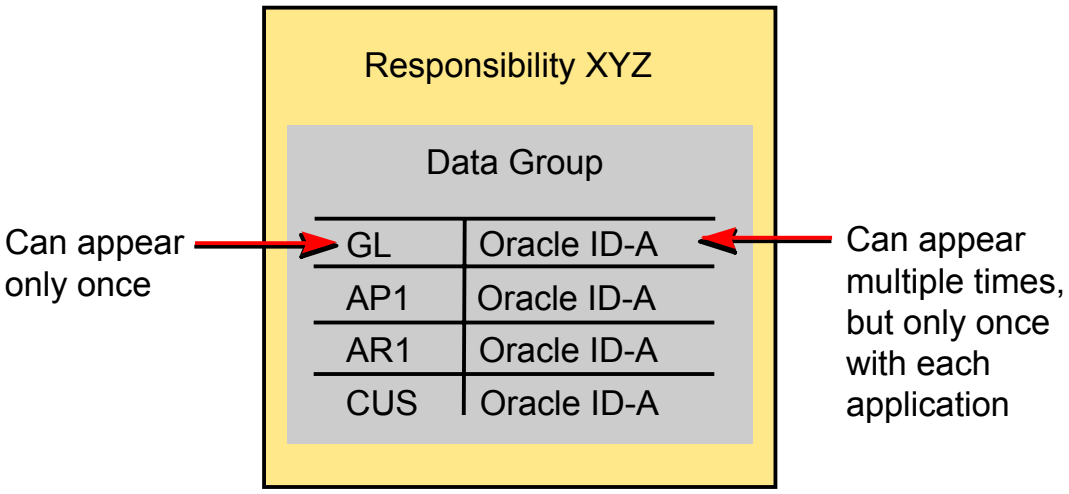
Data Groups Serve Two Purposes

- Identify the Oracle ID to which forms connect when users select the responsibility. Choose one application-Oracle ID pair from the data group to associate with your responsibility's windows.
- Concurrent managers use a data group to match the application that owns a report or concurrent program with a unique Oracle ID.

Additionally, you can incorporate custom applications by granting application database access to custom applications that you develop. Alternatively, use custom applications to name customized versions of concurrent programs, responsibilities, and concurrent managers to protect your customizations during an upgrade.

Data Groups

A Data Group is collection of collection of pairings of an application with an oracle ID



ORACLE

What Is a Data Group?

A data group is a collection of pairings of an application with an Oracle ID. Data groups automatically support concurrent processing and cross-application reporting. They guarantee that an application connects to a unique application database account.

Note: The installation process automatically defines data groups for Oracle Applications. For custom applications, you should use the the standard data groups as well. Information on data groups is given for legacy purposes.

Application-Oracle ID Pairs

- An application can be listed only once in a data group.
- An Oracle ID can be paired with more than one application.
- A custom application registered with Oracle Applications can be included in a data group.

Data Groups and Application Object Library

Application Object Library owns the database tables referred to during concurrent processing and the standard submission of reports by any Oracle Application. Therefore all applications need access to the Application Object Library tables. When you are defining a data group, the application Application Object Library is automatically included. The Application Object Library's Oracle ID cannot be updated or deleted.

Relating Data Groups to Forms and Programs

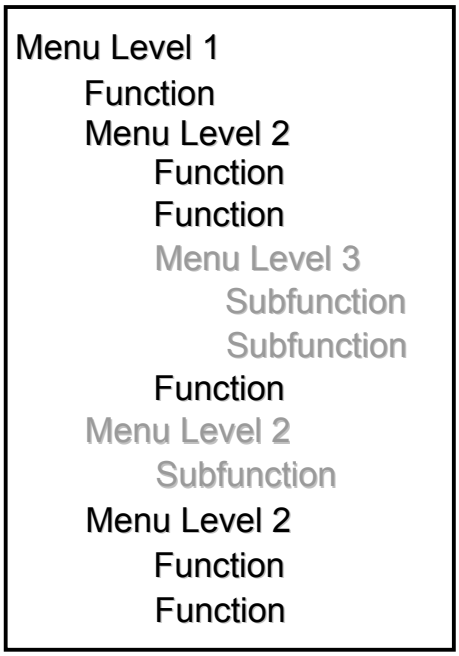
You can control the relationship among applications, forms, and concurrent programs by defining a data group.

Applications, Forms, and Programs

- A window connects to the application database account designated by the responsibility associated with the application.
- A data group determines the pairing of an application with a unique application database account or Oracle ID.
- A program connects to the application database account associated with the application that owns the program.
- Application-Oracle ID Pairs
- An Oracle ID is a username and password that allows access to application tables in an Oracle database.
- A data group lists the Oracle ID assigned to each Oracle application.
- A custom application registered with Oracle Applications can be included in a data group and paired with an Oracle ID.

.

Securing Functions



Function: A set of Executable code available as menu option

Subfunction: A subset of form's Functionality

ORACLE

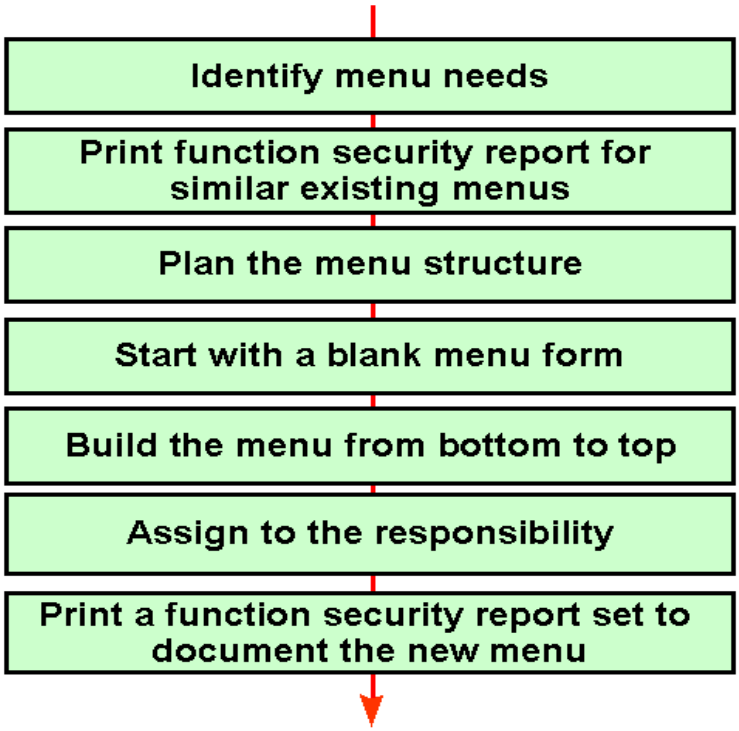
Securing Functions

You can manage security by controlling access to individual functions through menu definitions.

About Functions

- A function is a set of code in Oracle Applications that is executed only if the name of the function is present in a list maintained within a responsibility.
- There are two types of functions: a form function and a nonform function or subfunction.
- A subfunction represents a securable subset of a form's functionality.
- Adding Functions to or Removing Functions from a Responsibility
- Maintain menu structures while eliminating specific functionality.
- Exclude individual functions from a responsibility.Adding or Removing Menus of Functions
- Use menus to group functions together.
- Exclude groups of functions by excluding a menu from a responsibility.

Defining Menu



ORACLE

Menus Link Functions to Responsibilities

Use the Menus form to define menus pointing to functions that you want to make available to a new responsibility.

Make New Responsibilities, Not New Menus If possible, apply exclusion rules to existing menus to customize a responsibility rather than constructing an entirely new menu structure.

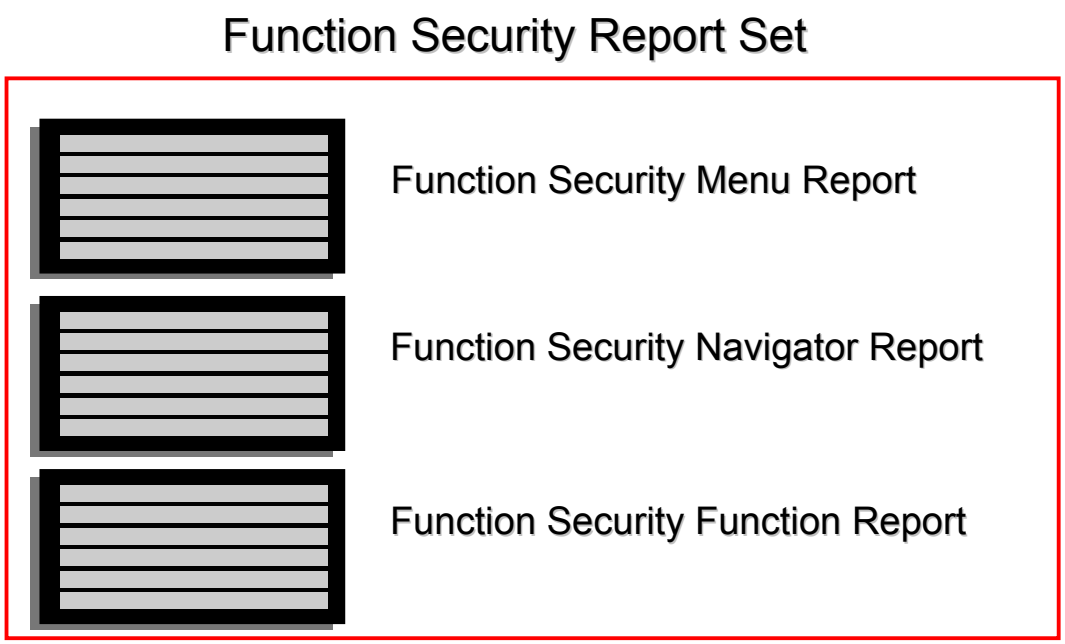
Determine the Application Functionality Required

- Different jobs require access to different functions.
- Identify predefined menus, forms, and form subfunctions to use as entries when defining a new menu.

Plan Your Menu Structure

- Start with a blank Menus form (blank screen). Menus cannot be copied. A menu saved under a different name overwrites the original menu (there is no Save As feature).
- Start with the lowest-level menus. A menu must be defined before it can be selected as an entry on another menu.
- Assign menus and functions to higher-level menus.
- Assign the menu structure to the new responsibility by using the responsibilities form.
- Document your menu structure by printing the Function Security Menu Report

Identifying Existing Menu Structures



Designing a New Menu Structure

Before designing a new menu structure, identify any possible existing menu structures that can be modified or modeled from, and document custom menus by using reports in the Function Security Report Set.

Function Security Menu Report

- Lists the full menu name of the responsibility
- Indicates any excluded menu items, with the rules that exclude them Function Security Navigator Report
- Lists the menu as it appears in the Navigator for the responsibility specified
- Does not include items excluded by function security rules Function Security Function Report
- Lists the functions accessible by the responsibility specified
- Does not include items excluded by function security rules

Creating Menu

Use following guidelines to build your menu

- Build your menu from bottom. A menu structure must already exist for a menu at higher level to reference it
- Give your menu both internal and user name
- The sequence number specifies the order in which your options are displayed on the menu
- The navigator prompt is the prompt that the user sees to invoke this function or menu
- Each entry on the menu definition form refers to either a function or another submenu

ORACLE

Creating a Menu

Use the Menus window to create a menu:

(N) Application > Menu

After you save your changes in this form, the Compile Security concurrent program is automatically submitted to compile the menu data. Compiling your menu data allows for the system to determine more quickly, during a user's session, whether a function is available to a particular responsibility or menu.

New Menus form fields:

- Menu Type – used to specify the purpose of the menu. The possible values are:
  - Standard – for menus that would be used in the Navigator form
  - Tab – for menus used in Self-Service applications
  - Security – for menus used to aggregate functions for data security or specific function security purposes, but would not be used in the Navigator form.
- Grant check box – checking this box indicates that this function is automatically enabled for the user.

Creating a Menu (continued)

Note: If a menu entry has both a submenu and a function defined, then the behavior depends on whether or not the function is executable. If it is executable, then the submenu is treated as content to be rendered by the function. The submenu will not appear on the navigation tree, but will be available in function security tests (FND_FUNCTION.TEST calls). If the function is not executable, then it is treated as a "tag" for enforcing exclusion rules, and the submenu is displayed on the navigation tree.

A function is considered executable if it can be executed directly from the current running user interface.

Modifying Existing Menu Definition

- Menus are called by their user menu name
- Any change to user menu name takes effect immediately
- Any existing menus that call the modified menu use the new name
- The previous name no longer appears
- Any menu entry modifications takes effect immediately

Modifying a Menu

You can modify existing custom menus to suit specific job functions.

Overwrite a User Menu Name

When you change a menu's user menu name, the menu entries are not affected. The menu's definition still exists, but under a new name. Other menus that call the menu by its old user menu name now call the same menu by its new (revised) user menu name.

The old user menu name is no longer valid. When you are defining menus, or if you are selecting main menus when defining a responsibility, the previously named menu is no longer displayed in any list of values.

Modify a Menu Entry

When you are modifying a previously defined menu, all other menus that call that menu display the menu's modifications. For example, if you modify XXX_GL_USER by adding another prompt that calls a form, all menus that call GL_SUPERUSER4.0 display the additional prompt when XXX_GL_USER is displayed.

Note: Modifying shipped menu definitions is not supported. The customized definitions may be overwritten during an upgrade process.

Adding a Custom Form Function

Custom form functions can be added to a menu using the following steps:

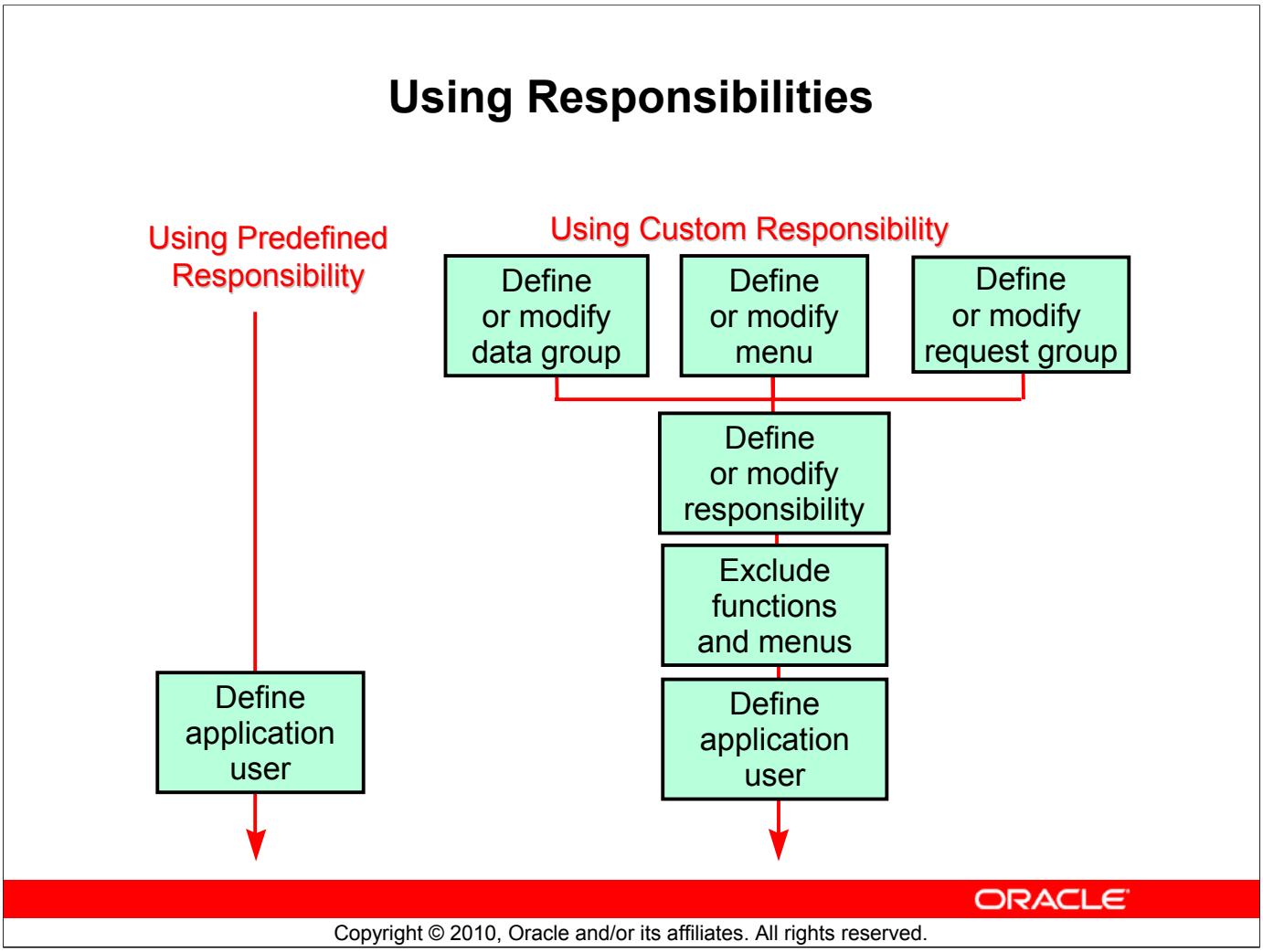
- Registration of the form in the forms window
- Creation of the function in the forms function window to provide access to the form
- Addition of form function to menu

ORACLE

Procedure for incorporating a custom form function

Use the following procedure to incorporate a custom form function.

1. The application developer (using the Application Developer responsibility) registers the form in the Forms window.
2. The application developer or system administrator creates a function that accesses the new form in the Form Functions window.
3. The application developer or system administrator adds the form function to a menu in the Menus window.



Creating a New Responsibility

Generally you relate new application users to existing predefined responsibilities. However, you can customize an existing responsibility or create new responsibilities to accommodate the needs of different users or different categories of users. When creating a new responsibility, it is generally easier to modify an existing responsibility in one of two ways:

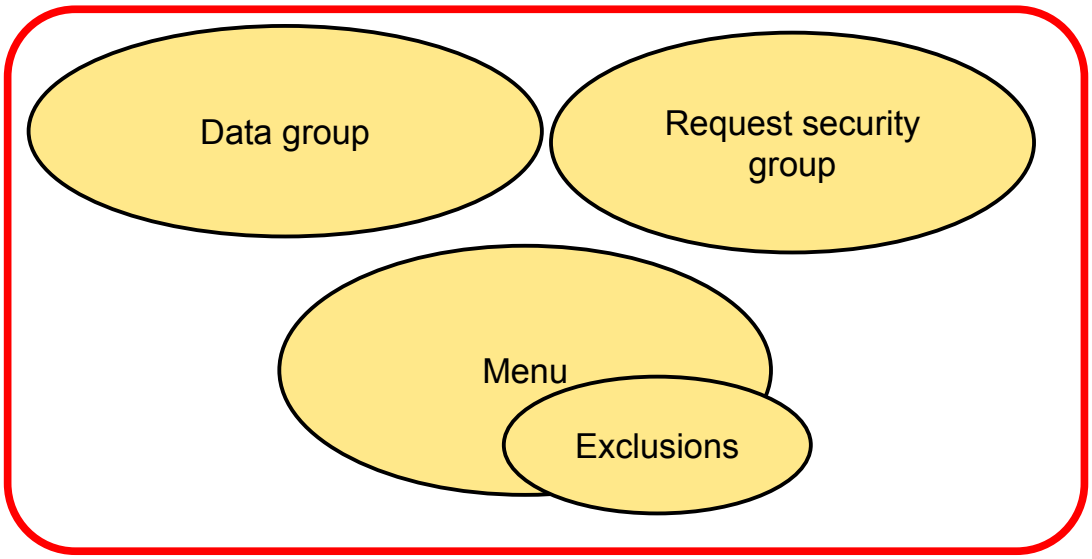
- Extend Privileges a Responsibility Owns When users require additional reporting and summary information, you can:
- Use request groups to add additional program or report privileges to a responsibility
- Use menus to add windows and tasks to a responsibility
- Restrict Privileges a Responsibility Owns Sometimes it is easier to remove authorizations from an existing responsibility.

To do this you can:

- Use exclusions against a responsibility to limit menu and function access privileges to those required for job duties
- • Use request groups to limit program or report privileges for a responsibility that requires only data entry privileges

After you have defined a new responsibility, you can associate it with an application user.

Responsibility Components



Components of a Responsibility

A responsibility has two required components and two optional components.

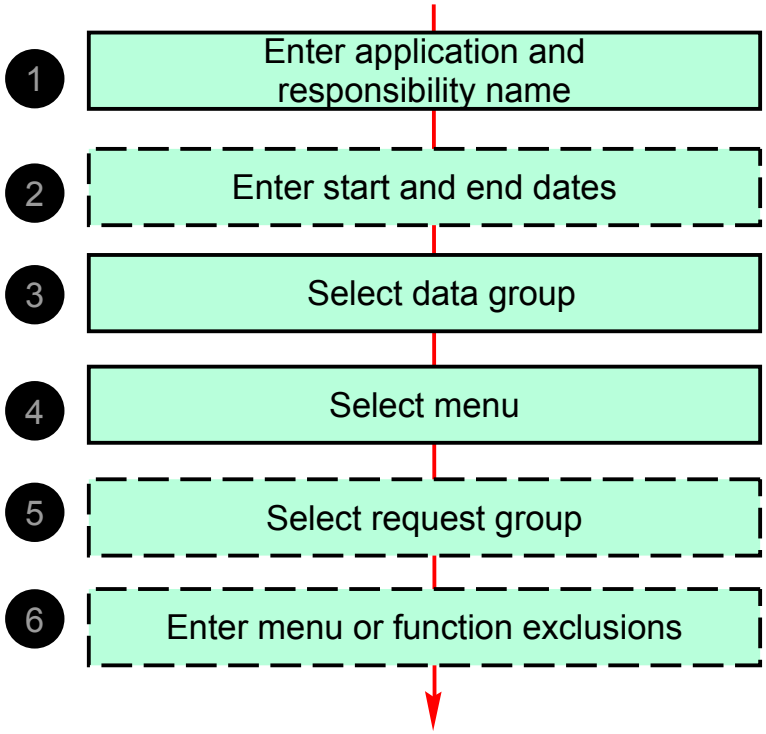
Required Components

- **Data group:** specifies the Oracle Application database accounts to which a responsibility's forms and concurrent programs connect.
- **Menu:** specifies the forms that a responsibility can display and the functions it can access.

Optional Components

- **Request security group:** lists the concurrent programs that a responsibility can run. When a request group is assigned to a responsibility, it is referred to as a request security group.
- **Exclusions:** modify a responsibility's access to the forms and functions specified by amenu.

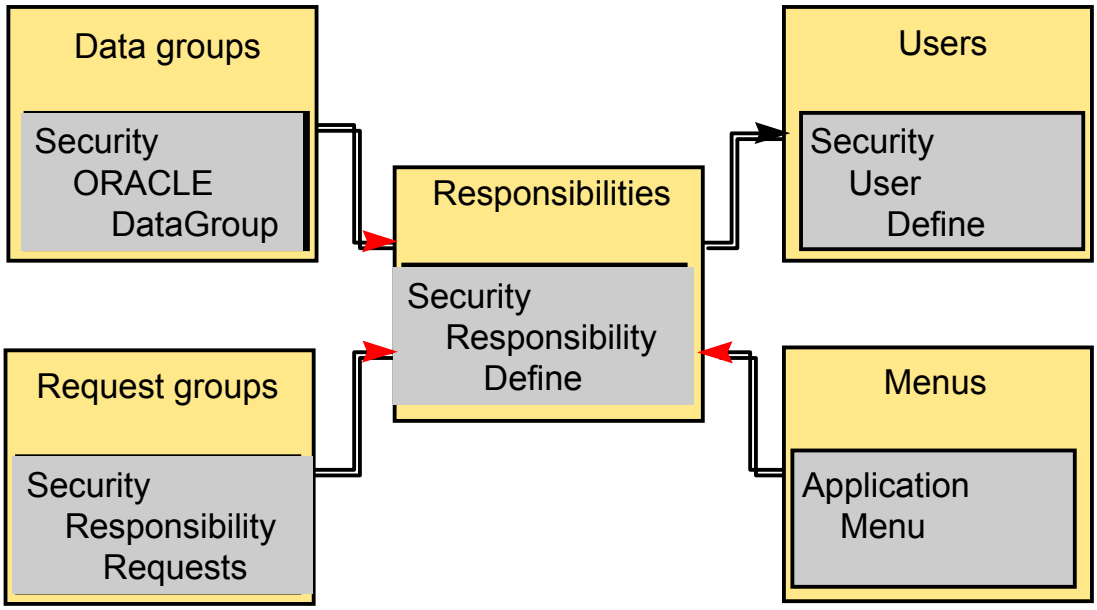
Defining New Responsibility



Defining a New Responsibility

- Assemble the components of application privileges to create a responsibility.
- Define the responsibility by assembling a menu, report security group, and data group and defining any function security (any menu or function exclusions).

Responsibility Creation Process



Responsibility Creation Process

There are five forms involved in the responsibility creation process. These are available under the System Administrator responsibility. This slide displays the navigation paths for these functions.

Defining New Responsibility

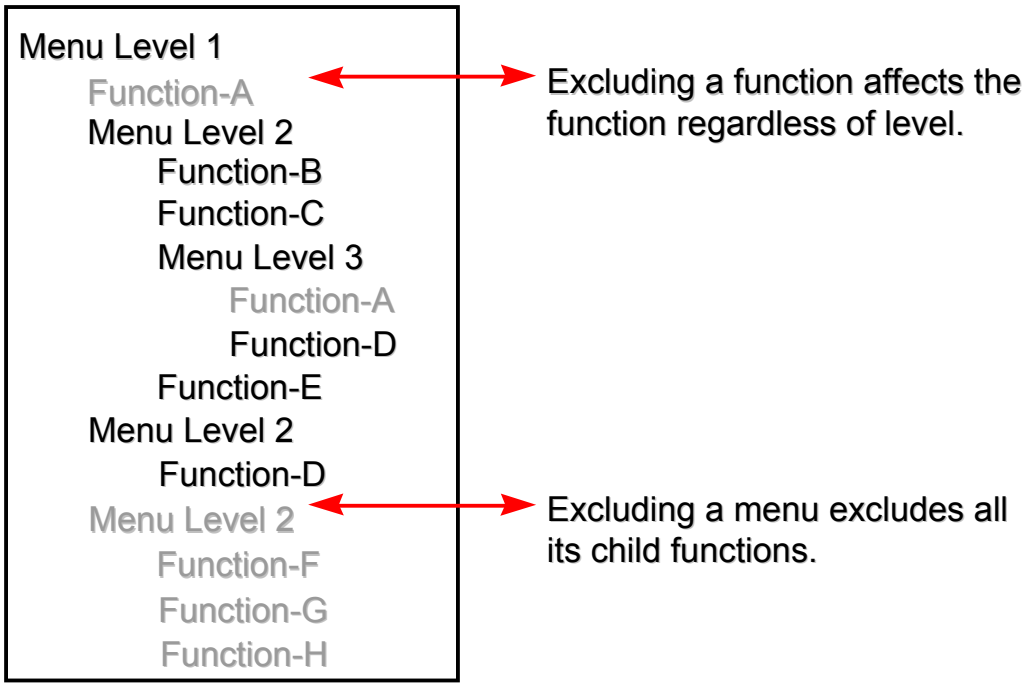
- You must Assign the following to your new responsibility:
  - A data group to supply the form,report,and program connect privileges
  - A menu to supply access to forms within an application
- You can assign the following
  - Any function or menu exclusions to control access to the functionality of the application
  - A report security group to control access to reports and concurrent programs

Defining a Responsibility

(N) Security > Responsibility > Define

A responsibility determines the type of applications a user accesses, which application functions a user can use, which reports and concurrent programs the user can run, and which data those reports and concurrent programs can access. Most responsibilities are available from Oracle Applications (the Forms navigator or the HTML-based E-Business Suite Home page), others may be accessible from Oracle Self-Service Web Applications or Oracle Mobile Applications.

Excluding Functions and Menus



Exclusion of Functions and Menus

Use exclusion rules to customize a responsibility. You can exclude functions at any level.

- When you exclude a menu item from a responsibility, all menus and functions that are nested in that menu are also excluded.
- When you exclude a function from a responsibility, all occurrences of that function throughout the menu structure of a responsibility are excluded.

Summary

You should be able to do the following

- Control access to functionality by defining menus
- Control access to functions and reports by defining responsibilities
- Control access to Oracle Applications by defining user accounts

ORACLE

Lesson Summary

Function Security provides you with the ability to control Oracle Applications Security. The system administrator specifies which users have access to which functions. User accounts provide the user access to an application. Responsibilities control that user's access to various functions of the application. Online access is controlled through the use of menus and menu exclusions. Report and concurrent program access is controlled through the use of request groups.

2

Application Object Library

Concurrent Programs

Objectives

After completing this course you should be able to:

- Identify an executable to Oracle Applications
- Define a concurrent program
- Specify concurrent program parameter information
- Define Custom Value Sets

ORACLE

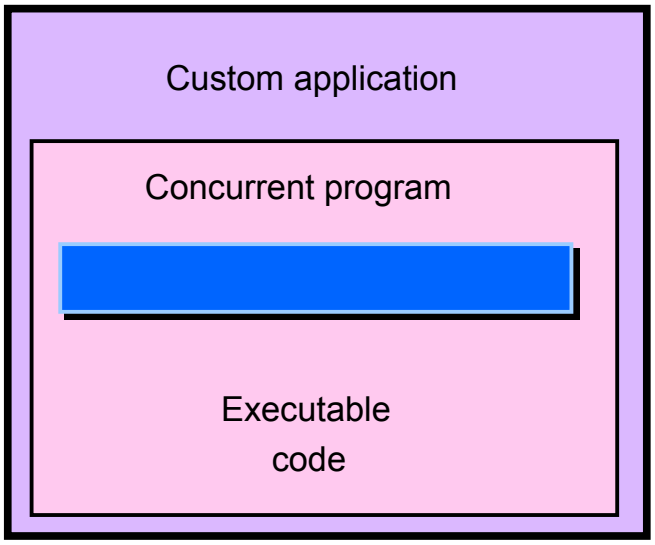
Objectives

Lesson Aim

Oracle Applications uses concurrent programs to process and report on data stored in Oracle Applications tables. These programs are secured, processed, and managed within the Oracle Applications environment, using the operations that you have seen in other lessons. However, you can take your custom applications and programs and integrate them into the Oracle Applications environment as well. In that environment they can be secured and managed the same way as regular Oracle Applications programs.

This lesson covers how to integrate user-developed programs into Oracle Applications.

Components of a Concurrent Program



Components of a Concurrent Program

You need to define several components of a concurrent program when adding your custom program to the Oracle Applications environment. A concurrent program consists of an executable module—for example, an Oracle Reports program or a PL/ SQL procedure—and the input parameters required by the program or procedure. A concurrent program must be associated with either an existing Oracle application or a custom application. In most cases, you will associate any custom concurrent programs with a registered custom application. This protects your concurrent programs during an upgrade.

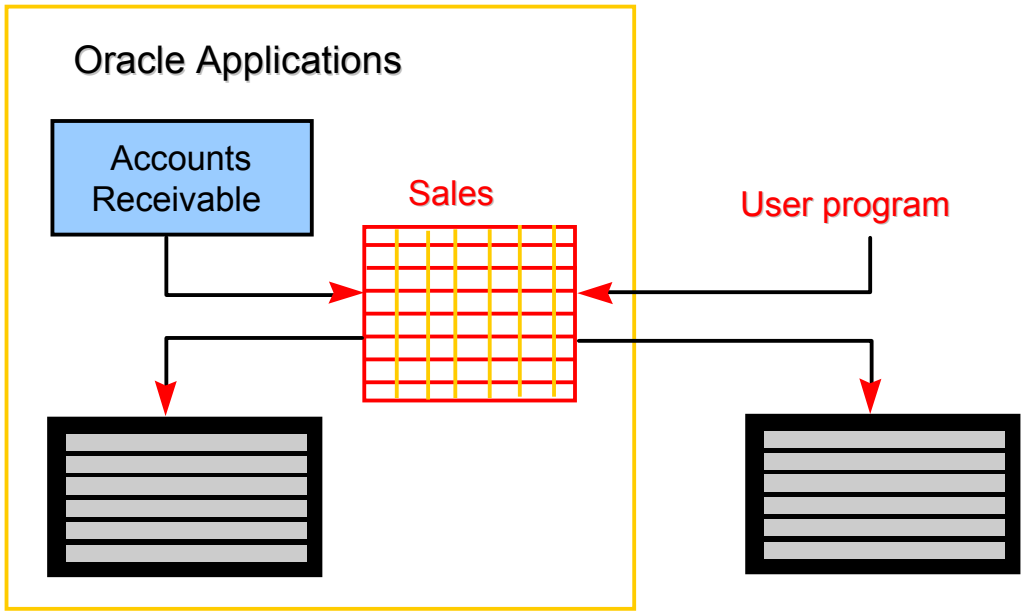
Concurrent Program Executable

A Concurrent program executable links an execution file or method used to execute it with defined concurrent program under concurrent processing. An Execution method can be written in standard language(pl/sql),a reporting tool or an operating system language.

Concurrent Program Definition

A concurrent Program is an instance of an executable file along with parameter and incompatibilities. Concurrent Programs use Concurrent Program Executable to locate the correct execution file

Using Custom Programs in Oracle Applications



Integrating Custom Programs with Oracle Applications

Although Oracle Applications provides you with a complete range of functionality, you may need to develop and implement your own custom programs to perform specialized processing. You can integrate such custom applications into the Oracle Applications environment so that they are processed like all your other Oracle Applications programs and reports.

For example, a sales table used by Oracle Receivables can also be accessed by a user program for special reporting needs. Such a user reporting program can be added to the Oracle Applications environment.

Assumptions for Incorporating a New Program

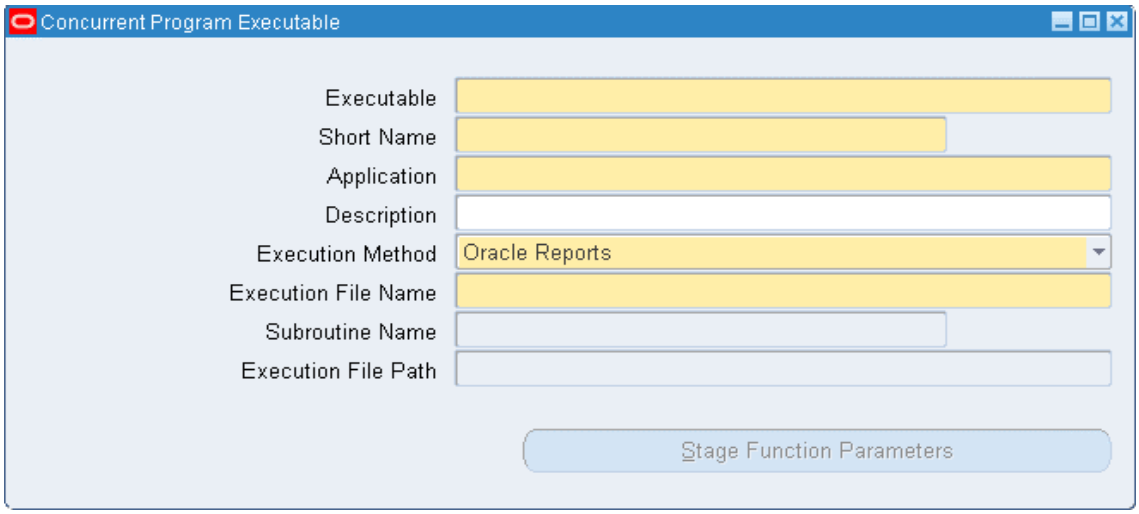
- The program has been completed
- The program parameters have been documented
- Any value sets needed already exist
- Tokens for Oracle Reports are defined

Adding a Custom Program to Oracle Applications

- Develop the program or report.
- Identify the program as an executable and register it with an application.
- Create a concurrent program containing the executable and its parameters.
- Add the concurrent program to a request group or set for processing.

Identifying the Executable

- The first step in adding a custom program to Oracle Applications is to identify the program or report as an executable.
- Use the Concurrent Program Executable window to identify your executable to Oracle Applications.



Identifying the Executable

Navigation

Concurrent > Program > Executable (Forms interface, using the System Administrator responsibility)

The Concurrent Program Executable window prompts for the following information:

- Executable – The name for your concurrent program executable. In the Concurrent Programs window you assign this name to a concurrent program to associate your concurrent program with your executable logic.
- Short Name – The short name for your concurrent program executable.
- Application – The concurrent managers use the specified application to determine in which directory structure to look for your execution file.
- Execution Method – Specifies the type of program, such as a PL/SQL procedure or an Oracle Reports program. The execution method cannot be changed once the concurrent program executable has been assigned to one or more concurrent programs in the Concurrent Programs window.
- Execution File Name – The operating system name of your execution file.
- Subroutine Name – The name of your C or Pro*C program subroutine. Only immediate programs or spawned programs using the Unified C API use this field.

Identifying the Executable (continued)

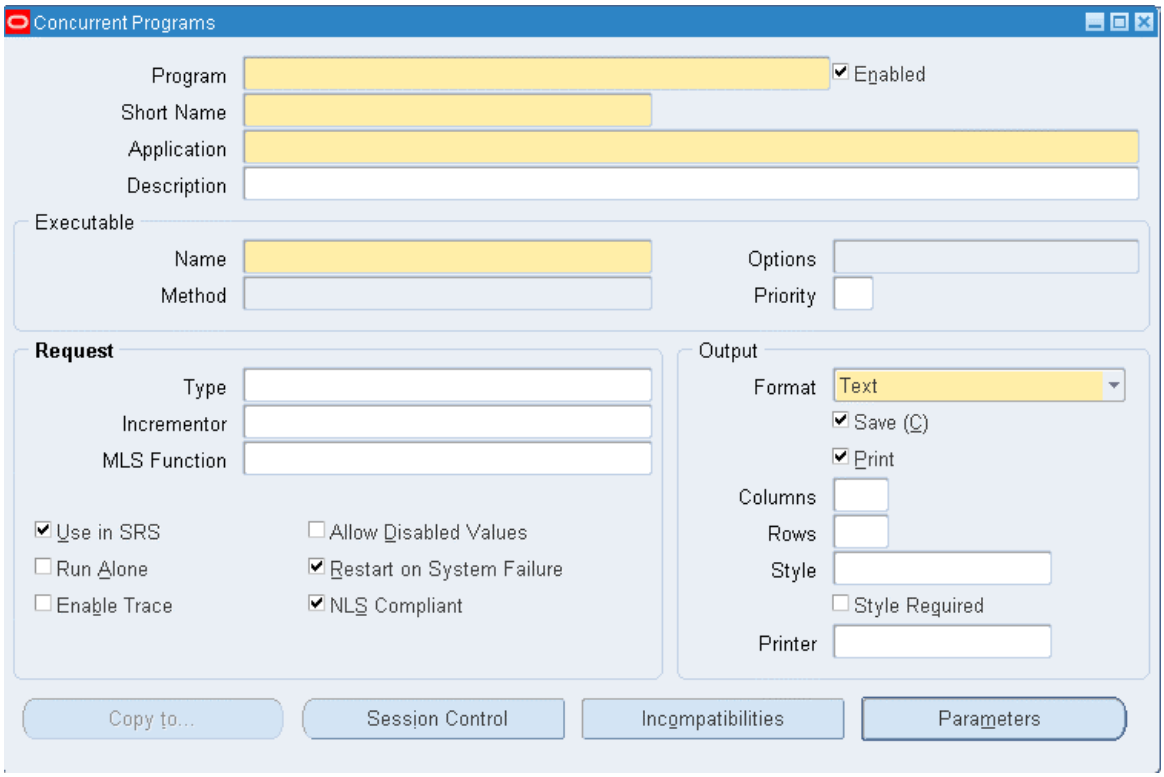
Note: Defining new immediate concurrent programs is not recommended. Use either a PL/SQL stored procedure or a spawned C program instead.

- Execution File Path
- Stage Function Parameters button – Opens a window that allows you to enter parameters for the Request set Stage Function. This button is only enabled when you select Request Set Stage Function as you Execution Method.

For more information see:

(Help) Applied Technology > Oracle Applications System Administration > Configuration > Overview of Concurrent Programs and Requests > Concurrent Program Executable Window

Concurrent Program Definition Window



Creating the Concurrent Program

- Use the Concurrent Programs window to define the details about your concurrent program.
- Use the Concurrent Programs Parameters window to enter and update the program parameters that you want to pass to the program executable.
- The program parameters defined here should match the variables in your executable.

Concurrent Programs Window Fields

(N) Concurrent > Program > Define (Forms interface, using the System Administrator responsibility)

The Concurrent Program window prompts for the following information:

- Program – A descriptive name for your concurrent program. This is the name you see when you view your requests in the Requests window. If this concurrent program runs through Standard Request Submission, you see this name in the Submit Requests window.
- Short Name – A short name that Oracle Applications can use to associate your program with a concurrent program executable.
- Application – The program's application determines what ORACLE username your program runs in and where to place the log and output files.
- Enabled – Indicate whether users will be able to submit requests to run this program and the concurrent managers will be able to run your program. Disabled programs do not display in users' lists, and do not appear in any concurrent manager queues.
- Executable: Name – The concurrent program executable for your program. The executable is defined using the Concurrent Program Executable window.

Concurrent Programs Window Fields (continued)

- Executable: Method – The execution method used by your concurrent program displays in this field.
- Executable: Options – Contains options for SQL*Report programs.
- Executable: Priority – Priority for this program. If you do not assign a priority, the user's profile option Concurrent:Priority sets the request's priority at submission time.
- Request: Type – Certain concurrent managers are specialized to run only certain request types.
- MLS Function – The MLS function used by the program (if applicable). The Multilingual Concurrent Request feature allows a user to submit a request once to be run multiple tomes, each time in a different language. If this program utilizes this feature the MLS function determines which installed languages are needed for the request.
- Use in SRS – Indicates whether this program can be run using Standard Request Submission. If this box is checked, you must register the parameters of this program.
- Allow Disabled Values – For a program authorized for SRS submission, check this box to allow a user to enter disabled or outdated parameter values.
- Run Alone – Check this box if your program is incompatible with other programs in its logical database and should therefore be run alone.
- Restart on System Failure – Check this box if you want this program to be restarted automatically when the concurrent manager is restored after a system failure.
- Enable Trace – turns on SQL tracing when program runs.
- NLS Compliant – Check this box if the program allows a request to reflect a language and territory that are different from the language and territory that the user is operating in. This option should be set only by the developer of the program. The program must be
- Output Fields – Specify the handling or output from executions of this program.
- Concurrent Programs Window Buttons
- Copy to… - Choose this button to create another concurrent program using the same executable, request, and report information.
- Session Control – Specify options for the database session of the concurrent program when it is executed. These options are used to optimize system performance.
- Incompatibilities – Opens the Incompatible Programs window to define which programs should not be run when this program is running.
- Parameters – Opens the Program Parameters window.

Associating a Program with a Request Group

- For a concurrent program to be accessible and eligible for submission, you must define it to a request group.
- Use the Request Groups window to assign to qualify your program.

Associate a Program with a Request Group

(N) Security > Responsibility > Request

For the definition of the fields in this window see

(Help) Applied Technology > Oracle Applications System Administration > Request Group Window

Submitting the Concurrent Program

Use the Standard Request Submission window to execute the program:

(N) Requests > Run

Summary

You should now be able to:

- Identify an executable to Oracle Applications
- Define a concurrent program
- Specify concurrent program parameter information

Practice #1 – Without Parameter

Overview:

This practice covers the following topics:

- Create a Package
- Create Executable
- Create Concurrent Program Definition
- Add the custom concurrent program to request group
- Test your program

ORACLE

Practice 1

Create package sample 1 as follows

Package Specification

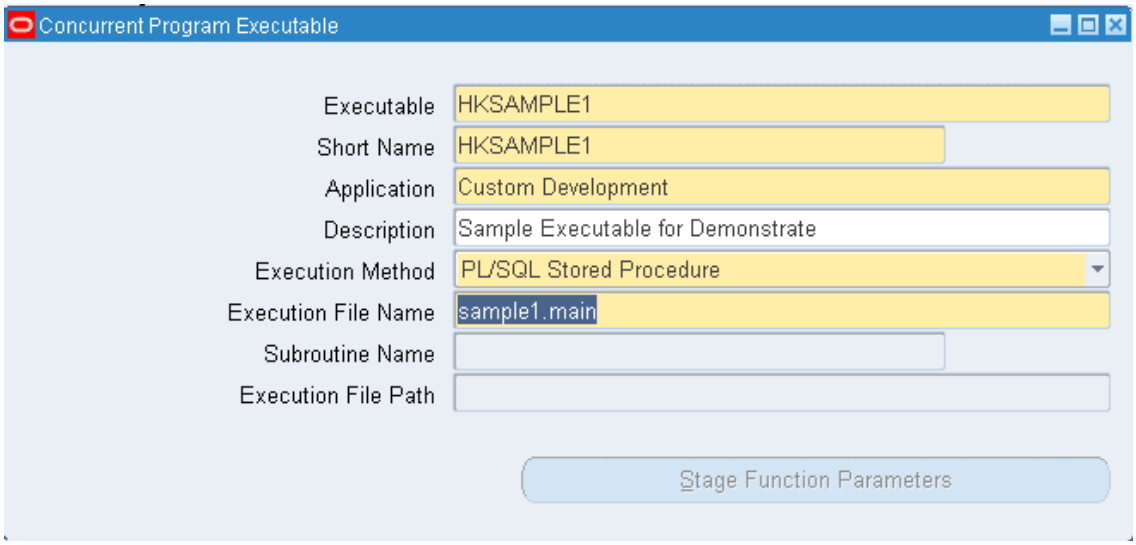
```
CREATE OR REPLACE PACKAGE sample1
AS
PROCEDURE main
 (
   p_errbuf OUT VARCHAR2,
   p_retcode OUT NUMBER );
END sample1;
```

Package Body

```
CREATE OR REPLACE PACKAGE BODY sample1
AS
PROCEDURE main
 (
   p_errbuf OUT VARCHAR2,
   p_retcode OUT NUMBER )
IS
BEGIN
 fnd_file.put_line(fnd_file.log,'Welcome to Oracle Application');
 fnd_file.put_line(fnd_file.output,'Welcome to Oracle Application');
END;
END sample1;
```

Note: Always Create custom objects with prefix your initials to maintain uniformity

Executable Definition



Executable Definition

Navigation

System Administrator→Concurrent→Program→Executable

Executable : HKSAMPLE1

Short Name : HKSAMPLE1

Application : Custom Development

Description : Sample Executable for Demonstrate

Execution Method : PL/SQL Stored Procedure

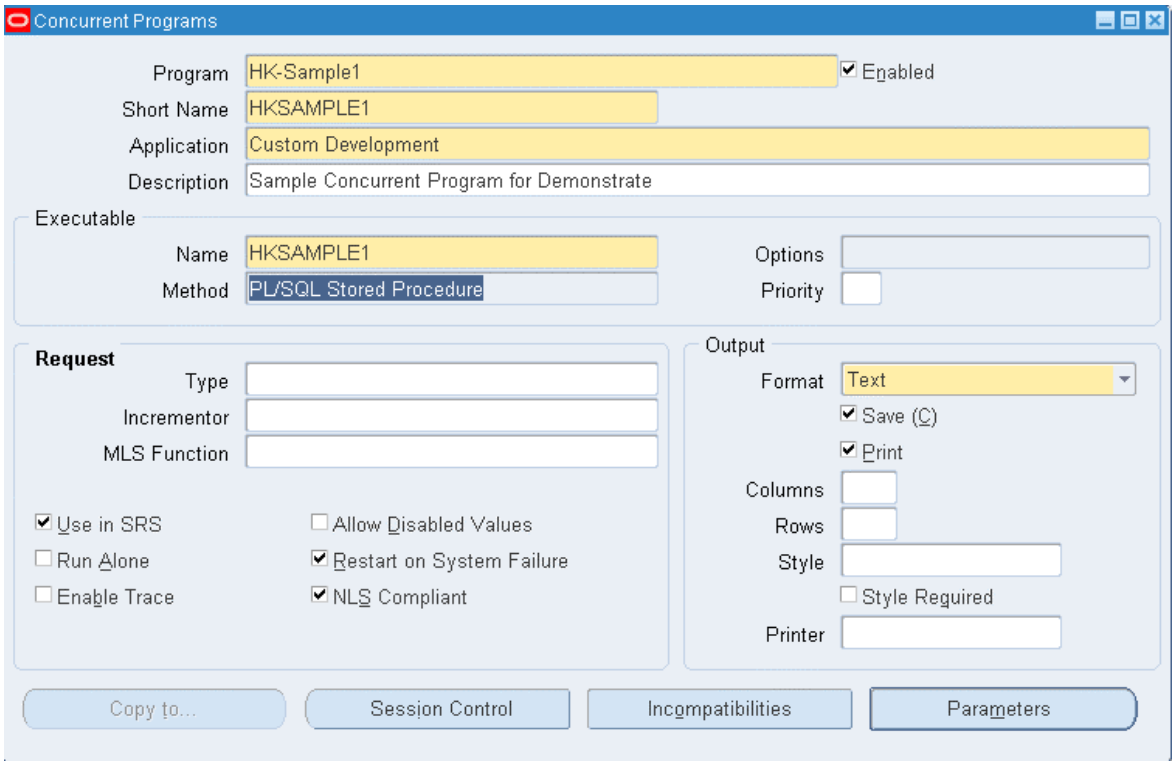
Execution File Name : sample1.Main

Save the form

Note:

Sample1.main →Database package name with procedure name in it.

Concurrent Program Definition



Concurrent Program Definition

Navigation

System Administrator→Concurrent→Program→Define

Program : HK-Sample1

Short Name : HKSample1

Application :Custom Development

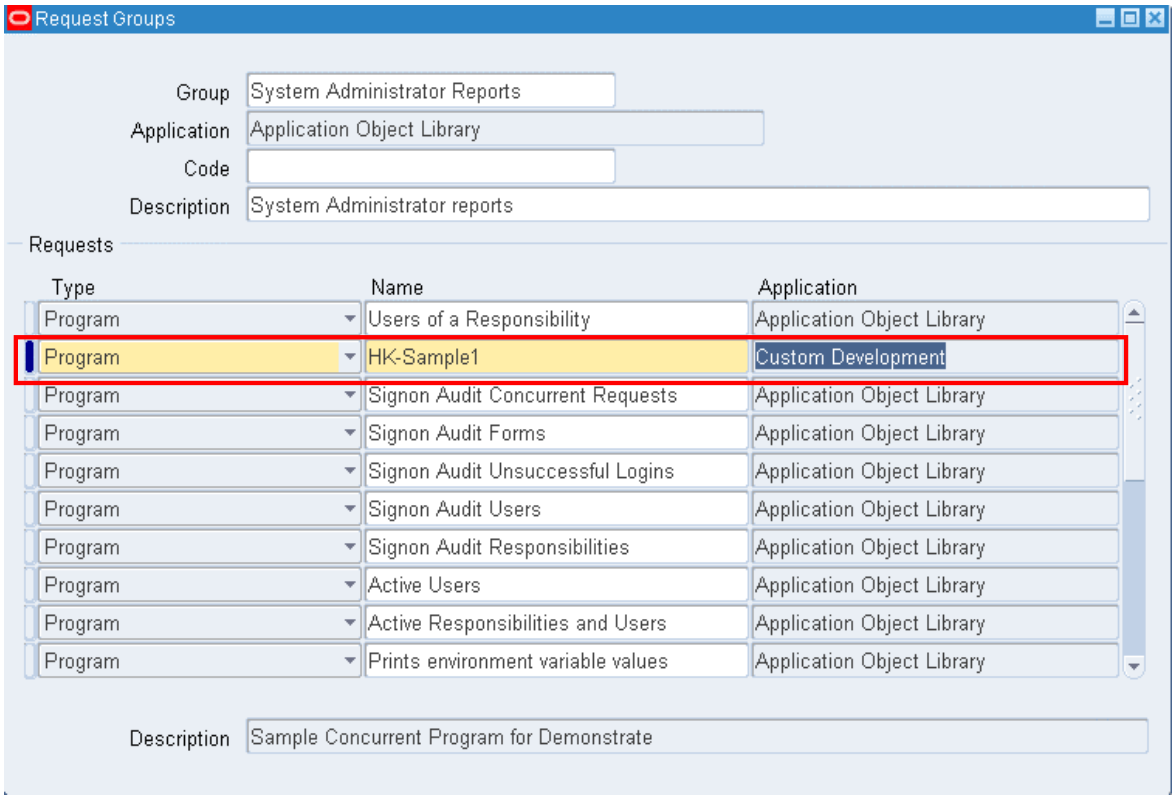
Description : Sample Concurrent Program for Demonstrate

In Executable Block:

Name : HKSAMPLE1 (Is the Executable Name which is define in the last slide)

Save the form

Associating a Program with a Request Group



Adding to Request Group

Navigation

System Administrator→Security→Responsibility→Request

Query for *System Administrator Reports* Request Group

(To query for the request Group

Press F4

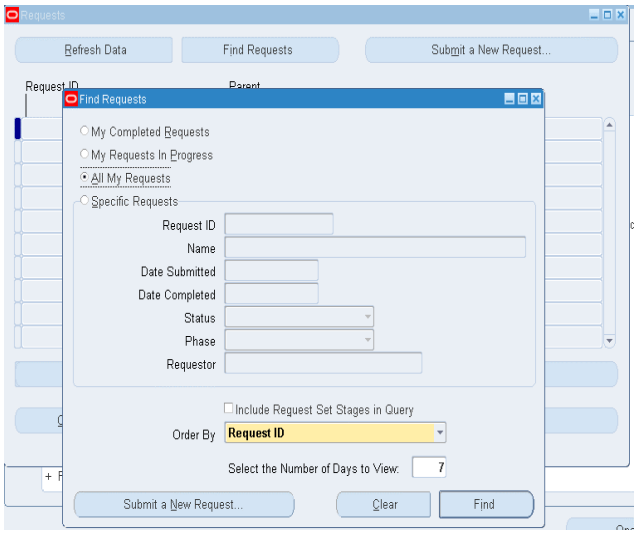
Enter 'System Administrator Reports' in Group Filed

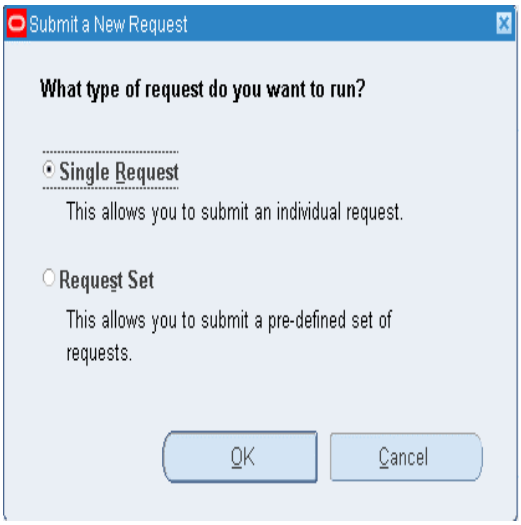
Press F11)

Add the Custom Concurrent Program which you created

Save the work

Submit New Request





ORACLE

Submit New Request

From the menu:

View→Request

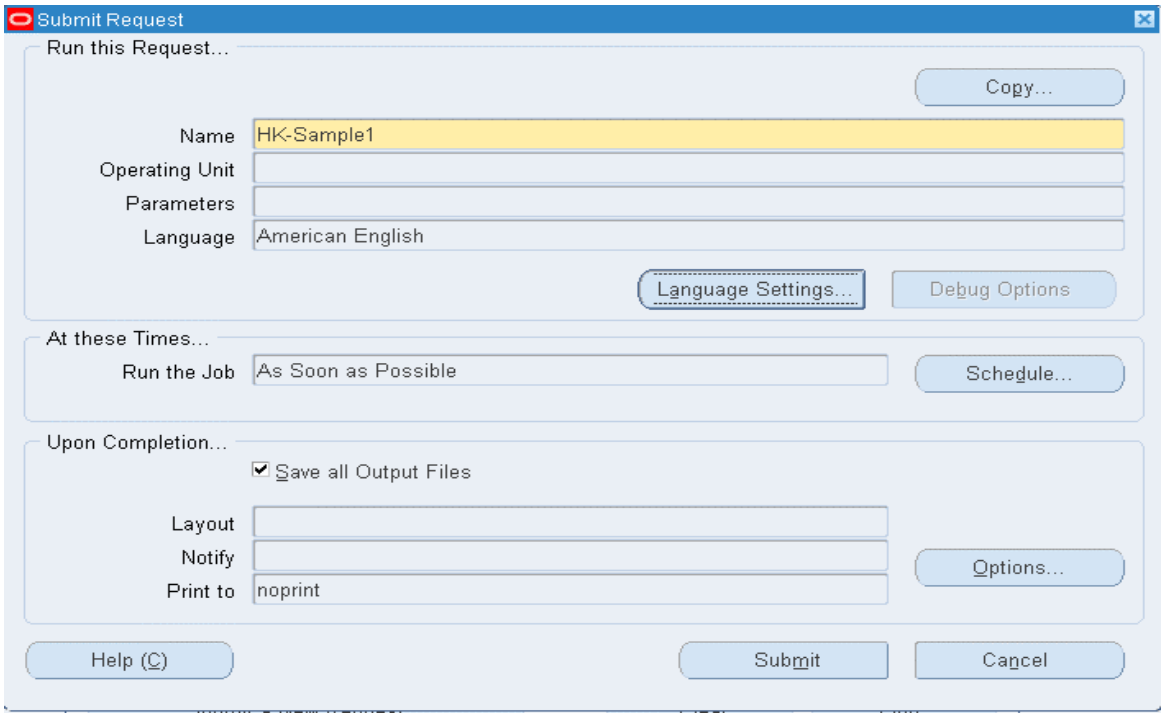
Select the Submit a New Request

New Form will popup

Select Single Request Radio button

Select OK

Submit New Request



ORACLE

Adding to Request Group

In the Name Field Select your Custom Concurrent Program

Click on Submit Button

Then go to Find request Form

Highlight your request

Select View output button

A browser will pop up with the message containing “Welcome to Oracle Application “

This the end the of the Practice #1

Value Set

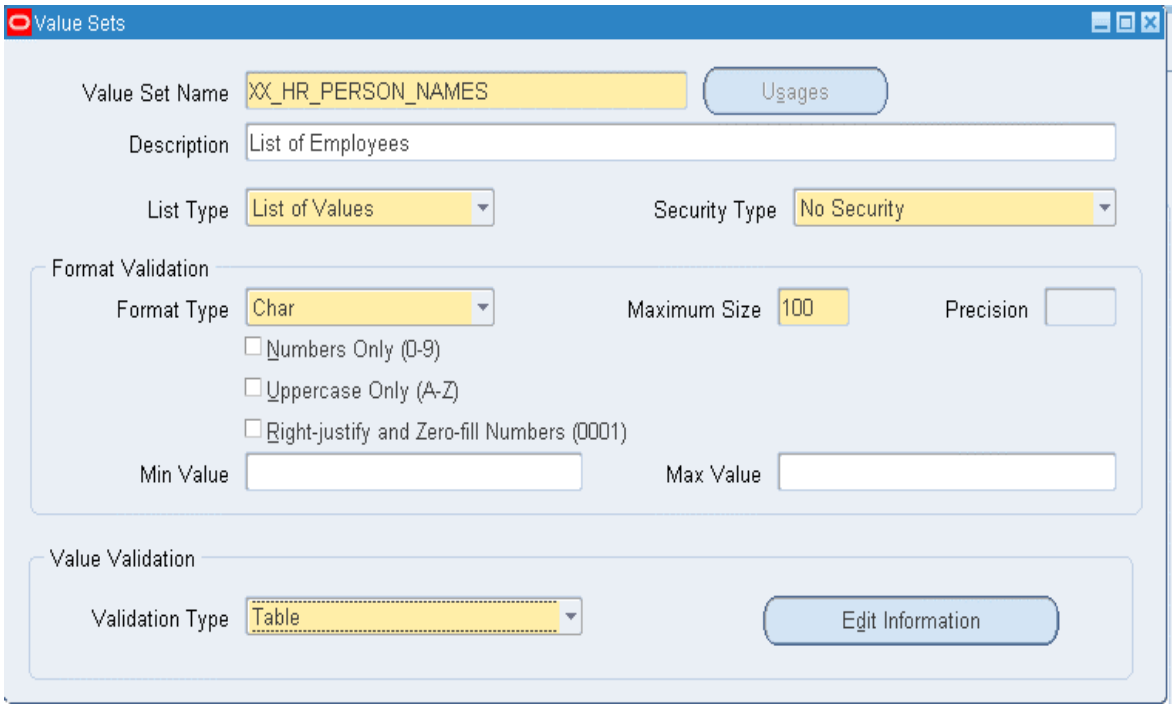
- Value sets are definitions of allowable values for use with a flexfield segment or report parameter.
- This lesson covers how to create new table type value set
- In the below example our aim is to create distinct employee name

ORACLE

Define value sets

This lesson will demonstrate how to create a table type value set and later will be used in custom concurrent program

Define Value Set



Define Value Set

Navigation

System Administrator→Application→Validation→Set

Value Set Name :XX_HR_PERSON_NAMES

Description :List of Employees

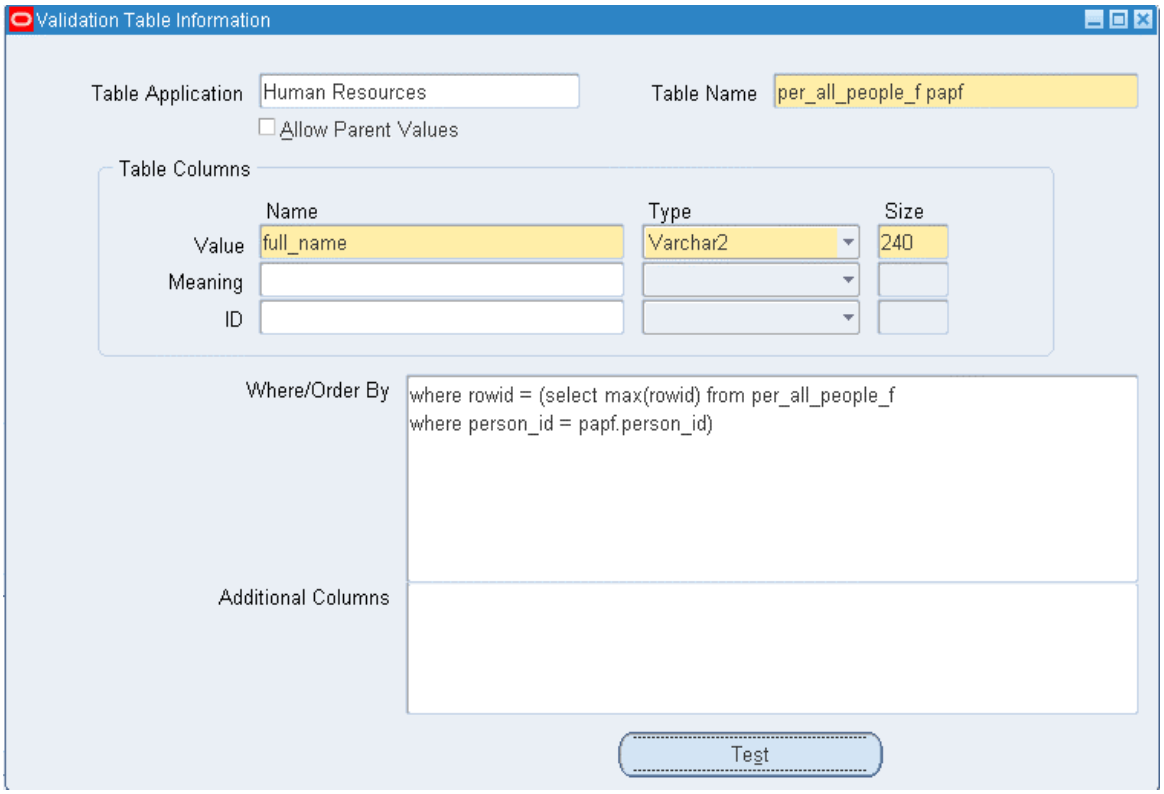
Format Type : Char

Maximum Size :100

Validation Type :Table

Now click Edit Information Button

Define Value Set



ORACLE

Define Value Set (continued)

Table Application : Human Resources

Table Name : per_all_people_f papf

Value :

Name : full_name

Type : Varchar2

Size : 240

Where /Order By :

where rowid = (select max(rowid) from per_all_people_f

where person_id = papf.person_id)

To validate the sql query press Test Button.

After validating the sql query save the form

Practice #2 – With Parameters

Overview:

This practice covers the following topics:

- Create a Package
- Create Executable
- Create Concurrent Program Definition
- Add Parameters to the Concurrent Program
- Add the custom concurrent program to request group
- Test your program

ORACLE

Practice 2

Create package sample 2 as follows

Package Specification

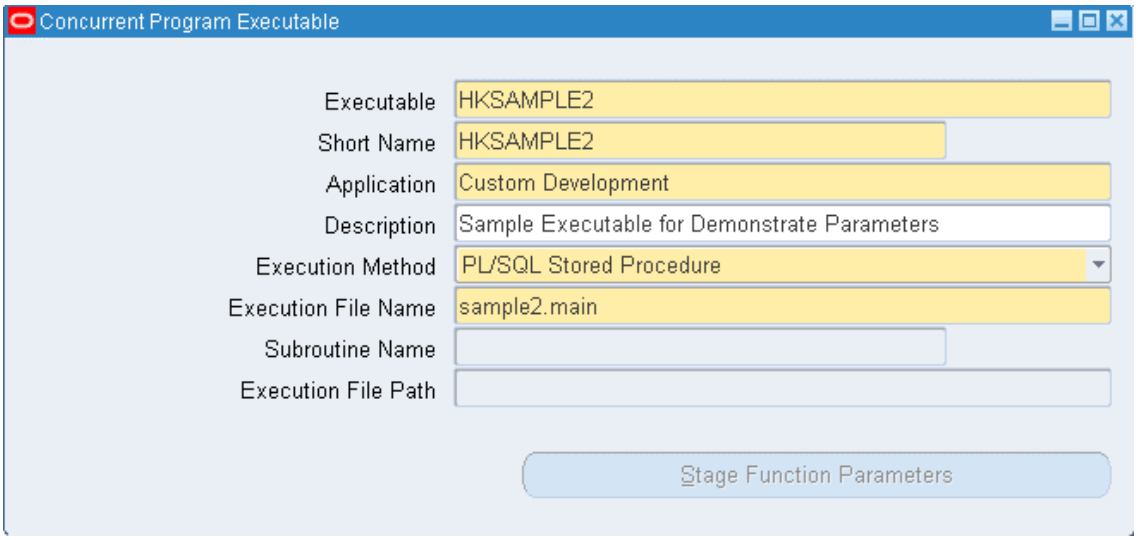
```
CREATE OR REPLACE PACKAGE sample2
AS
PROCEDURE main(
  p_errbuf OUT VARCHAR2,
   p_retcode OUT NUMBER,
   p_name in varchar2);
END sample2;
```

Package Body

```
CREATE OR REPLACE PACKAGE BODY sample2
AS
PROCEDURE main
 (
   p_errbuf OUT VARCHAR2,
   p_retcode OUT NUMBER,
   p_name in varchar2
   )
IS
BEGIN
 fnd_file.put_line(fnd_file.output,'Hello '||p_name);
END;
END sample2;
```

Note: Always Create custom objects with prefix your initials to maintain uniformity

Executable Definition



Executable Definition

Navigation

System Administrator→Concurrent→Program→Executable

Executable : HKSAMPLE2

Short Name : HKSAMPLE2

Application : Custom Development

Description : Sample Executable for Demonstrate Parameters

Execution Method : PL/SQL Stored Procedure

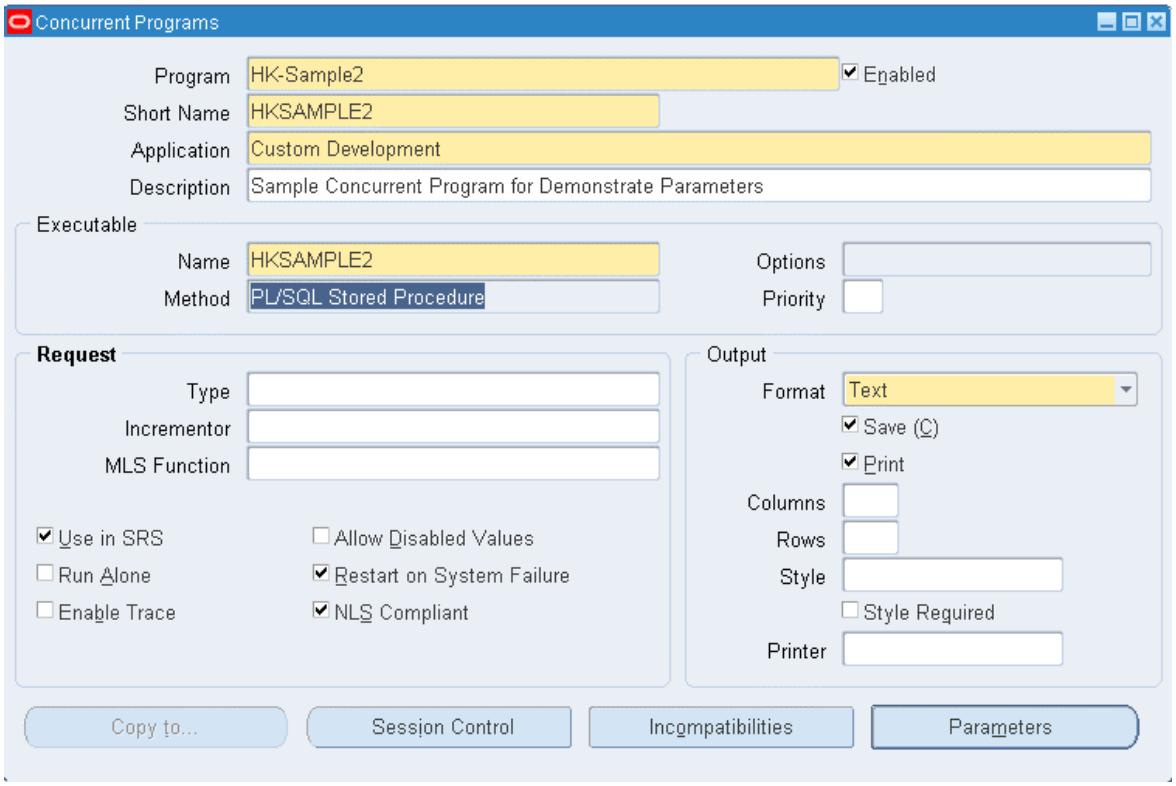
Execution File Name : sample2.Main

Save the form

Note:

Sample2.main →Database package name with procedure name in it.

Concurrent Program Definition



Concurrent Program Definition

Navigation

System Administrator→Concurrent→Program→Define

Program : HK-Sample2

Short Name : HKSample2

Application :Custom Development

Description : Sample Concurrent Program for Demonstrate Parameters

In Executable Block:

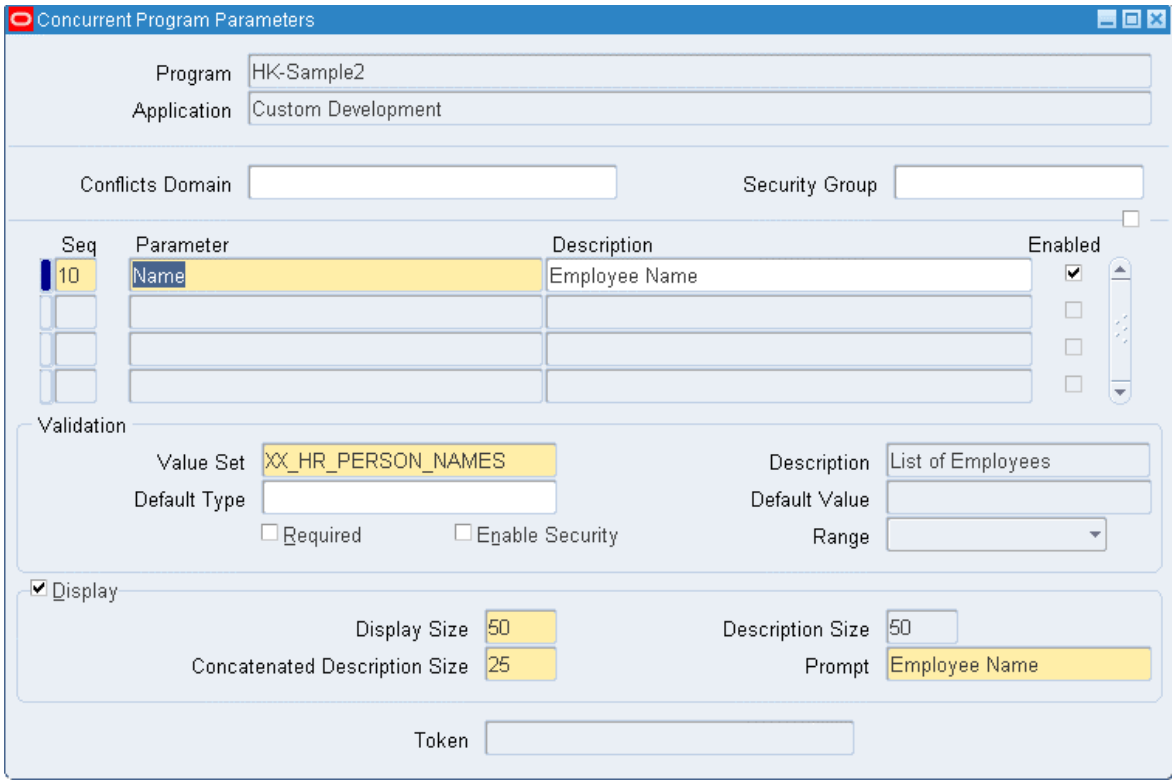
Name : HKSAMPLE2 (Is the Executable Name which is define in the last slide)

Save the form

To Enter Parameters Details

Press *Parameter* Button to open Parameter Form

Parameter 1



ORACLE

Parameter 1

Seq : 10

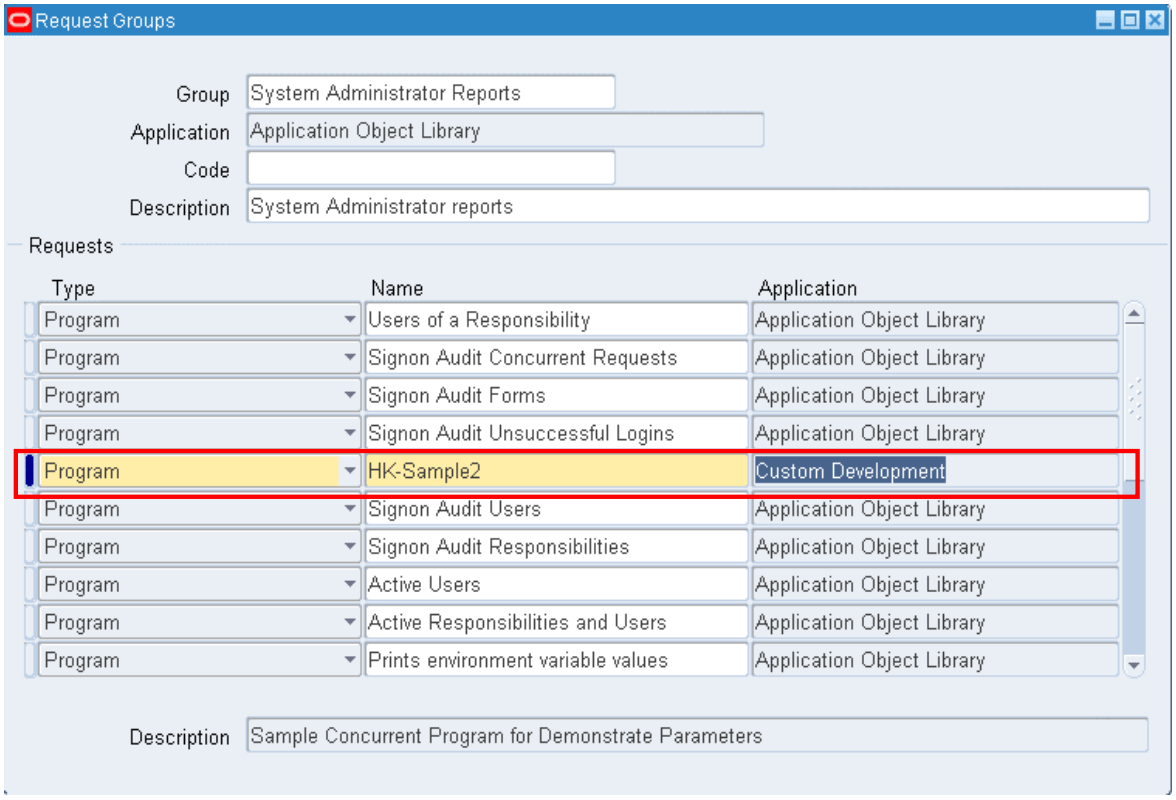
Parameter : Name

Description : Employee Name

Value Set : XX_HR_PERSON_NAME (Which is created in the last lesson)

Prompt : Employee Name

Associating a Program with a Request Group



ORACLE

Adding to Request Group

Navigation

System Administrator→Security→Responsibility→Request

Query for *System Administrator Reports* Request Group

(To query for the request Group

Press F4

Enter 'System Administrator Reports' in Group Filed

Press F11)

Add the Custom Concurrent Program which you created

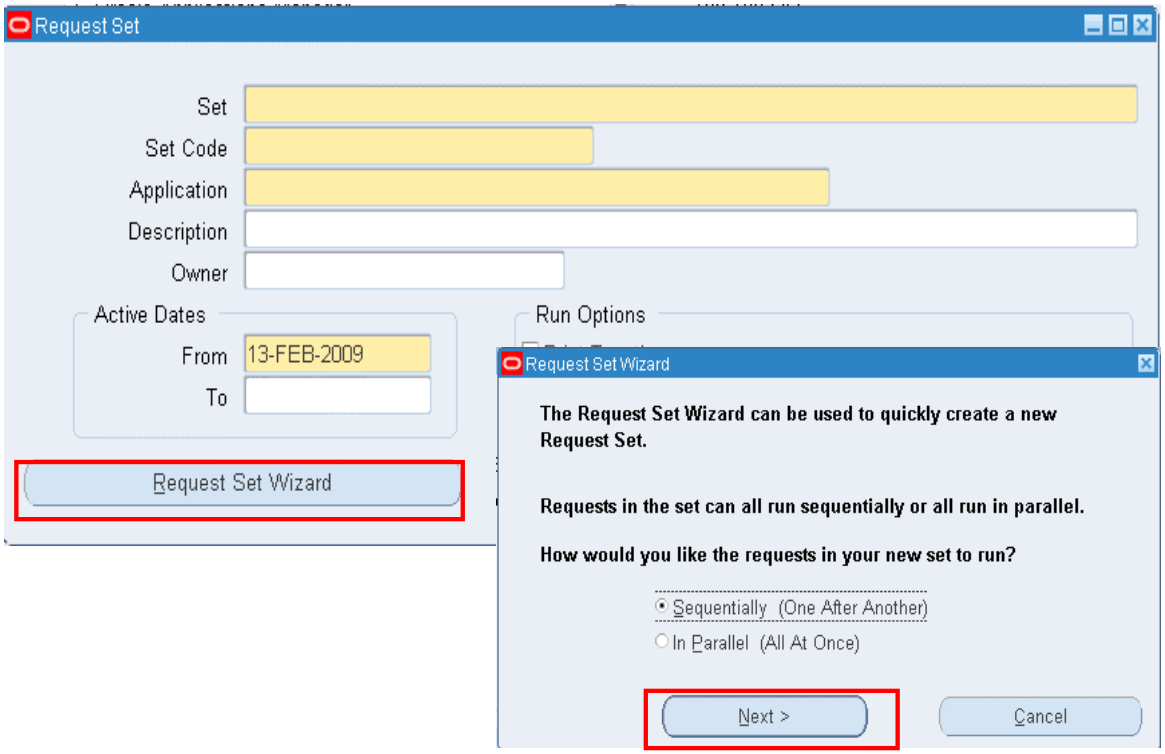
Save the work

Test your program like Practice 1

Defining Request Set

- Request sets are a quick and convenient way to run several concurrent programs with predefined print options and parameter values.
- By defining request sets, you can submit the same set of requests regularly using a single transaction.
- Use the Request Set wizard to create simple request sets.

Defining Request Set Using Wizard



ORACLE

Defining Request Set Using Wizard

We will define Request using wizard and add two custom concurrent program that we have created for practice 1 and practice 2

Navigation

System Administrator→Concurrent→Set

Press Request *Set Wizard* to start wizard

Will popup one more window (Request Set Wizard)

Select *Sequentially (One After Another)* Option

Press OK Button

Defining Request Set Using Wizard

Fig 1

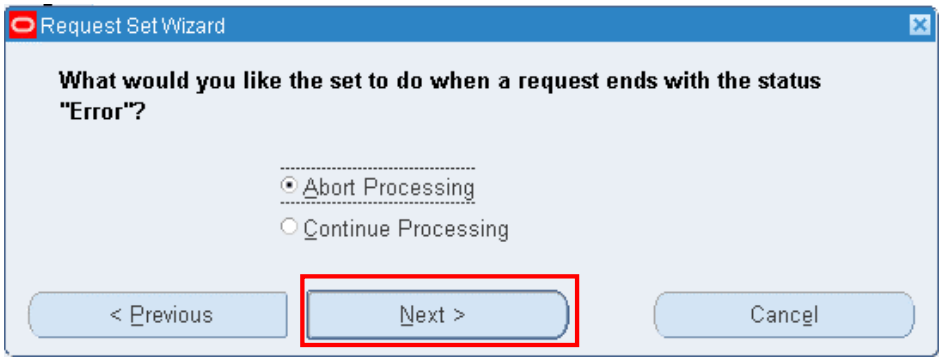
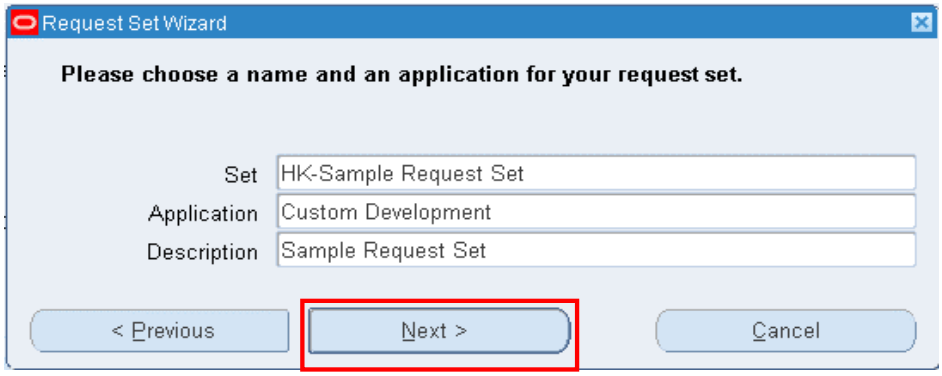


Fig 2



ORACLE

Defining Request Set Using Wizard (continued)

Will Popup a window as shown in Fig 1

Select Abort Processing Option

Press Next Button

Will popup a window as shown in fig 2

Enter the following

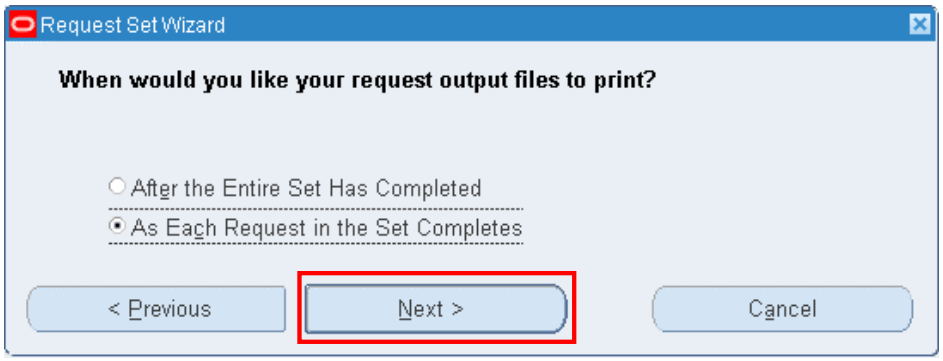
Set : HK-Sample Request Set

Application : Custom Development

Description : Sample Request Set

Press Next Button

Defining Request Set Using Wizard

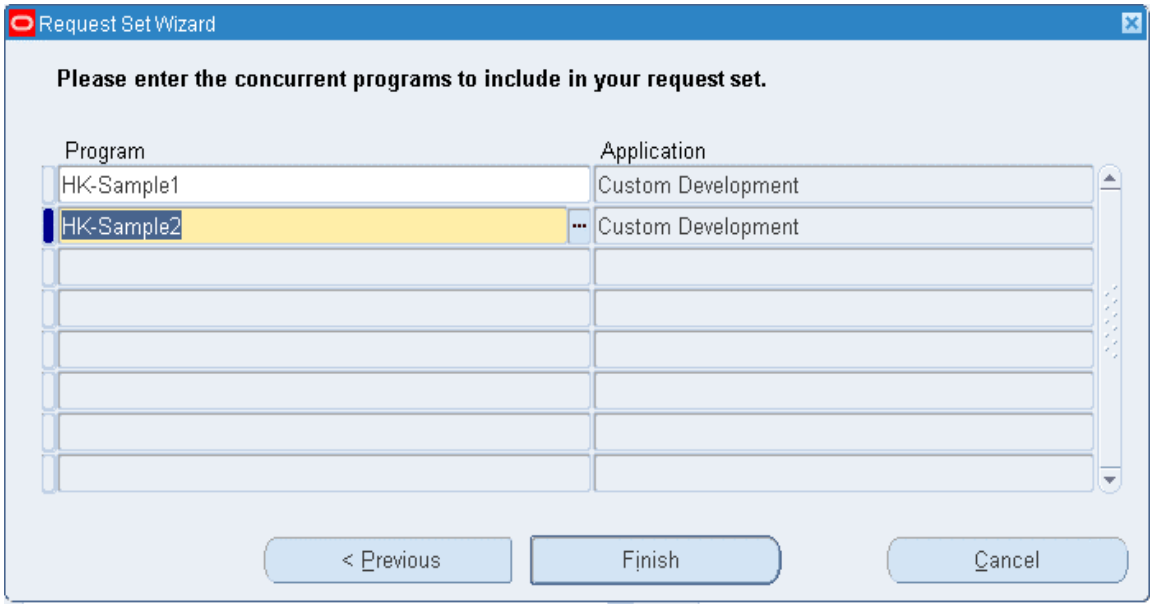


Defining Request Set Using Wizard (continued)

Select As Each Request in the Set Completed

And press Next to proceed

Defining Request Set Using Wizard



Defining Request Set Using Wizard (continued)

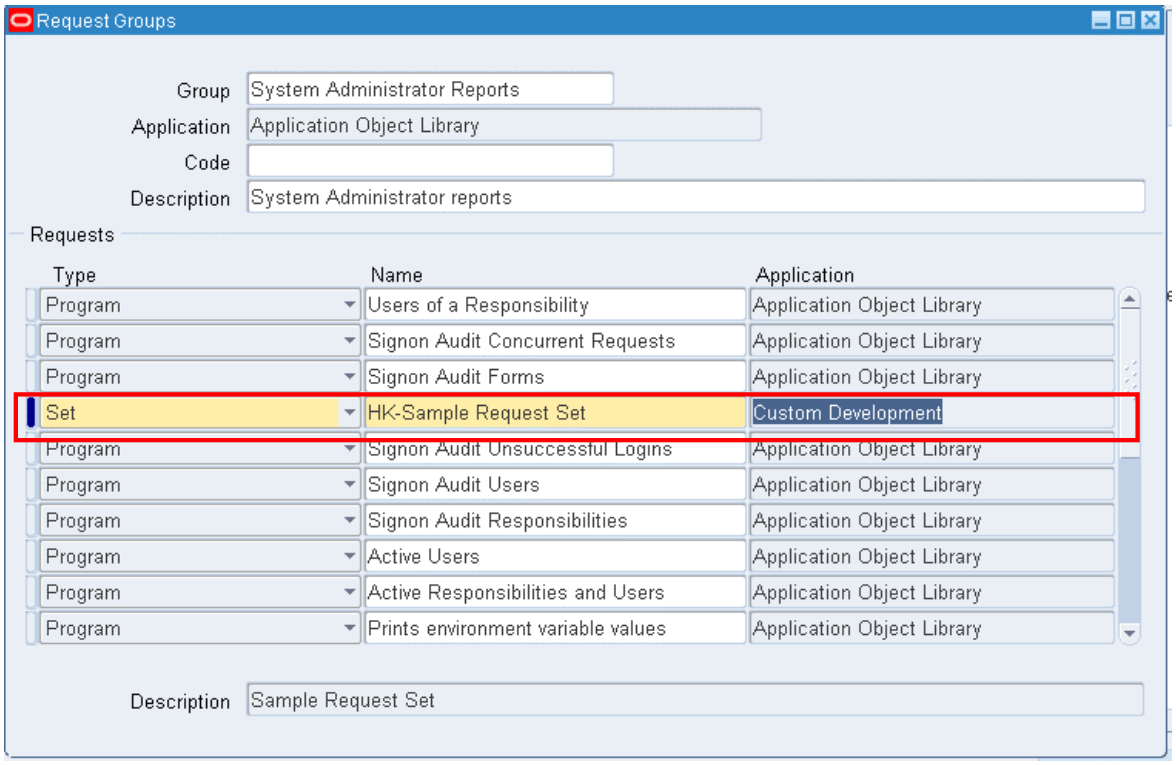
In this form add two concurrent Program that already created

HK-Sample1

HK-Sample2

Press Finish to Finish the Request Set wizard

Associating Request with a Request Group



Adding to Request Group

Navigation

System Administrator→Security→Responsibility→Request

Query for *System Administrator Reports* Request Group

(To query for the request Group

Press F4

Enter 'System Administrator Reports' in Group Filed

Press F11)

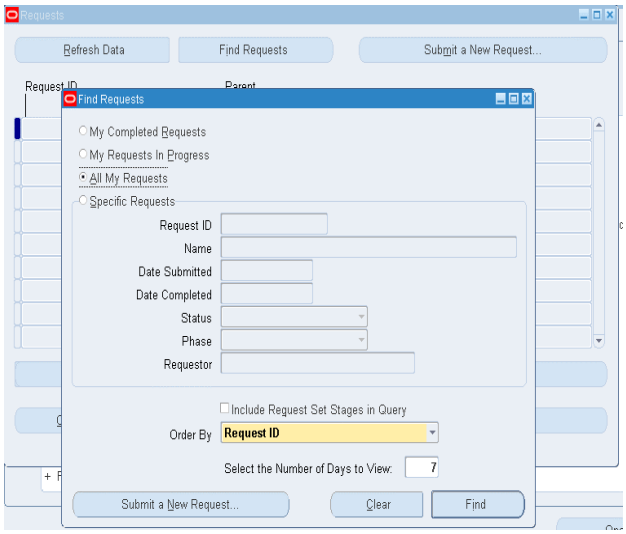
Select Set in the type field

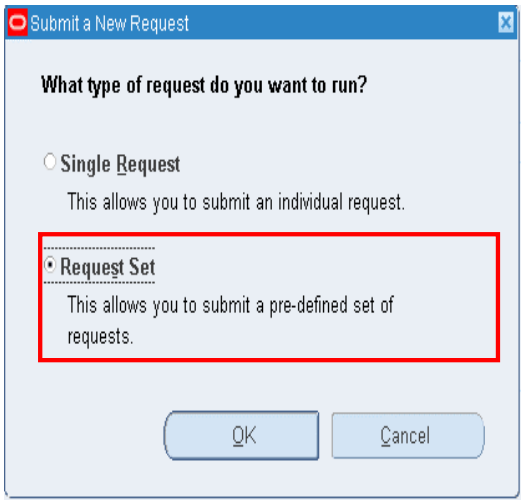
Add the Custom Request set which you created

Save the work

Test your program as of Practice 1

Submit New Request Set





ORACLE

Submit New Request

From the menu:

View→Request

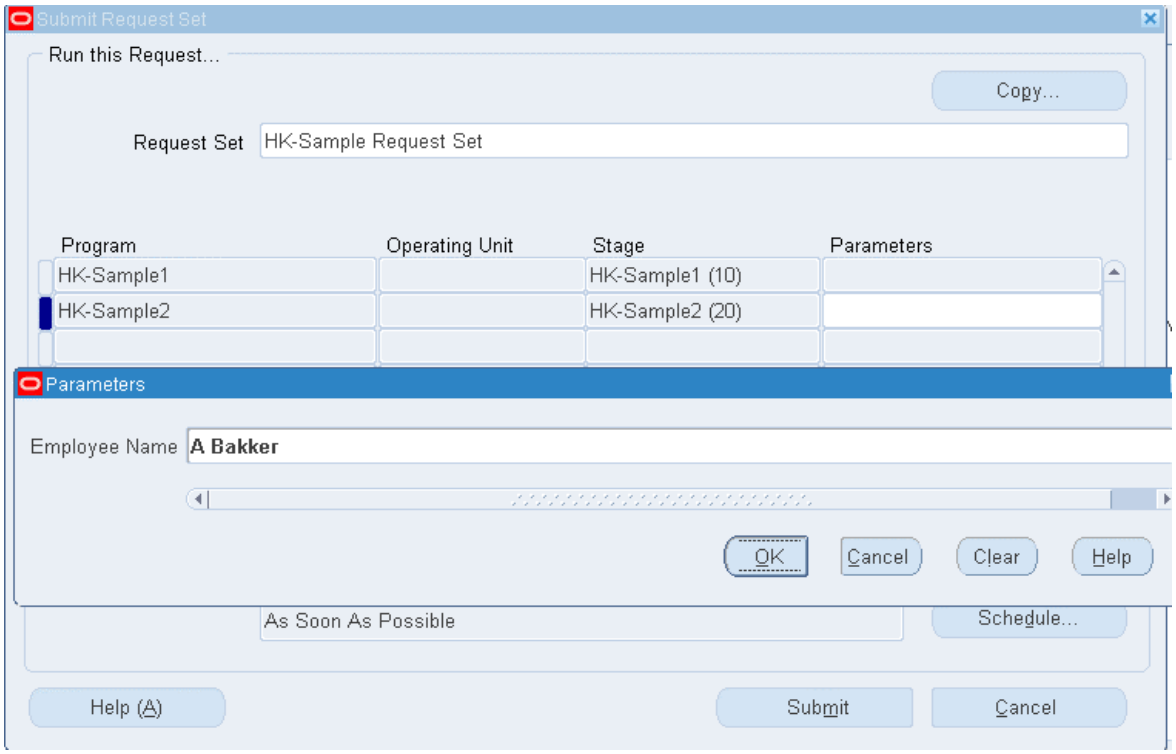
Select the Submit a New Request

New Form will popup

Select Request Set Radio button

Select OK

Submit New Request Set



Submit New Request Set (continued)

In the Request Set Field

Enter HK-Sample Request Set

Click on filed Parameter of Second request

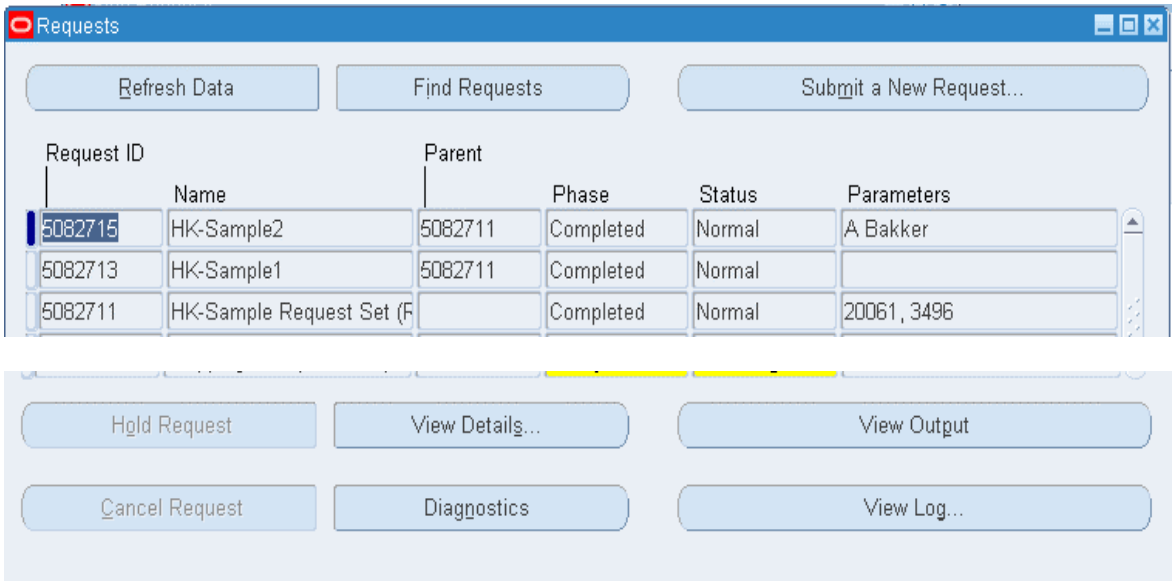
Will pop up the parameter window

Select value from LOV

Select OK

And Submit the Request Set

Submit New Request Set



ORACLE

Submit New Request Set (continued)

In the Concurrent Request form,you can find 3 concurrent request

In the above screen the request No: 5082711 is the parent request for other two request 5082713 and 5082715

You can the output of the each Concurrent Program by opening View output window

Summary

Now you able to do following:

Identify an executable to Oracle Applications

- Define a concurrent program
- Specify concurrent program parameter information
- Define Custom Value Sets

ORACLE

3

Application Object Library

Report Registration

Objectives

After completing this chapter you should be able to:

- Identify the application top to keep rdf file
- Define executable
- Define concurrent program
- Attaching concurrent program to request group

ORACLE

Objectives

Lesson Aim

In this lesson we will how to register the oracle reports into Oracle application

Register the report to oracle application

After completion of development of the report,ftp the rdf file into server to the following location of the custom_top

Assumption

This Chapter assumes that you have already developed the report using Oracle reports tool as per the Oracle application standard

File Location

Custom reports to be placed in the application

The Developed report to be placed under the following directory

$Custom_top/reports/US/

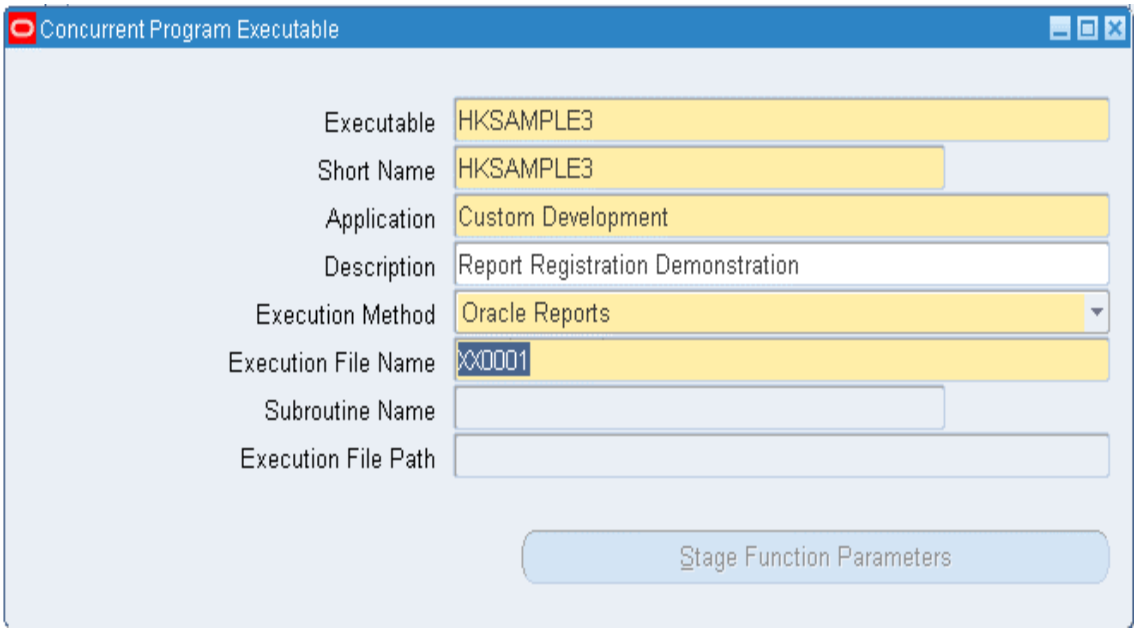
File location

For the demonstrate purpose consider the following

XX0001.rdf

Application: Custom Development

Executable Definition



Executable Definition

Navigation

System Administrator→Concurrent→Program→Executable

Executable : HKSAMPLE3

Short Name : HKSAMPLE3

Application : Custom Development

Description : Report Registration Demonstration

Execution Method : Oracle Report

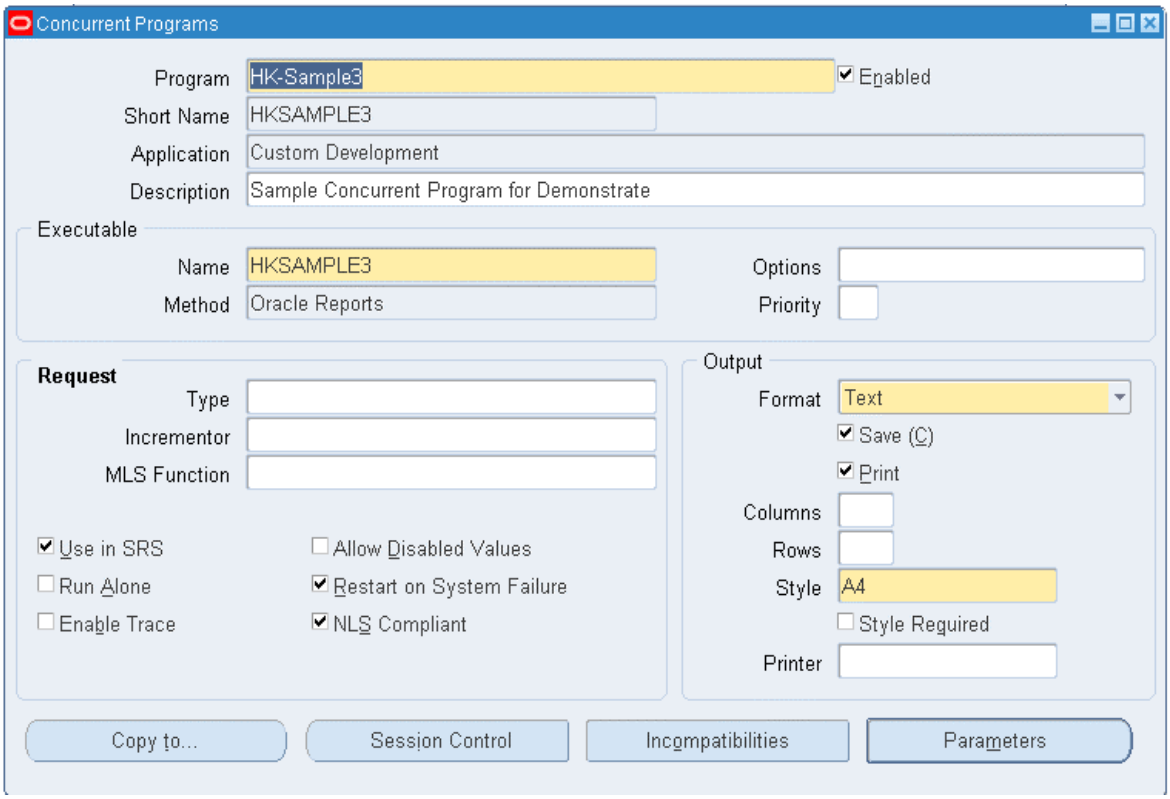
Execution File Name : XX001

Save the form

Note:

XX001 is the rdf file name without extension

Concurrent Program Definition



ORACLE

Concurrent Program Definition

Navigation

System Administrator→Concurrent→Program→Define

Program : HK-Sample3

Short Name : HKSample3

Application :Custom Development

Description : Sample Concurrent Program for Demonstrate

Style :A4

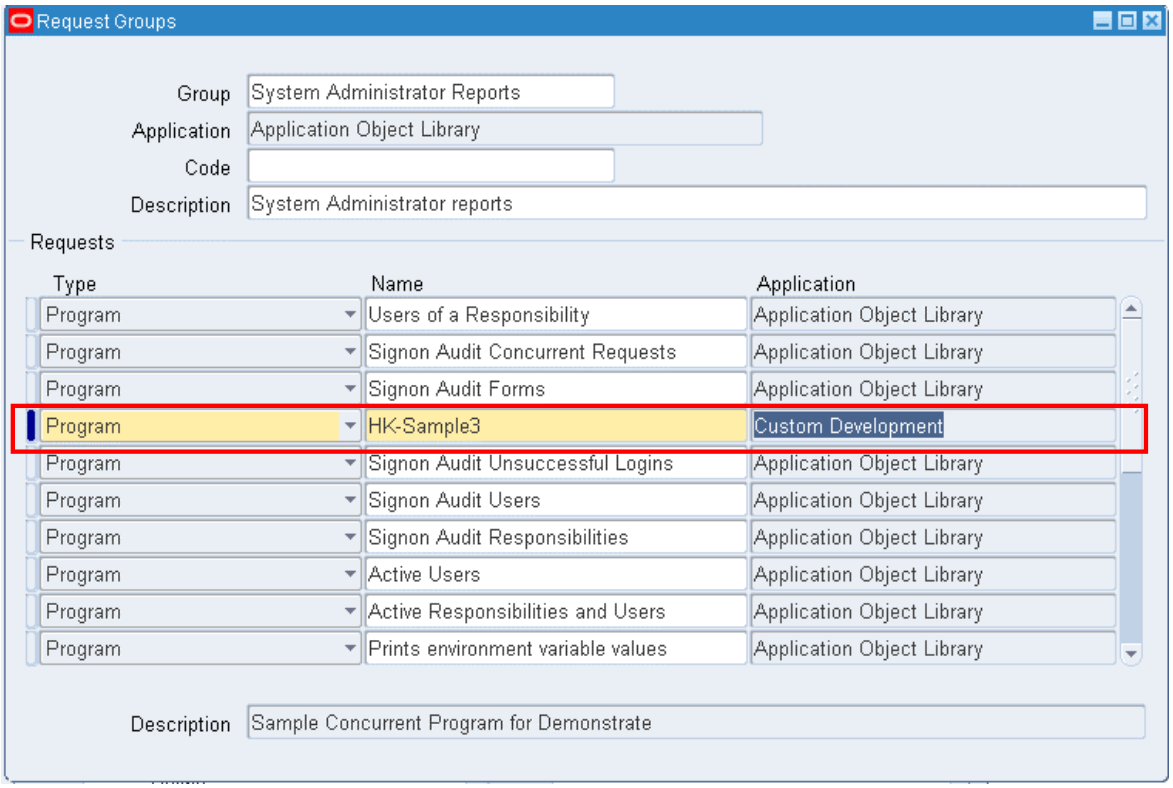
In Executable Block:

Name : HKSAMPLE3 (Is the Executable Name which is define in the last slide)

Note : If the Parameters are provided in the reports then you must define the parameter in the Parameter form And token is must match with name of the parameter in the report

Save the form

Associating a Program with a Request Group



Adding to Request Group

Navigation

System Administrator→Security→Responsibility→Request

Query for *System Administrator Reports* Request Group

(To query for the request Group

Press F4

Enter 'System Administrator Reports' in Group Filed

Press F11)

Add the Custom Concurrent Program which you created

Save the work

Summary

Now you able to:

- Identify the application top to keep rdf file
- Define executable
- Define concurrent program
- Attaching concurrent program to request group

4

Application Object Library

Forms Registration

Objectives

After completing this chapter you should be able to:

- How to develop a form for Oracle application
- How to compile in Server
- Registering form in oracle application
- Define function
- Attaching function to a menu

ORACLE

Objectives

Lesson Aim

- How to develop a form for Oracle application
- How to compile in Server
- Registering form in oracle application
- Define function
- Attaching function to a menu

Creating New form

To Create a new form for oracle application should follow the following steps

- Download Template.fmb from $AU_TOP/forms/US
- Save this template.fmb as another name.Make appropriate changes
- Verify that all the standard libraries are present in the form such as Custom,Appcore, Appcore2,appdaypk, fndsqf etc
- Compile the form and after removing the errors save it say Test1.fmb
- On FTP put Test1.fmb in binary mode to $Au_top/forms/US path.

ORACLE

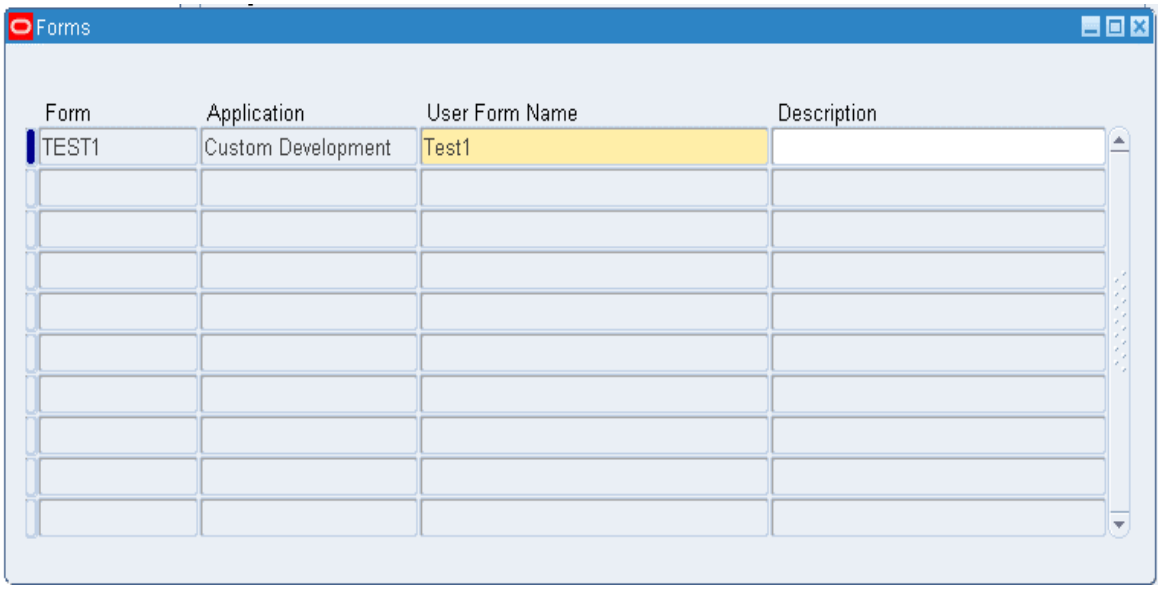
Creating New form

- From this $Au_top directory generate the form with f60gen module=Test1.fmb userid= apps/apps.It will create executable file .fmx.

Creating New form

- From this $Au_top directory generate the form with f60gen module=Test1.fmb userid= apps/apps.It will create executable file .fmx.
- Copy this fmx into custom top($Custom_top/forms/US)

Form Registration



Form Registration

Navigation

Application Developer→Application→Forms

Form : TEST1

Application : Custom Development

User Form Name : Test1

Save the form

Function Registration

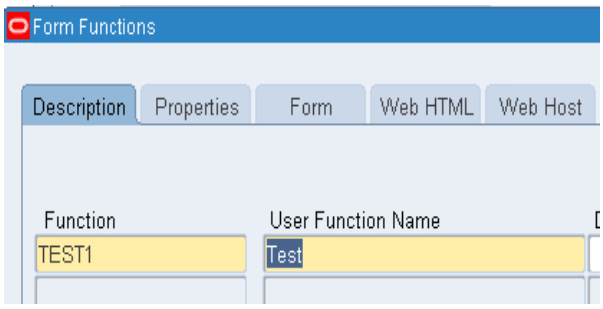


Fig1

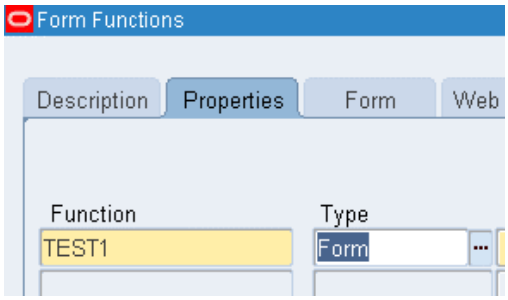


Fig2

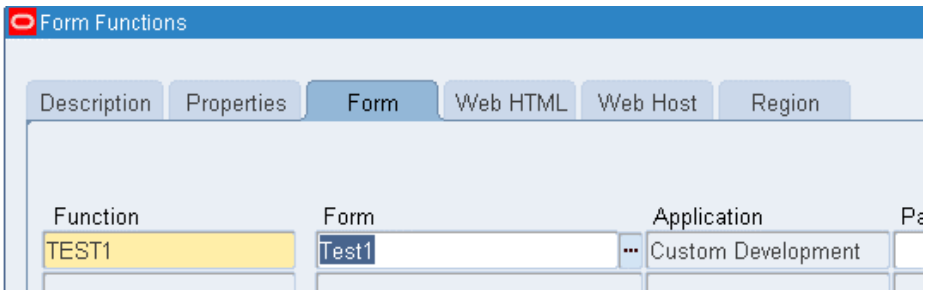


Fig3

ORACLE

Function Registration

Navigation

Application Developer→Application→Function

Description Tab: Fig1

Function : TEST1

User Function name : Test

Properties Tab: Fig2

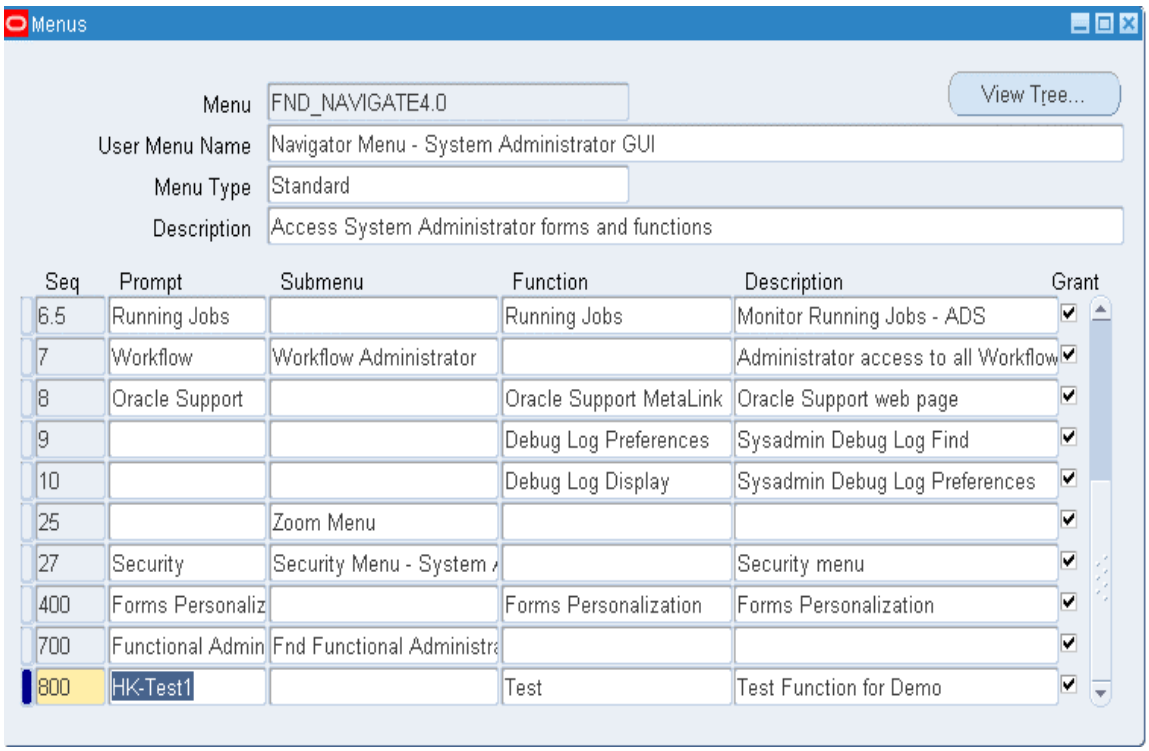
Type : Form

Form Tab: Fig3

Form : Test1 (Which is the form name registered in the last slide)

Save the form

Attaching Function to Menu



ORACLE

Attaching Function to Menu

Application Developer→Application→Menu

Query for Menu name : FND_NAVIGATE4.0

Go the last record in the Menu seq filed and add one more record

Seq : 800

Prompt : HK-Test1

Function : Test

Description : Test Function for Demo

Save the form

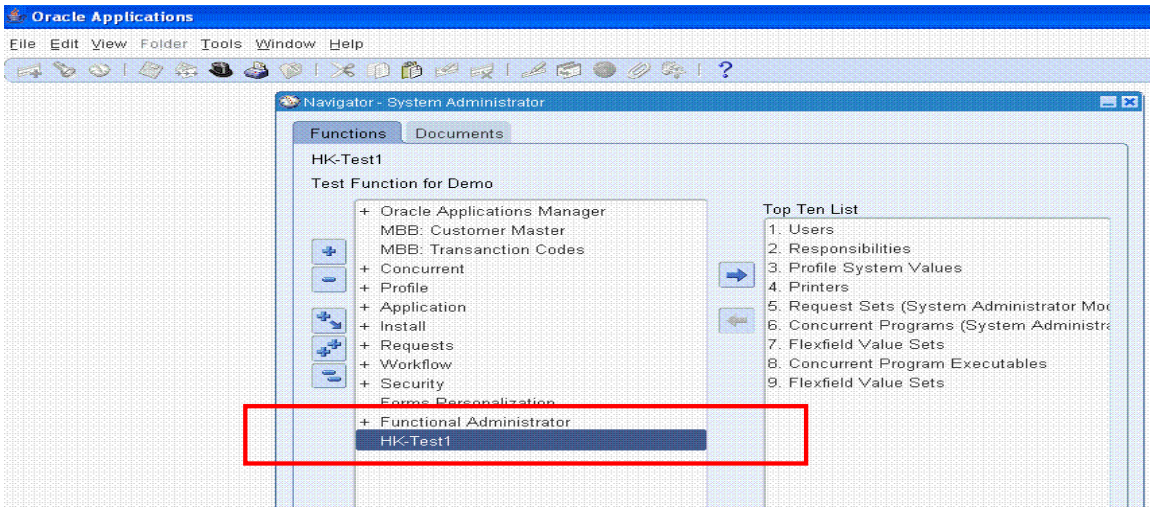
After saving the form it will give the message for compilation, just say ok

Now the menu is compiled and ready use

Attaching Function to Menu

FND_NAVIGATE4.0 menu is attached to Sysadministration responsibility

•Open the Responsibility



ORACLE

FND_NAVIGATE4.0 menu is attached to Sysadministration responsibility

Now if we open the systemadministrator responsibility we can see the function name that we have created now

Summary

Now you able to:

- To develop a form for Oracle application
- Compile in Server
- Registering form in oracle application
- Define function
- Attaching function to a menu
- Test the New Form

ORACLE

Objectives

Lesson Aim

- How to develop a form for Oracle application
- How to compile in Server
- Registering form in oracle application
- Define function
- Attaching function to a menu

CASE LITE

AOL – R12 EBS …… Duration 1 day.

Business Scenario:

Vision Corporation Inc. USA, a leading manufacturing company of Computer Systems has built in-house applications in its IT division for automating their Financial and Human Resources business activities. Now the company is planning to go for Oracle E-biz Suite for their Finance department. Company is also interested for its in-house Human resources application to run in Oracle E-biz Suite environment. For its Human Resources Application they have already done the Database related changes in Oracle Applications Database. But those changes still need to be updated in the front end of E-biz Suite and also the access of all the forms/UIs, PL-SQL Procedures related to Human Resources Application needs to be given to the end users of that department. Since there are few changes to be made in UIs as per the standards of Oracle E-biz suite in this application; company has already hired few Developers and also decided to deploy one senior and two junior System administrators. One Junior System administrator will be on temporary basis till the final implementation for the six months. So as a System Administrator you are expected to create Users, Responsibilities, Functions, Menus/Submenus, Data Groups, Applications, Concurrent Programs, Request Groups and Request Sets.

Assumptions:-

Now assume you are being hired as a senior system administrator and you have to do few system administration related activities which are defined as follows:

- Vision Corporation has already provided you two application users to do your initial work [OPERATIONS AND SYSADMIN].

- Company has given the testing environment to define the entire Oracle E-biz related objects and after testing they will implement them on the Production System.

Problem Statements:-

1. All the UIs and Reports files of In-house Human Resources Application are being kept in the file system Directory which can be seen by a System Variable named CUST_TOP. You are expected to create a new application by the name- Vision Human Resource.
2. You need to create two users for the Junior System Administrators.
3. As one the Junior System Administrator is only for 6 months, this must be ensured while creating the account for him/her.
4. Initially these two JSAs are also expected to do some of the things related to Application Developers as well.
5. For one of the JSA you are expected to create one customized Responsibility which will be associated with Vision Human Resource Application and will have customized menu, data group and security request group.
6. In the menu you need to add following functions:-
    a. Function to define a User.
    b. Function to define a Responsibility
    c. Function to define a Value Set
    d. Function to define Values for the Value Set
    e. Function to define a Data Group
    f. Function to define a Request Group

g. Function to define a Menu

h. Function to define the Structure for Descriptive Flex Field

i. Function to Run the concurrent Programs

j. A Submenu with following functions:-

    i. A function to register Applications.

    ii. A Function to define Executable

    iii. A function to define Concurrent Program.

7. This customized responsibility will have a Data group and application for the Data Group will be Vision Human Resource.

8. Second JSA in his/her responsibility will have two basic requirements.

    i. Will access the same menu and data group as accessed by first JSA, without Registering Application Function and Request Group Function.

    ii. Will access the User Define Function only to query for the user and not for Insert/Update of it.

9. These users also need to run following two Database programs as concurrent Programs (CP).

    i. The First program will take User name as parameter from the user and will display the list of responsibilities owned by that user, when user presses the “View Output” button in SRS Function. It will also display Total number (Count) of responsibilities if user presses “View Log” Button. Display

appropriate messages in case there is no responsibility assigned to the user.

ii. The Second Program will take user name from the user and will check which responsibilities are assigned for more than two years to a user and stores User name, Responsibility name and Duration in a Table. If it finds some records then on clicking the “View Output” button it should show the complete list of Responsibilities which are assigned for more than two years to that user. And if it finds no responsibility which is assigned to that user for more than two years then it should display “User Joined after Oracle E-biz Implementation” on pressing the “View Log Button”. Display appropriate messages in case there is no responsibility assigned to the user. [Hint : The PL/SQL Program must be created in a package]

10. These two JSAs should also be able to run three reports related to Function Security through a request set, which is part of their Request Group.

11. These function security reports must take the same parameters and need to share those values among themselves.

12. These functions security reports must be specified in such a manner that even if the first fails/warns, the next one must run.

13. Test all Problem Statements.