



Grado en Ingeniería Informática

APLICACIONES DE BASES DE DATOS

SESIÓN 2

Formularios e Informes

Docentes:

Raúl Marticorena

Jesús Maudes

Mario Martínez Abad



Índice de contenidos

1. INTRODUCCIÓN.....	3
2. OBJETIVOS.....	3
3. CREACIÓN DE APLICACIONES EN APEX.....	3
3.1 Creación del <i>workspace</i> de trabajo.....	3
3.2 Importación de los componentes al <i>workspace</i>	6
3.3 Creación de una nueva aplicación.....	11
3.4 Creación de formularios en una aplicación.....	16
3.5 Creación de formularios tabulares.....	21
3.6 Creación de formularios maestro-detalle.....	23
3.7 Creación de informes.....	26
3.8 Creación de informes interactivos.....	28
3.9 Creación de gráficos.....	30
4. COMPONENTES COMPARTIDOS.....	34
4.1 Listas de valores.....	34
4.2 Migas de pan.....	38
4.3 Menú de navegación.....	40
5. ANEXO.....	42
5.1 Exportación e importación en APEX de aplicaciones.....	42



1. Introducción

El presente guión da una guía básica de instalación, configuración y ajuste para la creación de formularios e informes con APEX. Partiendo de un *script* que crea un esquema de ejemplo, se indican la serie de pasos a realizar. **Dada la complejidad de APEX, solo se plantean los casos básicos que permite la herramienta.**

2. Objetivos

- Familiarizarse con el entorno de APEX para la creación de aplicaciones web.
- Instalar la aplicación de ejemplo.
- Guiar a la elaboración de formularios e informes sobre el esquema importado.

3. Creación de aplicaciones en APEX

3.1 Creación del *workspace* de trabajo

Nos validamos como **administradores de APEX** en el enlace de administración ([Application Express Administration Services](http://localhost:8080/apex/apex_admin) – http://localhost:8080/apex/apex_admin) y creamos un *workspace* con nombre OEHR en la opción **/Manage Worspaces/Create Workspace**.

Se introduce el nombre del *workspace* OEHR y se deja el campo *Workspace ID* vacío para que tome el valor por defecto.

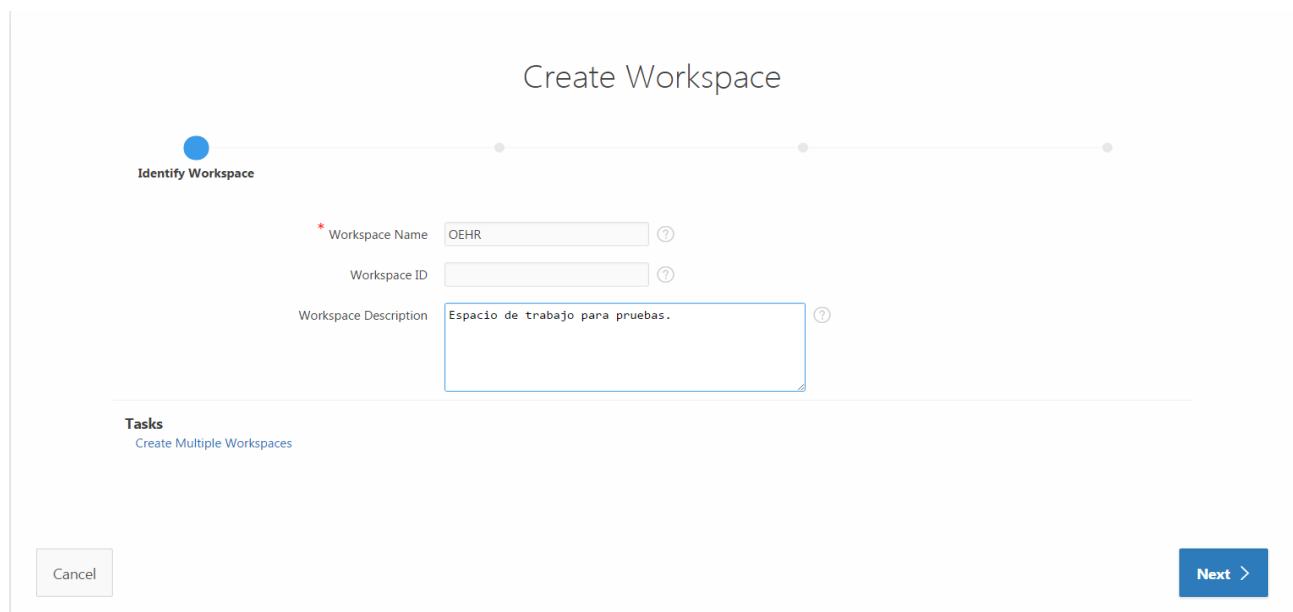


Ilustración 1: Creación de un workspace



No se reutiliza ningún esquema de Oracle, sino que se crea uno nuevo:

- nombre de esquema OEHR
- *password* OEHR.

Aceptamos la cuota de disco dada.

The screenshot shows the 'Create Workspace' wizard at the 'Identify Schema' step. A progress bar at the top has three steps: the first is completed (green checkmark), the second is the current step (blue circle), and the third is pending (grey circle). Below the progress bar, the title 'Create Workspace' is centered. The instruction reads: 'Select whether or not the schema already exists. If the schema exists, select the schema from the list. If the schema does not exist, enter a name and password and choose the size of the associated tablespace to be created.' The form contains the following fields: 'Re-use existing schema?' with a dropdown set to 'No'; 'Schema Name' with a text input containing 'OEHR'; 'Schema Password' with a masked password input '****'; and 'Space Quota (MB)' with a dropdown set to '100'. At the bottom left are '<' and 'Cancel' buttons, and at the bottom right is a 'Next >' button.

Ilustración 2: Configuración del workspace

Creamos el usuario ADMIN, con password OEHR e introducimos los datos adicionales. Ej: Raúl Marticorena admin@ubu.es.

The screenshot shows the 'Create Workspace' wizard at the 'Identify Administrator' step. The progress bar now has two completed steps (green checkmarks) and the current step is highlighted with a blue circle. The title 'Create Workspace' remains centered. The instruction is not explicitly repeated but the fields are for administrator details. The form contains: 'Administrator Username' with 'ADMIN'; 'Administrator Password' with '****'; 'First Name' with 'Raúl'; 'Last Name' with 'Marticorena'; and 'Email' with 'admin@ubu.es'. Each field has a red asterisk indicating it is required. At the bottom left are '<' and 'Cancel' buttons, and at the bottom right is a 'Next >' button.

Ilustración 3: Datos de usuario administrador del workspace

Se nos muestra un breve resumen de las opciones introducidas y confirmamos la creación. Confirmamos usando el botón **Create Workspace**. Este proceso puede tardar un cierto tiempo.



Create Workspace

You have requested to provision a new Workspace.

Workspace Information:

Name	OEHR
Workspace ID	System Assigned
Description	Espacio de trabajo para pruebas....

Administrator Information:

User Name	ADMIN
E-mail	admin@ubu.es

Schema Information:

Reuse Existing Schema	No
Schema Name	OEHR
Tablespace will be created	APEX_XXX
Datafile for tablespace	C:/ORACLEXE/APP/ORACLE/ORADATA/XE/APEX_XXX.DBF

< Cancel Create Workspace

Ilustración 4: Resumen de opciones de configuración del workspace

En una instalación típica en Windows, la ruta sera `c:\oraclexe\app\oracle\oradata\xe\apex_XXX.dbf`. En GNU/Linux nos encontraremos con un ruta similar a `/u01/app/oracle/oradata/XE/APEX_XXX.DBF`. Donde posteriormente se sustituye XXX por un código propio de Oracle.

Si el proceso ha finalizado correctamente, se presentará una ventana similar a la mostrada en Ilustración 5, donde la ruta física cambiará en función del sistema operativo utilizado:

Workspace Created

Workspace OEHR provisioned with administrator ADMIN. Database user OEHR created with default tablespace APEX_1710433682109030 using datafile C:\ORACLEXE\APP\ORACLE\ORADATA\XE\APEX_1710433682109030.DBF

Edit Workspace Done

Ilustración 5: Confirmación de la creación del workspace

Pulsamos el botón **Done**. En la ventana de administración, en la opción **Manage Workspaces/Manage Developers and Users** seleccionamos el usuario *admin* de este *workspace* que acabamos de crear



(OEHR), y marcamos en la opción *Require Change of Password on First Use* con valor **No**. **Este paso no es obligatorio**, pero evita que en el primer acceso, sea obligatorio tener que cambiar la contraseña de usuarios. Si no, simplemente estaremos obligados a cambiar la contraseña la primera vez que se accede (aunque en principio, admite repetir la contraseña antigua). Para guardar las modificaciones pulsamos en **Apply Changes**.

Salimos de la sesión (opción **Sign out** en esquina superior derecha, en el icono de usuario).

Debemos prestar atención, puesto que al abandonar la sesión (web) se nos muestra la pantalla de bienvenida en APEX (**NO la ventana de administración**), preguntándonos por tres valores: **workspace**, **login** y **password** (mientras que la ventana de validación como administrador de APEX sólo nos preguntará por el **login** y **password**). **Es muy importante tener claro con qué rol estamos accediendo a APEX en todo momento.**

3.2 Importación de los componentes al *workspace*

Nos validamos como **administradores del workspace** OEHR (que se acaba de crear en el paso previo) con *username* ADMIN y *password* OEHR.

Por defecto nos pedirá cambiar el *password*. Para evitar esto, se debe haber realizado el ajuste mencionado en la sección previa.

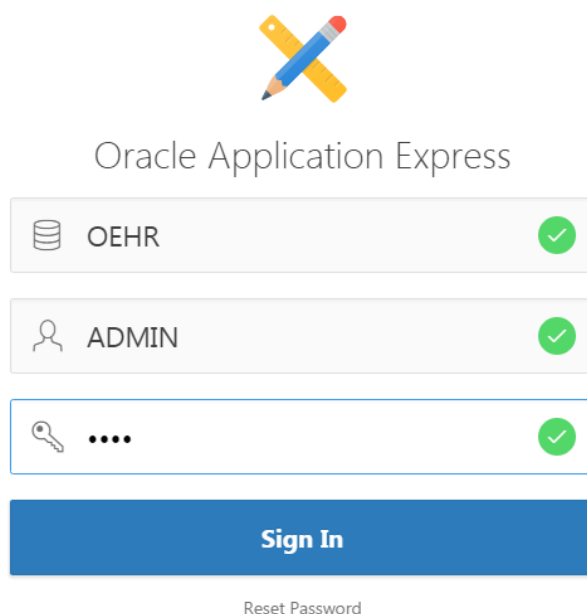


Ilustración 6: Validación para la gestión de workspaces

Al entrar en el *workspace* seleccionamos la opción *App Builder* en la esquina superior izquierda.

Inicialmente tendremos disponibles un conjunto de 11 tablas¹ (9 que tendrán un prefijo `DEMO_` y las 2 tablas `DEPT` y `EMP`), junto con una aplicación de ejemplo llamada *Sample Database Application 100*.

Si queremos ver qué tipo de aplicaciones se pueden construir, en *App Builder* escogemos dicha aplicación de ejemplo. Se muestran el conjunto de páginas que componen la aplicación (ver Ilustración 7).

¹ Existen algunas tablas adicionales con prefijo `APEX_` para uso propio de APEX. Se pueden consultar las tablas desde SQLPLUS con los comandos vistos en la sesión previa.



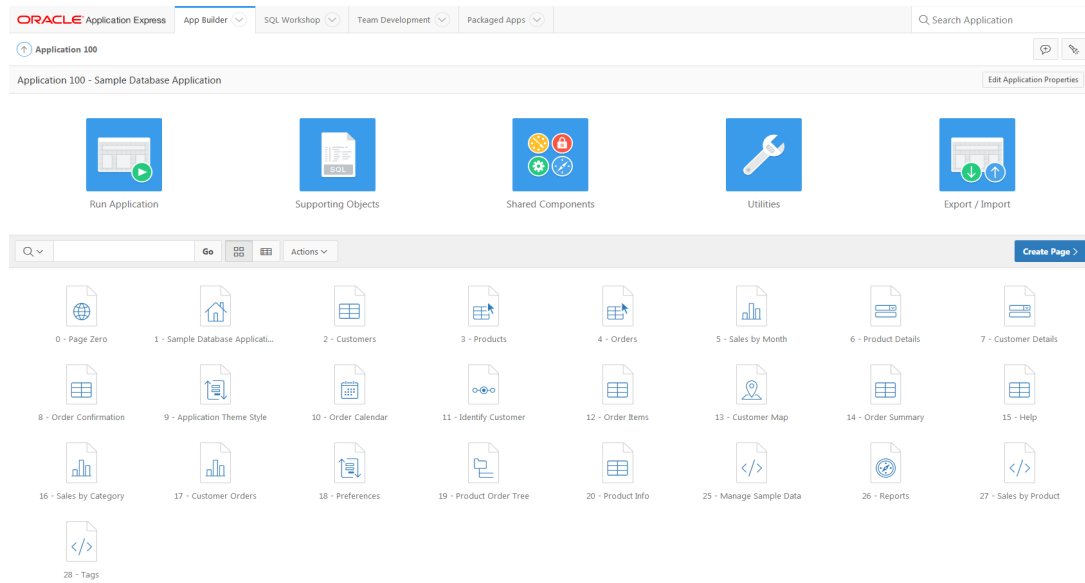


Ilustración 7: Aplicación de ejemplo instalada

Para ejecutar la aplicación, elegimos la opción *Run Application* en la esquina superior izquierda (se nos solicita que usemos una cuenta del *workspace*, en nuestro caso, usamos *admin* y OEHR). Podemos navegar por los formularios e informes. Para volver podemos utilizar la barra de botones inferior, presionando en *Home* o *Aplicación <Nº>* para volver a la página inicial del *workspace*, o de edición de la aplicación, respectivamente.

Obviamente en esta asignatura no se pretende hacer una aplicación tan elaborada, pero permite tomar una idea de todas las posibilidades que ofrece una herramienta tan potente como APEX. Finalizado un primer recorrido por una aplicación APEX, abordaremos la importación de tablas y datos adicionales.

Seleccionamos la pestaña superior **App Builder** y seleccionamos el botón **Import**. Cargamos el fichero (debe estar disponible en la plataforma UBUVirtual):

- `oehr_sample_objects_installer.sql`

La aplicación contiene además los *Supporting Objects* necesarios incluyendo tablas, vistas, secuencias, etc. para añadir en el esquema de bases de datos OEHR.



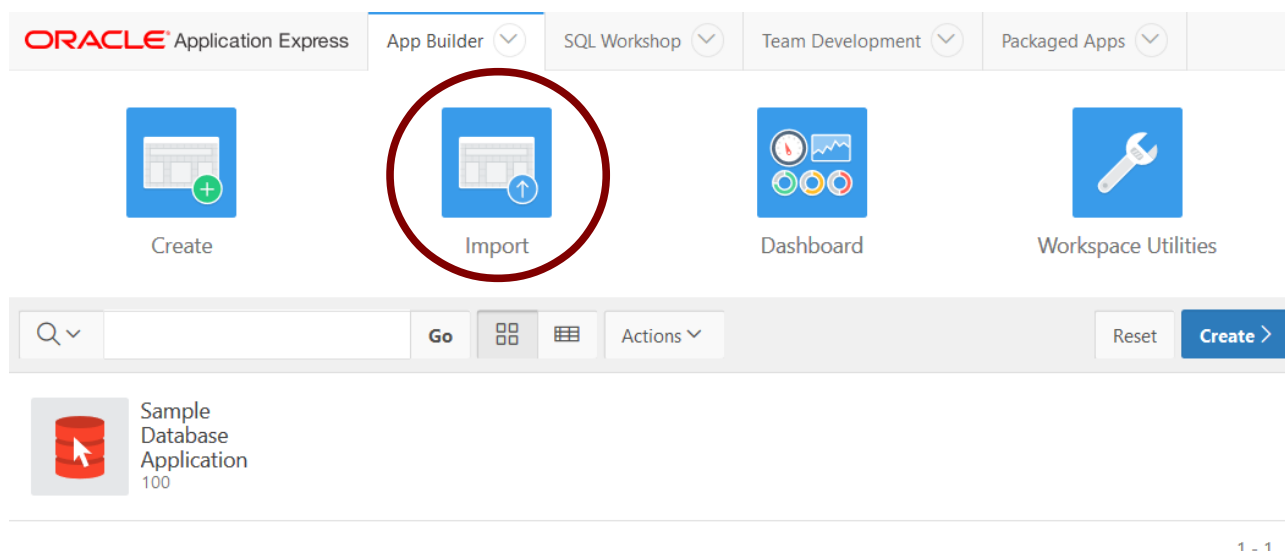


Ilustración 8: Vista general de aplicaciones

Una vez seleccionado el fichero, se deja la opción marcada en *File Type* -> *Database Application, Page or Component Export*. **La importación incluirá los objetos de la base de datos y la aplicación de ejemplo a utilizar en las sesiones de prácticas.** Dejamos seleccionada la codificación en Unicode UTF-8.

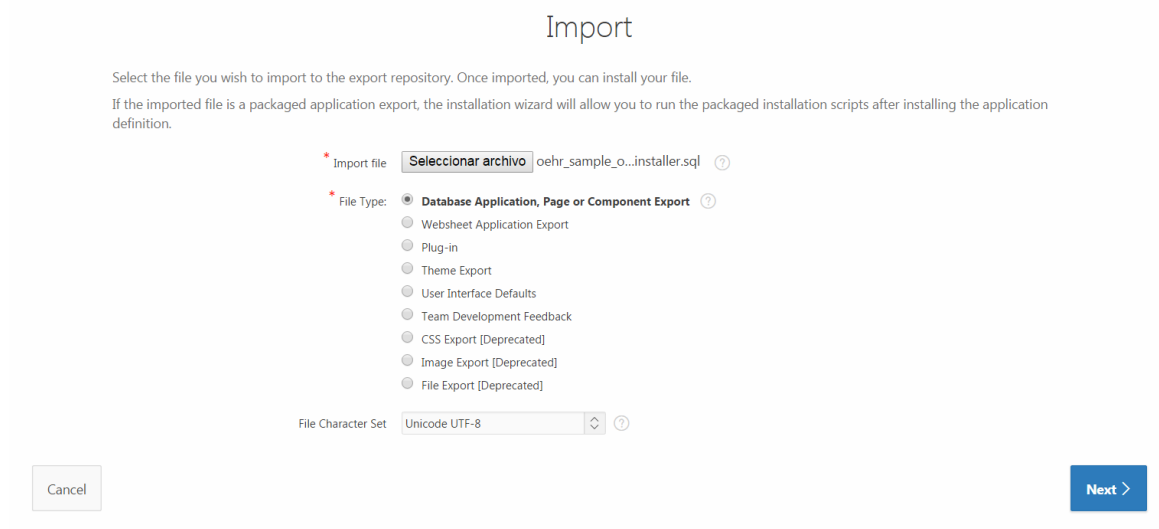


Ilustración 9: Especificando el fichero a importar

Confirmamos con **Next**. Si el proceso ha sido correcto se nos muestra una ventana con subtítulo *File Import Confirmation*. Pulsamos **Next** para continuar ahora con la instalación.



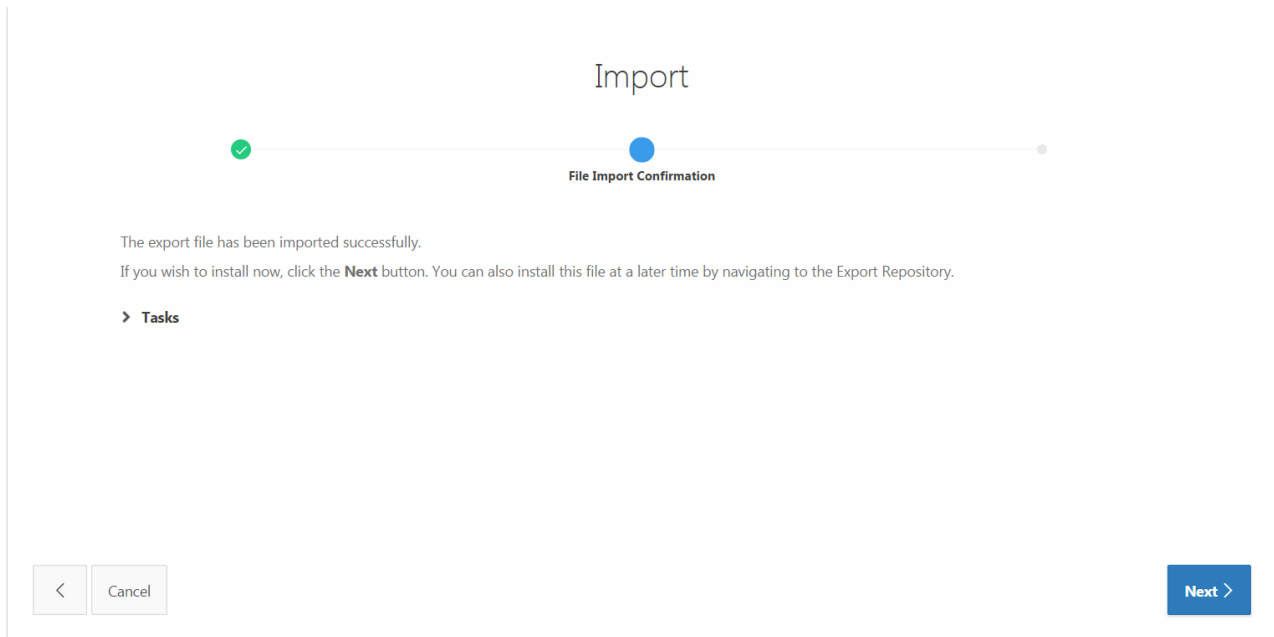


Ilustración 10: Confirmación de instalación

A continuación pasamos a instalar el fichero importado.

En el siguiente cuadro *Install Database Application*, seleccionamos:

- *Parsing schema* seleccionamos OEHR.
- *Build status* seleccionamos *Run and build application*
- *Install As Application* seleccionamos *Auto Assign New Application ID*



Install Database Application

✓ ✓ ● **Install**

When you install an application having the same ID as an existing application in the current workspace, the existing application is deleted and then replaced by the new application. If you attempt to install an application having the same ID as an existing application in a different workspace, a benign error message displays. If you are importing a packaged Application Express application, the installation wizard will allow you to install supporting objects.

Current Workspace: **OEHR** ⓘ

Export File Workspace ID: **1748711313955887** ⓘ

Export File Application ID: **558** ⓘ

Export File Version: **2012.01.01** ⓘ

Export File Parsing Schema: **PATRICK** ⓘ

Application Origin: **This application was exported from another workspace.** ⓘ

* Parsing Schema: **OEHR** ⓘ

* Build Status: **Run and Build Application** ⓘ

* Install As Application:

- ☒ **Auto Assign New Application ID** ⓘ
- ☐ Reuse Application ID 558 From Export File
- ☐ Change Application ID

> Tasks

< Cancel
Install Application

Ilustración 11: Instalación de aplicaciones de la base de datos

A continuación pide confirmación para instalar los objetos necesarios (*Supporting Objects*) con sus datos y la aplicación de ejemplo.

Install Application

● Supporting Objects

The installation of this application creates the objects necessary to use the Oracle Application Express 2 Day Plus Developer's Guide and the Oracle Application Express Advanced Tutorials Guide. It also loads the data necessary for the tutorial. Deleting this application and selecting to deinstall supporting objects will completely remove all objects that were created.

Application: **102 - OEHR Sample Objects** ⓘ

Parsing Schema: **OEHR** ⓘ

Free Space Required in KB: **4,800** ⓘ

Install Supporting Objects:

- ☒ **Yes** ⓘ
- ☐ No

> Tasks

Cancel
Next >

Ilustración 12: Instalación de Supporting Objects

Confirmamos la creación de objetos de apoyo: *Install Supporting Objects: Yes*



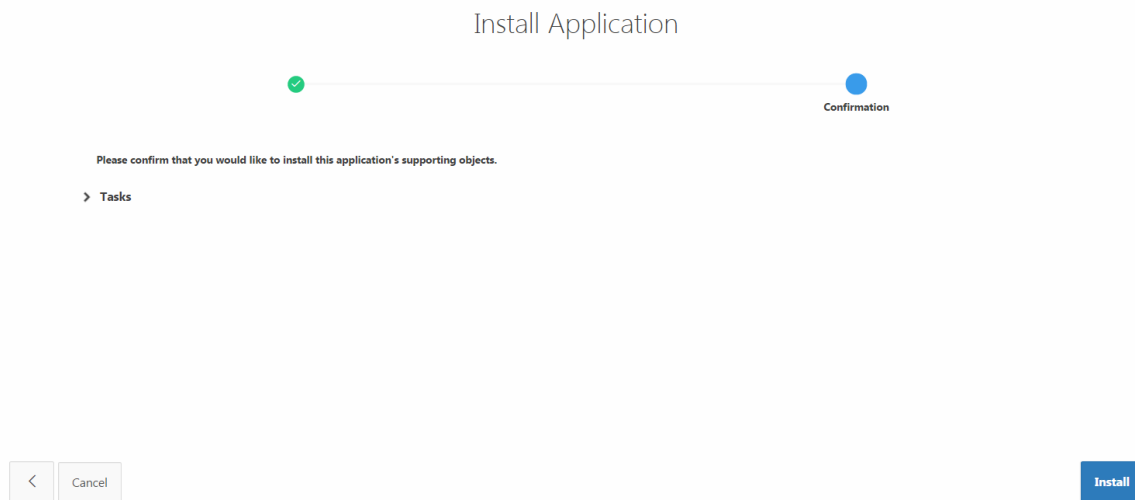


Ilustración 13: Confirmación de importación e inicio de instalación

Confirmamos la instalación con **Install**. El proceso puede llevar un cierto tiempo.

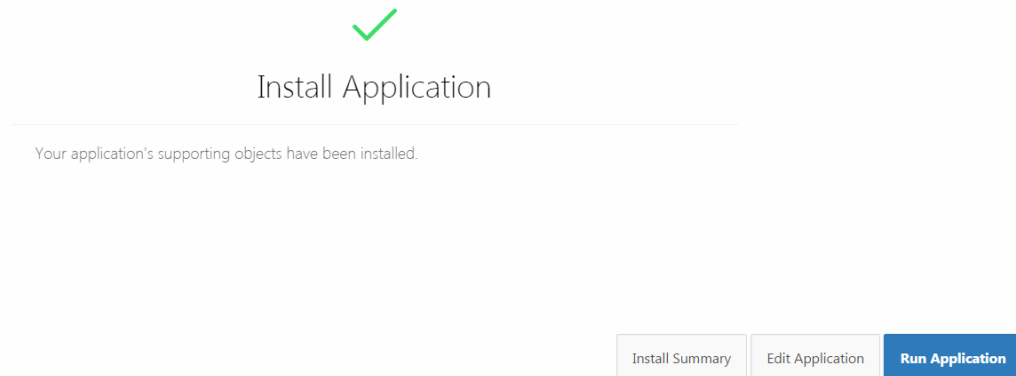


Ilustración 14: Mensaje final de confirmación de instalación

Se muestran las opciones de ejecución, edición o ver el resumen de la instalación. Visualizamos el resumen para ver el desglose de elementos instalados (*Install Summary*).



ORACLE

Application Express

Application Builder

SQL Workshop

Team Development

Packaged Apps

Q

🔍

🔄

👤

Rad

1 Install Summary

Application:

102 - OEHR Sample Objects

Install Summary

This page displays information about the Installation and Deinstallation script results for the current application.

Tasks

Install Supporting Objects

Upgrade Supporting Objects

Deinstall Application

Edit Installation Scripts

Manage Supporting Objects

Edit Application

Run Application

Install Summary

Script Name	Status	Statements Successful	Statements with Errors	Total Statements
create tables	Success	40	0	40
create procedures	Success	3	0	3
create views	Success	15	0	15
create triggers	Success	55	0	55
create comments	Success	86	0	86
seed customers	Success	563	0	563
seed products	Success	576	0	576
seed orders	Success	1889	0	1889
seed online catalog	Success	31	0	31
Report Total:		3258	0	3258

1 - 9

Con esto hemos finalizado la importación de tablas y objetos adicionales. En la siguiente sección, se detallará como crear una nueva aplicación desde cero a partir de dichos datos.

Seleccionamos **App Builder / Create**. Se puede crear una aplicación de escritorio (*Desktop*), orientada a móviles (*Mobile*), hoja web (*Websheet*) o a partir de aplicaciones empaquetadas (*Packaged App*). En esta asignatura nos centraremos sólo en el primer tipo, aplicaciones de escritorio, no siendo objeto de estudio los otros casos.

Ilustración 16: Opciones de creación de una aplicación

Create an Application

Name

* User Interface **Desktop**

Schema ?

* Name ?

* Application ?

Theme ^ ?

Theme Style ?

Ilustración 17: Definiendo el esquema y nombre de la aplicación

A continuación se pueden añadir nuevas páginas. Se pueden añadir en esta etapa, o bien añadir posteriormente, pero como **mínimo** se incluye una página inicial (puede ser una página en blanco de tipo *Blank*).

Create an Application

Pages

Page	Name	Type	Page Mode	Source Type	Source	Parent Page
	1	Home	Blank	Normal	-	-

Ilustración 18: Añadiendo páginas a la aplicación

Añadimos una página en blanco para continuar con el asistente. Se pueden copiar componentes compartidos desde otras aplicaciones (en el ejemplo se marca **No**):



Create an Application

✓

✓

●

Shared Components

Shared components are common application elements that can be displayed or applied across multiple pages in an application. To save time or maintain consistency between applications, copy the shared components from an existing application.

Copy Shared Components from Another Application: ☐ Yes ☒ No [?](#)

< Cancel

Next >

Ilustración 19: Copiando componentes compartidos

Seguidamente, se define el sistema de autenticación (cómo nos vamos a validar para acceder a la aplicación que estamos creando). Puede ser:

- Nivel de aplicación APEX con las cuentas creadas por el administrador del *workspace* APEX (Application Express)
- Sin autenticación.
- Con autenticación de usuarios de la base de datos Oracle (Ej: un usuario `HR` u `OEHR`).

Junto con la validación, se pueden establecer idioma, preferencias de idioma y formato de fechas.

Se selecciona validación *Application Express* y automáticamente se crea una página *101. Login Page*, que realiza el *login* de los usuarios antes de entrar en la aplicación, validando contra usuarios de APEX. Cambiamos las preferencias de idioma y formatos de fecha y hora.



Create an Application

Attributes

Authentication Scheme: Application Express Accounts ?

Language: Spanish (Traditional Sort) (es) ?

User Language Preference Derived From: Browser (use browser language preference) ?

Date Format: ^ ?

Date Time Format: ^ ?

Timestamp Format: ^ ?

Timestamp Time Zone Format: ^ ?

< Cancel Next >

Ilustración 20: Autenticación y configuración regional

Finalmente se presenta un resumen o resumen de las opciones.

Create an Application

Confirm

You have requested to create an application with the following attributes. Please confirm your selections.

Application	101
Name	ABD Test 1
Parsing Schema	OEHR
Default Language	en
Navigation	Navigation Menu List
Default Authentication Scheme	Application Express Authentication
Theme Type	Standard
Theme	Universal Theme
Subscribe Theme	No

< Cancel Create Application

Ilustración 21: Resumen de opciones creando una aplicación

Una vez pulsamos en el botón *Create Application*, se genera la nueva aplicación, sobre la cual podemos empezar a editar creando nuevas páginas con formularios e informes (botón *Create Page*) o bien por



ejemplo, ejecutar (botón *Run Application*). De momento siempre accedemos con el usuario previamente creado *admin* para nuestro *workspace*.

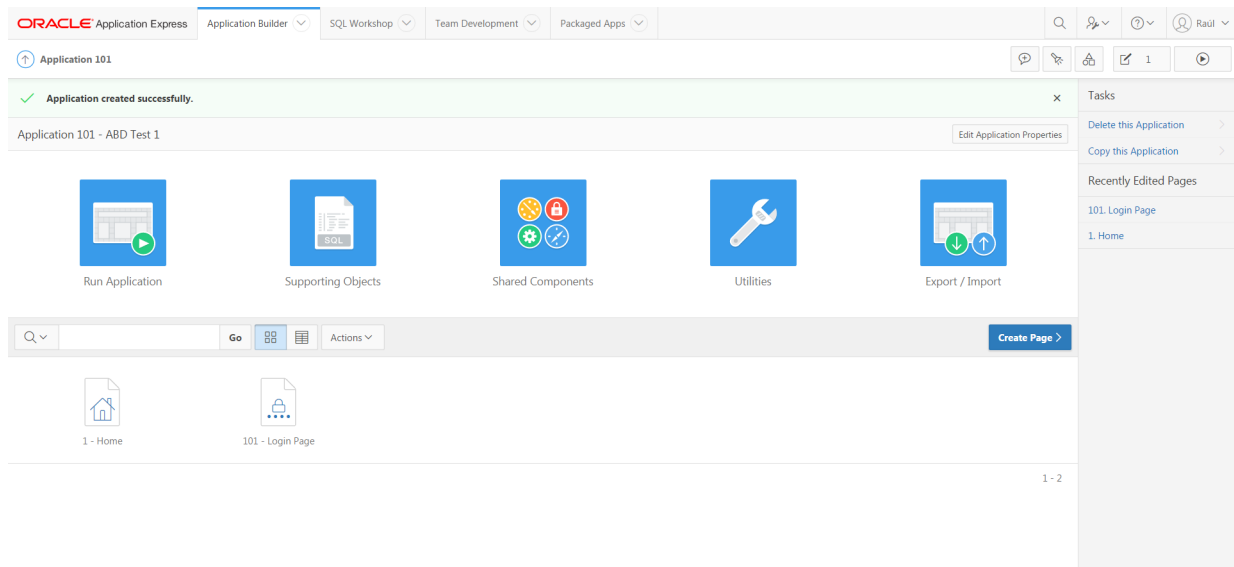


Ilustración 22: Página inicial de gestión de aplicaciones en Application Builder



3.4 Creación de formularios en una aplicación

Para añadir un nuevo formulario (o informe, etc) se puede hacer de distintas formas (páginas, regiones, ect.). En el ejemplo se creará una nueva página con la opción **Create Page**, que nos lleva a una página similar a la mostrada en la Ilustración 23.

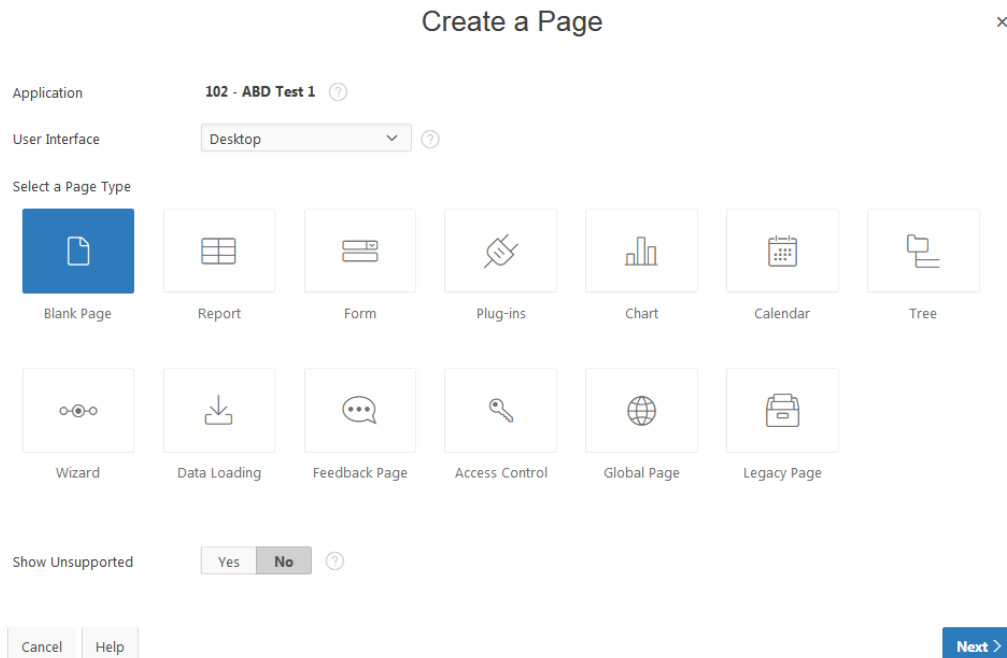


Ilustración 23: Tipos de páginas

Seleccionamos el tipo de página **Form**. A continuación se muestran el conjunto de posibles tipos de formularios que se pueden construir:

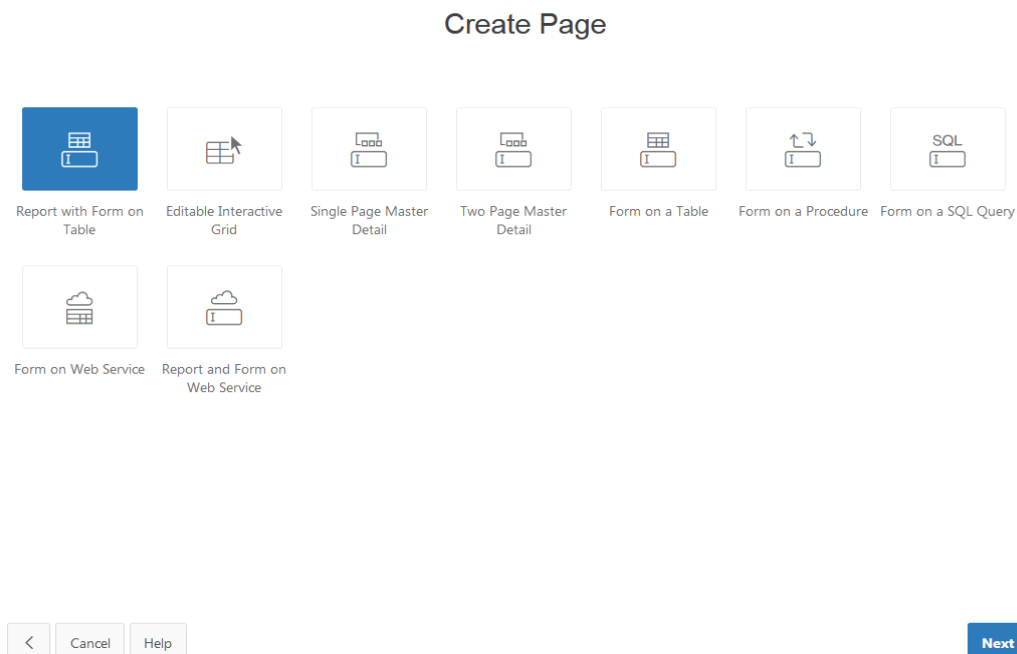


Ilustración 24: Tipos de formularios



En esta asignatura **no veremos todos los tipos de formularios**, puesto que excede en mucho los objetivos y duración de la asignatura, centrándonos en los **casos mas habituales o básicos** (formulario simple, formulario tabular y formulario maestro-detalle).

Seleccionamos **Form on a Table** y se inicia el asistente para configurar el formularios siguiendo los siguientes pasos:

Paso 1: Configuración de página. Opciones de personalización para la visualización, dando nombre al formulario (a la página), esto es obligatorio. Por motivos de simplicidad en todas las páginas seleccionamos *Normal* (no utilizaremos *Modal Dialog*), y sin personalizar nada respecto al *branching* de la página (estos parámetros son opcionales). Por el momento no añadimos migas de pan o *breadcrumbs*.

The screenshot shows the 'Create Form on Table' wizard with the title bar 'Create Form on Table' and a close button 'x'. A progress bar at the top has four steps, with the first step 'Page Attributes' highlighted by a blue dot. The 'Page Attributes' section contains the following fields:

- Page Number:** A text input with the value '6' and an up arrow icon.
- Page Name:** A text input field.
- Page Mode:** Two radio buttons, 'Normal' (selected) and 'Modal Dialog'.
- Branch Here on Submit:** A text input field.
- Cancel and Go To Page:** A text input field.
- Breadcrumb:** A dropdown menu with the selected option '- do not add breadcrumb region to page -'.

At the bottom of the wizard, there are three buttons: a back arrow, a 'Cancel' button, and a 'Next >' button.

Ilustración 25: Paso 1: Configuración de página

Paso 2: Configuración de navegación: se ajustan las opciones de menú relativas a la página actual. (En la parte final del documento se incluye un apartado relativo al uso de menús en APEX). Inicialmente la página creada no estará vinculada a ninguna entrada de menú.

The screenshot shows the 'Create Form on Table' wizard with the title bar 'Create Form on Table' and a close button 'x'. A progress bar at the top has four steps, with the second step 'Navigation Menu' highlighted by a blue dot. The 'Navigation Menu' section contains the following options:

- Navigation Preference:** Three radio buttons: 'Do not associate this page with a navigation menu entry' (selected), 'Create a new navigation menu entry', and 'Identify an existing navigation menu entry for this page'.

At the bottom of the wizard, there are three buttons: a back arrow, a 'Cancel' button, and a 'Next >' button.

Ilustración 26: Paso 2: Añadir opciones de menú



Paso 3: Selección de propietario, tabla y columnas. En nuestro caso el *workspace* está ligado sólo a un esquema y por lo tanto no hay múltiples opciones (Nota: un *workspace* puede estar ligado a varios esquemas de Oracle). Por ejemplo, elegimos la tabla de empleados (OEHR_EMPLOYEES). A continuación se eligen las columnas a mostrar en el formulario. En función del tipo de datos se asigna el correspondiente tipo de *item* para el formulario (cuadro de texto, selector de fecha, etc.). Esta asignación es editable posteriormente. Se mueven los elementos (campos) seleccionados de la lista de la izquierda a la derecha. En el ejemplo se han seleccionado todos los campos, pero no es obligatorio, pudiendo elegir solo aquellos que se crean convenientes.

Create Form on Table

Source

* Table / View Owner: OEHR

* Table / View Name: OEHR_EMPLOYEES (table)

* Select Columns

EMPLOYEE_ID (Number)
FIRST_NAME (Varchar2)
LAST_NAME (Varchar2)
EMAIL (Varchar2)
PHONE_NUMBER (Varchar2)
HIRE_DATE (Date)
JOB_ID (Varchar2)
SALARY (Number)
COMMISSION_PCT (Number)
MANAGER_ID (Number)

< Cancel Next >

Ilustración 27: Selección de propietario, tabla y columnas

Paso 4: Configuración de la clave primaria de la tabla.

Podemos utilizar el ROWID o bien clave primaria. En este segundo caso, estamos limitados a **claves primarias como máximo de dos campos (limitación APEX)**. Un número mayor exige definir procesos DML (preferiblemente en un paquete utilizando el Object Browser) o creando (y referenciando) una nueva clave primaria. Ej: crear una vista o tabla y generar una nueva clave primaria. En la versión 5 de APEX se recomienda en este último caso, el uso del ROWID. Elegimos **Select Primary Key Column(s)** dado que la clave es simple: `EMPLOYEE_ID` (en el segundo desplegable Primary Key Column no hay nada que seleccionar). Los campos de la clave primaria no son editables ni visibles en el formulario por defecto (aunque puede cambiarse este comportamiento). Como **Primary Key Source Type** vamos a utilizar secuencias de Oracle. En el ejemplo, seleccionamos la secuencia existente `OEHR_EMPLOYEES_SEQ`.



Create Form on Table

×

Form Attributes

Primary Key Type ☐ Managed by Database (ROWID)
☒ Select Primary Key Column(s)

Primary Key Column EMPLOYEE_ID (Number) ▼

Primary Key Column - Select Column - ▼

Primary Key Source Type ☐ Existing Trigger
☒ Existing Sequence

Sequence OEHR_EMPLOYEES_SEQ ▼

< Cancel Create

Ilustración 28: Paso 4: Configuración de la clave primaria de la tabla

Una vez creada la página entramos en el *Page Designer*, donde se pueden realizar ajustes y ediciones. A la nueva página creada se le da siempre un número como código identificativo. Podemos ejecutar la aplicación o la página, en *Application Builder*, seleccionando la aplicación y página, pulsando en botón de *Save and Run* (flecha sobre botón azul en esquina superior derecha). En pantalla se nos muestra el formulario que verá el usuario (con la típica barra de botones en la parte inferior para volver a la vista de edición pudiendo ir a *Home*, *Application* o *Page*). **Dado que esta página que acabamos de crear no están enlazada a otros elementos, inicialmente solo podremos ejecutarla seleccionándola directamente.**²

² Normalmente debe existir un camino de navegación desde la página de inicio (*login* o *home*) a nuestra página, en caso contrario tendremos páginas inalcanzables para nuestros usuarios. Se pueden crear por motivos de prueba o testing, pero finalmente deberían enlazarse, o bien eliminarse en el producto final.



3.5 Creación de formularios tabulares

Permite la edición de datos sobre tablas. El proceso de creación es exactamente igual que para un formulario normal, con la salvedad de que los datos no se presentan fila a fila (registro a registro, de uno en uno), sino que se visualizan varias filas a la vez, permitiendo editarlas en un formulario de tipo *Editable Interactive Grid*.

Paso 1: Configuración de página: indicamos el nombre del página para el formulario.

The screenshot shows the 'Create Interactive Grid' dialog box with the 'Page Attributes' step selected. The 'Type' is set to 'Interactive Grid'. The 'Page Number' is 20. The 'Page Name' field is empty. The 'Page Mode' is set to 'Normal'. The 'Breadcrumb' dropdown is set to '- do not use breadcrumbs on page -'. Navigation buttons include '< Cancel' and 'Next >'. A progress bar at the top shows the first step as active.

Ilustración 29: Configuración de página en formulario tabular

Paso 2: Configuración de navegación: se ajustan las opciones de menú relativas a la página actual.

The screenshot shows the 'Create Interactive Grid' dialog box with the 'Navigation Menu' step selected. The 'Navigation Preference' section has three radio button options: 'Do not associate this page with a navigation menu entry' (selected), 'Create a new navigation menu entry', and 'Identify an existing navigation menu entry for this page'. Navigation buttons include '< Cancel' and 'Next >'. A progress bar at the top shows the second step as active.

Ilustración 30: Configuración de navegación



Paso 3: Selección de fuente de datos: se selecciona si queremos editar o no los datos, la fuente de datos (tabla o consulta SQL) y las columnas a visualizar (y editar). Elegimos por ejemplo la tabla OEHR_EMPLOYEES y todas sus columnas, identificadas en este caso por ROWID.

Create Interactive Grid

Editing Enabled ?

* Source Type ☒ Table ☐ SQL Query ?

* Table / View Owner OEHR ?

* Table / View Name OEHR_EMPLOYEES (table) ?

Primary Key Column ROWID ?

* Select Columns ?

EMPLOYEE_ID (Number)
FIRST_NAME (Varchar2)
LAST_NAME (Varchar2)
EMAIL (Varchar2)
PHONE_NUMBER (Varchar2)
HIRE_DATE (Date)
JOB_ID (Varchar2)
SALARY (Number)

< Cancel Create

Ilustración 31: Configuración de datos a editar

Finalmente confirmamos la creación con el botón *Create*. Igual que antes, se nos mostrará el diseñador de páginas, y desde ahí podemos ejecutar para previsualizar el resultado. Se nos muestra un listado completo de empleados y si hacemos doble click sobre las celdas se permite la edición de campos.³

³ Aclarar que estos son meros ejemplos para recorrer las distintas opciones y mostrar los resultados, pero que obviamente en una aplicación final deberíamos cuidar más detalles y enganchar las páginas entre sí.



3.6 Creación de formularios maestro-detalle

El proceso de creación es exactamente igual que para un formulario normal, con la salvedad de que se trabaja con dos tablas: una juega el rol de **maestro** y la otra del **detalle**. Entre la tabla maestro y detalle **existe una relación de 1:N** (uno a varios) de tal forma que para un registro de la tabla maestra, se relacionan un número N registros de la tabla detalle (podría ser cero según el problema particular que estemos tratando). Si queremos que la navegación sea fluida, asociamos automáticamente el cambio de los registros **detalle** según se navega en las filas del **maestro**.

Se selecciona la opción *Two Page Master Detail* a la hora de crear el nuevo formulario (también existe la opción *Single Page Master Detail*, que por el momento no usaremos). Se crearán varias páginas.

Entre ambas tablas habrá una relación lógica que lleva a que la tabla **detalle** tenga como **clave foránea** la clave de la tabla **maestro**. Esto permite visualizar todos los registros relacionadas de la tabla **detalle** en función del registro seleccionado de la tabla **maestro**.

Paso 1: :

Create Master Detail

Page Attributes

* Master Page Number 25 ?

* Master Page Name Ordenes de Compra ?

* Detail Page Number 26 ?

* Detail Page Name Detalle de Ordenes de Compra ?

Breadcrumb - do not add breadcrumb region to page - ?

< Cancel Next >

Ilustración 32: Configuración de la página maestra

Paso 2: Configuración de navegación: se ajustan las opciones de menú relativas a la página actual. De momento sin asociar a entrada de menú.



Create Master Detail

Navigation Menu

Navigation Preference

- ☒ Do not associate this page with a navigation menu entry ?
- ☐ Create a new navigation menu entry
- ☐ Identify an existing navigation menu entry for this page

< Cancel Next >

Ilustración 33: Configuración de navegación

Paso 3: Seleccionar tabla maestra. Seleccionamos la tabla de órdenes de compra, junto con las columnas a utilizar, tal y como se muestra en la Ilustración 34, recorriendo por orden según la clave primaria ORDER_ID.

Create Master Detail

Master Source

* Table / View Owner: OEHR ?

* Table / View Name: OEHR_ORDERS (table) ?

* Primary Key Column: ROWID ?

* Form Navigation Order: ORDER_ID (Number) ?

* Select Columns ?

ORDER_ID (Number)
ORDER_DATE (Timestamp(6) With Local Time Zone)
ORDER_MODE (Varchar2)
CUSTOMER_ID (Number)
ORDER_STATUS (Number)
ORDER_TOTAL (Number)
SALES_REP_ID (Number)
PROMOTION_ID (Number)

< Cancel Next >

Ilustración 34: Seleccionar tabla maestra



Paso 4: Seleccionar tabla detalle. La tabla detalle tendrá una clave foránea que referencia al campo de la tabla maestra. APEX nos muestra solo aquellas tablas (esto es opcional y se puede desactivar en el *radio button Show Only Related Tables*) que tienen una clave foránea definida sobre la tabla maestra, facilitando la selección de este elemento. En nuestro ejemplo, las líneas de órdenes de compra en la tabla OEHR_ORDER_ITEMS indicando la navegación *Master Detail Foreign Key* como ORDER_ID → ORDER_ID.

The screenshot shows the 'Create Master Detail' dialog box in Oracle APEX. The dialog has a title bar with a close button. Below the title bar is a progress indicator with four steps, the fourth of which is active. The main area contains the following fields:

- Show Only Related Tables:** Radio buttons for 'Yes' (selected) and 'No'.
- Table / View Owner:** Text field with 'OEHR' and a help icon.
- * Table / View Name:** Dropdown menu with 'OEHR_ORDER_ITEMS' and a help icon.
- * Primary Key Column:** Dropdown menu with 'ORDER_ID (Number)' and a help icon.
- Primary Key Column 2:** Dropdown menu with '- Select Column -' and a help icon.
- * Master Detail Foreign Key:** Dropdown menu with 'ORDER_ID -> ORDER_ID' and a help icon.
- * Select Columns:** A list of columns to be displayed in the detail view. The list includes 'ORDER_ID (Number)', 'LINE_ITEM_ID (Number)', 'PRODUCT_ID (Number)', 'UNIT_PRICE (Number)', 'QUANTITY (Number)', and 'ORDER_ITEM_ID (Number)'. The list is currently empty, and the columns are shown in a separate pane on the right.

At the bottom of the dialog are three buttons: '<', 'Cancel', and 'Create'.

Ilustración 35: Seleccionar tabla detalle

Finalmente confirmamos la creación con el botón *Create* abriendo el diseñador de páginas. Ojo, en este ejemplo, se generan dos páginas: una primera con la visualización (informe) de las filas de la tabla maestra (con nombre "Órdenes de compra"), y una segunda página que muestra el formulario con los datos de esa fila maestra (órdenes de compra) y en la parte inferior una tabla (grid) con las filas detalle relacionadas (items) para su edición. Desde esta segunda página se puede navegar hacia adelante y hacia atrás, en el conjunto de filas de la tabla maestra.



3.7 Creación de informes

Los informes permiten visualizar la información, **sin permitir su edición**. Su finalidad es permitir **SOLO** la **consulta** y **visualización** de la información.

Creamos una nueva página y seleccionamos *Report*, en lugar de *Form* como hemos hecho previamente.

Paso 1: Selección del tipo de informe. En este ejemplo concreto, de los cinco posibles, seleccionamos *Classic Report*.

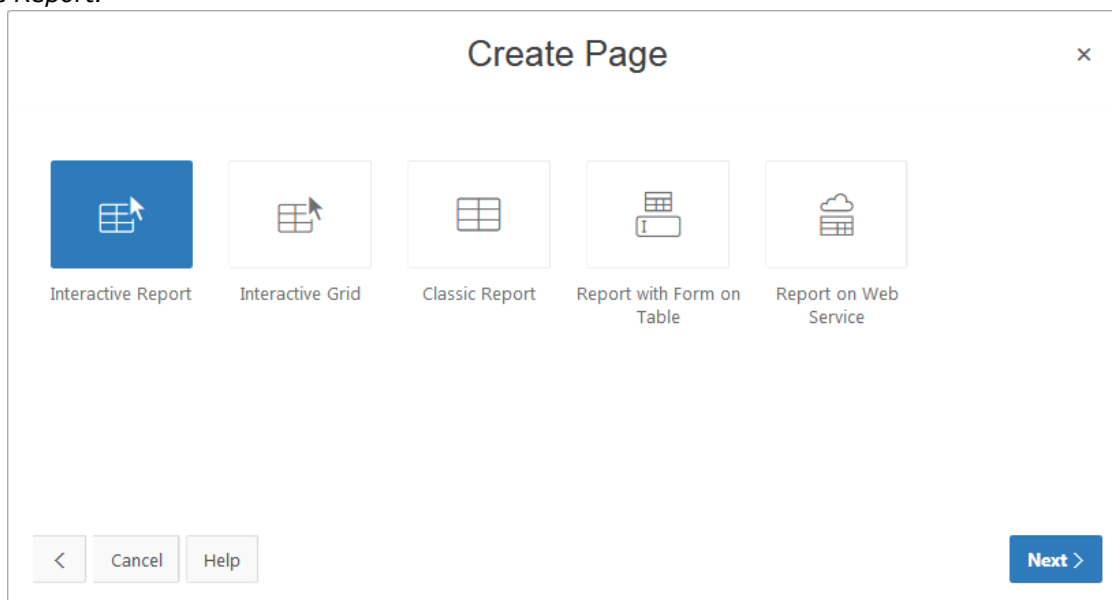


Ilustración 36: Selección de tipo de informe

Paso 2: Configuración de navegación: se ajustan las opciones de menú relativas a la página actual. Seguimos trabajando en modo Normal y sin migas de pan.

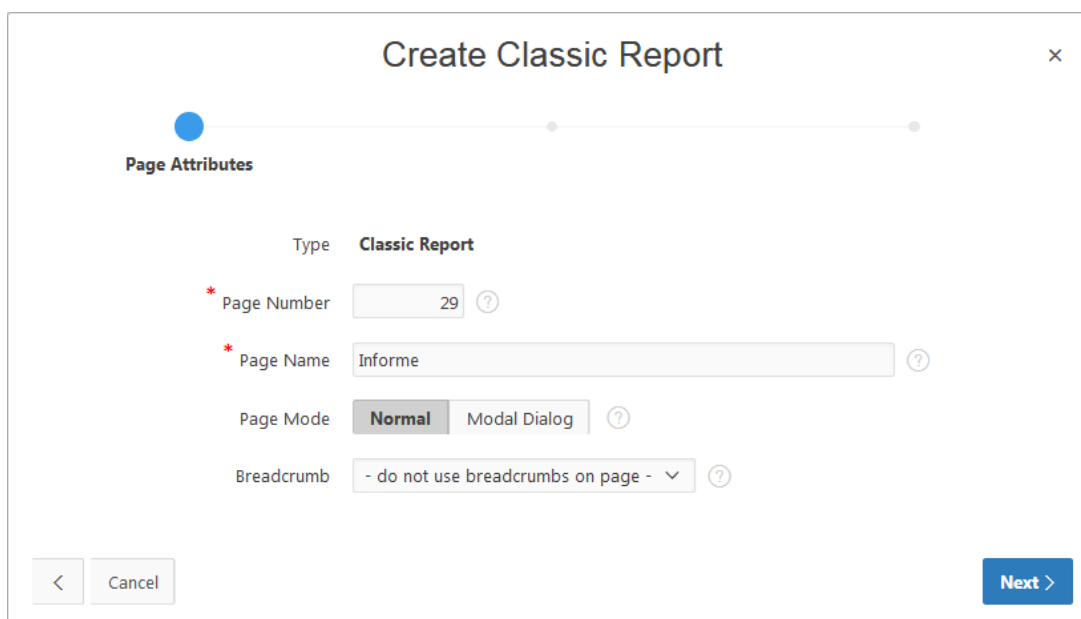


Ilustración 37: Configuración de la página



Paso 3: Configuración de navegación: se ajustan las opciones de menú relativas a la página actual, sin enganchar con menú.

The screenshot shows the 'Create Classic Report' dialog box with the title bar and a close button. A progress bar at the top has three steps: the first is completed (green checkmark), the second is active (blue dot), and the third is pending (grey dot). Below the progress bar, the text 'Navigation Menu' is centered. Under 'Navigation Preference', there are three radio button options: 'Do not associate this page with a navigation menu entry' (selected), 'Create a new navigation menu entry', and 'Identify an existing navigation menu entry for this page'. At the bottom left are '<' and 'Cancel' buttons, and at the bottom right is a 'Next >' button.

Ilustración 38: Configuración de navegación

Paso 4: Configuración de datos a mostrar. Escogemos la fuente de datos y datos concretos a visualizar en el informe. En el ejemplo, tomamos la tabla de empleados y todas sus columnas, por simplicidad. Este punto es vital para la correcta configuración del informe.

The screenshot shows the 'Create Classic Report' dialog box with the title bar and a close button. The progress bar now has two completed steps (green checkmarks) and one active step (blue dot). The text 'Report Source' is centered. Under 'Source Type', 'Table' is selected. Below, 'Table / View Owner' is set to 'OEHR' and 'Table / View Name' is set to 'OEHR_EMPLOYEES (table)'. Under 'Select Columns', a list of columns is shown: EMPLOYEE_ID (Number), FIRST_NAME (Varchar2), LAST_NAME (Varchar2), EMAIL (Varchar2), PHONE_NUMBER (Varchar2), HIRE_DATE (Date), JOB_ID (Varchar2), and SALARY (Number). At the bottom left are '<' and 'Cancel' buttons, and at the bottom right is a 'Create' button.

Ilustración 39: Configuración de datos a visualizar

Finalmente confirmamos la creación con el botón *Create*. Al igual que en casos anteriores, entramos en la ventana de *Page Designer* donde se puede editar el informe y ejecutar desde la opción (botón) en la parte



superior derecha (*Save and Run Page*). Este tipo de informe es “clásico” en el sentido que solo visualiza y **no permite ninguna interacción ni ajuste adicional** por parte del usuario.

3.8 Creación de informes interactivos

A la hora de crear un informe interactivo se selecciona la opción de *Interactive Report*. Al igual que los informes previos de tipo *Classic Report* (se crean siguiendo pasos muy similares) permiten visualizar datos, pero **además añaden opciones de personalización del informe por parte del usuario** como:

- añadir o eliminar columnas (*columns*)
- establecer filtros (*filter*)
- personalizar datos (ordenar, agregar, computar o realizar un *flashback*)
- personalizar formatos (control de ruptura, colores condicionales, filas por página).
- etc.

Como podemos ver en la Ilustración 40, el usuario, a través de un menú de acciones disponible en el propio informe, puede cambiar y personalizar las opciones de visualización. Esto hace que el informe sea precisamente “**interactivo**”, dado que es el usuario el que lo configura a su gusto.

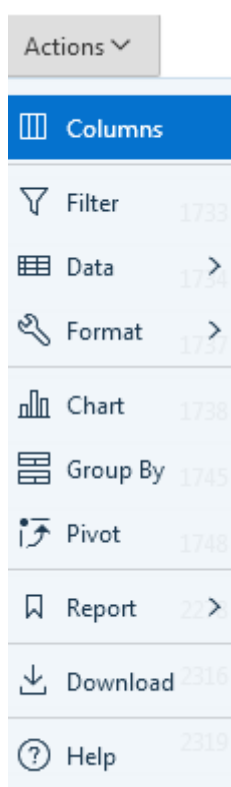


Ilustración 40: Opciones de personalización con menú desplegable

Una vez personalizados, se pueden guardar también las opciones del informe, que aparecerán en la lista de pestañas para posteriores usos.



3.9 Creación de gráficos

APEX permite la creación de un tipo especial de página con informe visual denominado *Chart*. Se corresponde con un gráfico generado a partir de los datos almacenados en el esquema actual.

Para ello seleccionamos en el tipo de página inicial, la opción *Chart*.

Paso 1: Selección del tipo de gráfico. En este ejemplo concreto seleccionamos como *Chart Type*, un gráfico de barras.

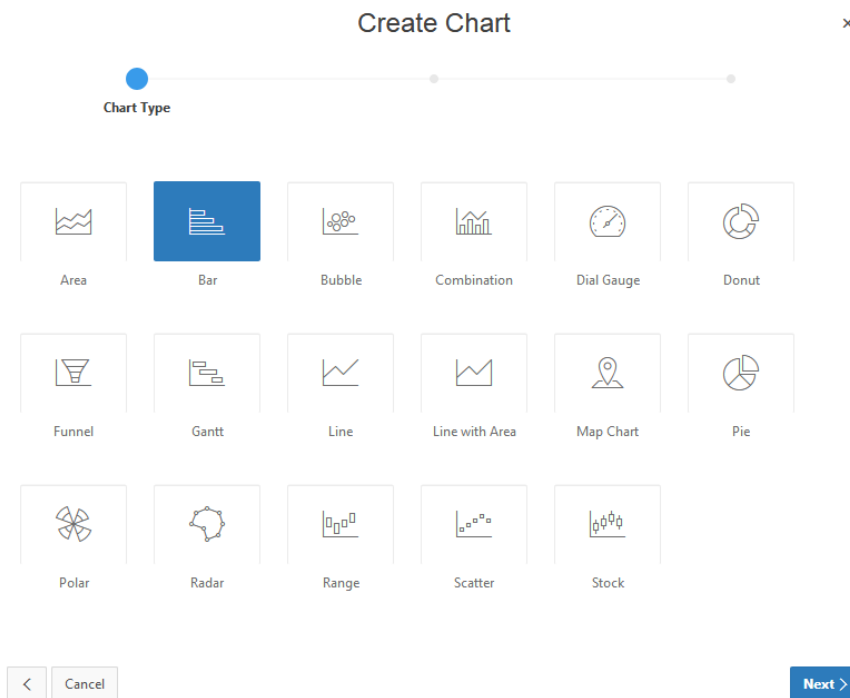


Ilustración 41: Creación de un gráfico

Paso 2: Configuración de página. Seguimos con página Normal, sin migas de pan.

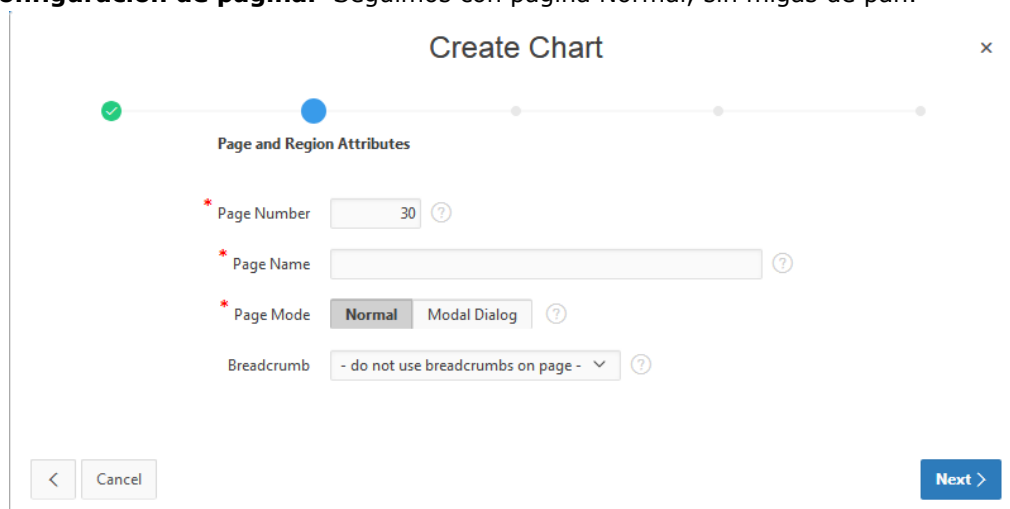


Ilustración 42: Personalización de página



Paso 3: Configuración de navegación: se ajustan las opciones de menú relativas a la página actual, sin enganchar con el menú.

Create Chart

Navigation Menu

Navigation Preference

- ☒ Do not associate this page with a navigation menu entry ?
- ☐ Create a new navigation menu entry
- ☐ Identify an existing navigation menu entry for this page

< Cancel

Next >

Ilustración 43: Creación del chart

Paso 4: Selección de los datos a mostrar. Se definen las fuentes de datos del gráfico. De nuevo, este punto es vital en la definición del gráfico. Aquí **es más habitual trabajar sobre una consulta SQL personalizada** para la generación del gráfico.

Create Chart

Source

Source Type

- ☐ Table
- ☒ SQL Query ?
- ☐ ?

* SQL Query ?

```
1 select OEHR_PRODUCTS.PRODUCT_NAME as PRODUCT_NAME,
2     OEHR_PRODUCTS.LIST_PRICE as LIST_PRICE
3 from OEHR_PRODUCTS OEHR_PRODUCTS
```

< Cancel

Next >

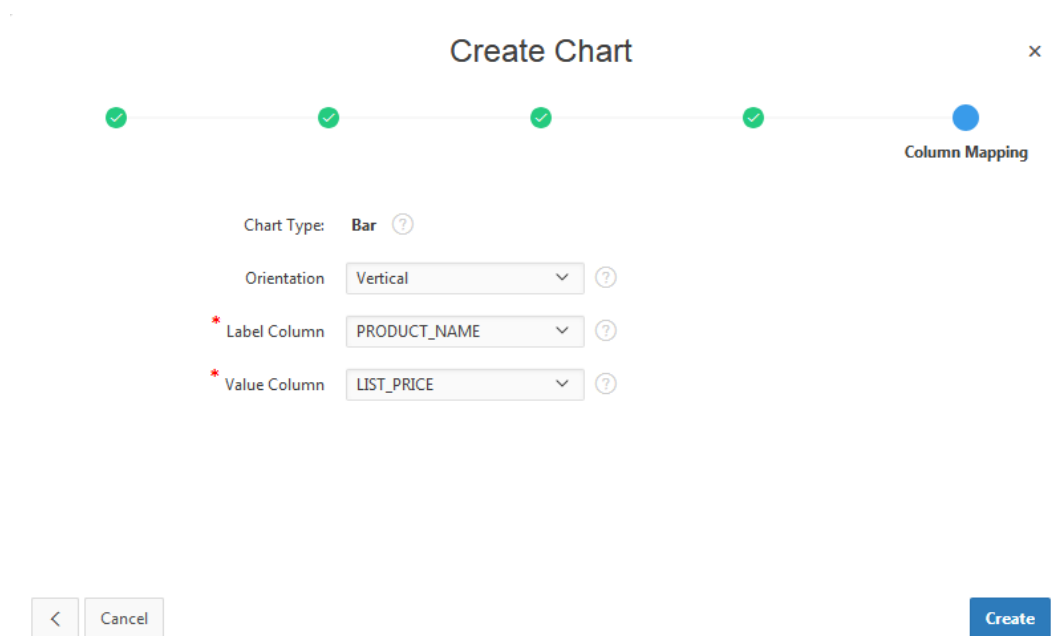
Ilustración 44: Configuración de la consulta SQL

En nuestro ejemplo hemos elegido una consulta SQL (muy simple por aligerar el ejemplo, en la práctica real la consulta estará más elaborada, pudiendo incluir funciones, agrupamientos, *joins*, etc). Es muy importante **generar correctamente esta sentencia** para que el gráfico funcione correctamente, necesitando siempre un **par de columnas**, una que proporciona la **columna numérica** (valor) y otra la



etiqueta. Por lo tanto se recomienda probar siempre primero la consulta (se puede probar desde SQL*Plus o SQLDeveloper o bien construir con el propio *Query Builder* integrado en APEX en el icono del martillo).

Paso 5: Configuración de columnas. Se configuran cuestiones relativas a qué columna de la consulta es la **etiqueta**, y cuál es el **valor** a mostrar. De nuevo, hay que poner especial cuidado en este punto, en el ejemplo concreto, la etiqueta es el nombre de producto y el valor el precio.



The screenshot shows the 'Create Chart' dialog box in APEX. At the top, there is a progress bar with five steps, each marked with a green checkmark. The fifth step, 'Column Mapping', is highlighted with a blue circle. Below the progress bar, the 'Chart Type' is set to 'Bar'. The 'Orientation' is set to 'Vertical'. The 'Label Column' is set to 'PRODUCT_NAME' and the 'Value Column' is set to 'LIST_PRICE'. There are 'Cancel' and 'Create' buttons at the bottom.

Ilustración 45: Ajuste de columnas para etiqueta y valor en el gráfico

Finalmente confirmamos la creación con el botón *Create*. Vamos al diseñador de página y ejecutamos para ver el resultado visual de nuestro *Chart*⁴.

4 El resultado del gráfico puede ser complicado de ver, dado el volumen de datos. Se deja a discreción del alumno refinar la consulta SQL para mostrar menos filas (Ej: productos cuyo LIST_PRICE > 1000). Desde la ventana del *Page Designer* en la parte derecha la consulta SQL es editable.



4. Componentes compartidos

A la hora de crear aplicaciones y páginas, podemos tener elementos o componentes comunes entre dichas páginas. Todos estos componentes se gestionan desde el botón de *Shared Components*.

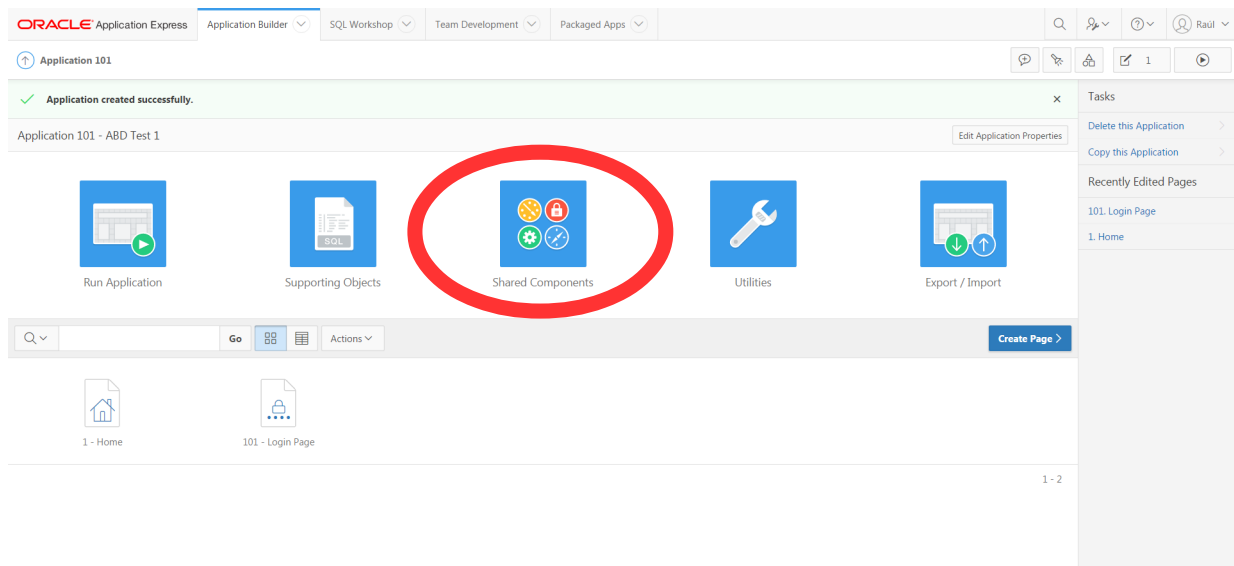


Ilustración 46: Menú de aplicación con botón *Shared Components*

Si seleccionamos el botón de *Shared Components*, nos aparecerá una nueva página con toda una serie de menús y elementos compartidos para personalizar.

4.1 Listas de valores

En muchas ocasiones se muestran valores en los formularios o informes que son identificadores. Dichos identificadores al usuario **no siempre le dan la información precisa o conveniente**, sin embargo esos identificadores están asociados un **texto que sí es descriptivo**.

Para enlazar y sustituir el identificador, por su descripción correspondiente, se utiliza en APEX el concepto de **listas de valores (List of Values o sus siglas LOV)**.

Las listas de valores pueden ser estáticas (cuando los identificadores y valores son estáticos y conocidos *a priori*, por ejemplo los meses del año) o dinámicas, cuando los valores se toman de una tabla que puede modificarse a lo largo del tiempo (por ejemplo, NIF como identificador y apellido como descriptor de una tabla de clientes que cambia y se modifica a lo largo del tiempo).

En primer lugar seleccionamos en el menú *Other Components* (parte derecha, esquina superior) la opción *List of Values*. Inicialmente la lista estará vacía.



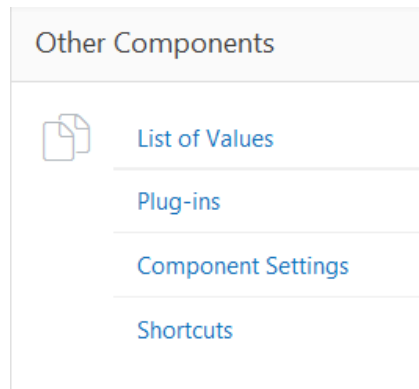


Ilustración 47: Menú con opción lista de valores

Para crear una lista de valores se realizan los siguientes pasos. Seleccionamos crear una nueva en el botón *Create*.

Paso 1. Crear una lista de valores. Seleccionamos si queremos crearla de cero (*from scratch*) o bien a partir de la copia de una lista de valores previa. En el ejemplo, partiremos de cero.

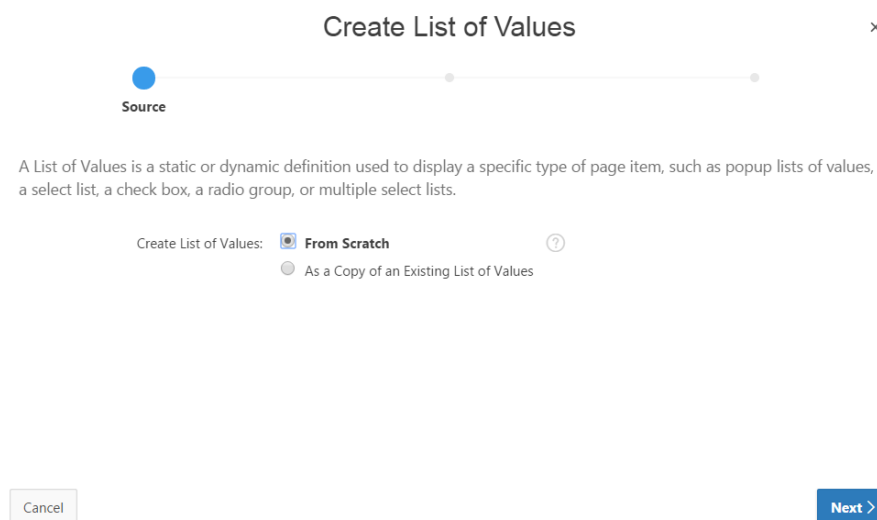


Ilustración 48: Creación de una lista de valores

Paso 2. Selección del nombre y tipo. Elegimos el nombre de la lista y si es estática (con valores prefijados) o dinámica (cargando datos dinámicamente de la base de datos). Se omite la captura de la creación de una lista estática. Básicamente consiste en dos columnas de campos a introducir, por un lado el identificador a utilizar (*Return value*) y por otro, el texto que lo sustituye y se muestra en pantalla (*Display value*).

Aunque en el ejemplo simplemente se ha indicado como nombre Clientes es conveniente seguir alguna convención de nombres, como por ejemplo utilizar un sufijo LOV (Ej: `Clientes_LOV`). De esta forma queda claro que papel juega este elemento (y que no es una tabla ni vista).



Create List of Values ×

Name and Type

Static lists are based on predefined pairs of display and return values. Dynamic lists are based on a SQL query you write that selects values from a table.

* Name: ?

Type: ☐ Static ? ☒ **Dynamic**

Next >

Ilustración 49: Elección del tipo de lista (estática vs. dinámica)

Paso 3. Definición de la consulta SQL. Se construye directamente o bien con el asistente (botón *Create Dynamic List of Values*). La consulta debe tener la siguiente estructura con dos columnas:

- la primera columna devuelve el campo descripción o etiqueta **a mostrar en pantalla**
- la segunda columna contiene habitualmente el identificador o clave primaria que identifica a la fila y que es el valor que realmente se maneja internamente.

Es importante que las listas estén siempre ordenadas por la primera columna, para facilitar el manejo de los datos por parte del usuario.

Create List of Values ×

Query or Static Values

Enter a SQL query that returns two columns. The first column is the display value. The display value is the value you see in a list of values. This column should be aliased and a different name than the return column. The second column is the return value. The return value is the value returned when the display value is selected. The return column should be aliased if it includes any operations. Use bind variable syntax within your SQL query to reference the session state of the application or page items.

List of Values Name: **CLIENTES** ?

* Query (SELECT DISPLAY_VALUE, RETURN_VALUE FROM...):

```
select CUST_LAST_NAME as display_value, CUSTOMER_ID as return_value
from OEHR_CUSTOMERS
order by 1
```

[Create Dynamic List of Values](#)

> **Examples**

Create List of Values

Ilustración 50: Definición de la consulta SQL

Paso 4. Enlazar la lista de valores en un formulario o informe. Una vez creada la lista de valores se guarda, pero es necesario utilizarla desde formularios o informes. Puede ser reutilizada cuantas veces se quiera en aquellas páginas donde se necesite. En el ejemplo mostrado en la siguiente figura, tenemos una LOV creada sobre los clientes con su apellido y su ID de cliente. Para sustituir en pantalla el id. por su apellido, debemos seleccionar el campo en un formulario sobre cliente, y en la zona derecha de propiedades:

- seleccionar como tipo del campo *Select List*



- elegir en el bloque *List of Values*, como valor de *Type*, en *Shared Components*.
 - como lista de valores escogemos la que acabamos de crear (Ej: CLIENTES⁵)
 - elegimos mostrar *null value* (yes)
 - damos un valor (por ejemplo --) para el valor nulo. Así se muestra -- cuando no tiene asociado un cliente y el campo vale nulo.

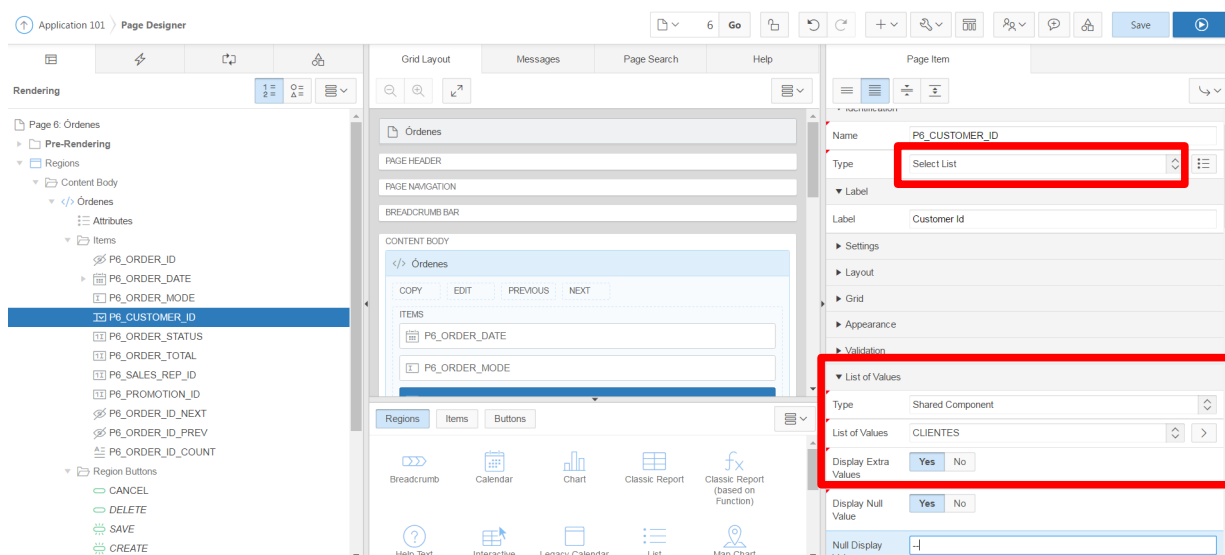


Ilustración 51: Captura de asociación a una lista de valores desde un formulario

Esta operación se podría repetir en todos aquellos formularios o informes donde estemos mostrando identificadores de clientes asociados, y sin embargo queda mucho mejor de cara al usuario, sustituir dichos valores por sus correspondientes apellidos.

Podemos mejorar la visualización de los datos modificando la consulta que recoge los datos de las tablas. Así, si vamos a la configuración de la LOV podremos utilizar una consulta SQL más compleja, para asegurarnos que mostramos toda la información disponible, concatenando valores de apellido y nombre. Revisa la Ilustración 52 para más detalles.

5 En la práctica deberíamos llamar a la lista CLIENTES_LOV para seguir una convención de nombres y no confundirla con otros elementos.



ORACLE App Builder SQL Workshop Team Development Packaged Apps

Application 102 \ Shared Components \ Lists of Values \ Edit

List of Values: CLIENTES

Cancel Delete **Apply Changes**

Show All Name Subscription Source: Lists of Values Query

Name

* List of Values Name CLIENTES

Subscription

Reference Master List of Values From Refresh

This is the "master" copy of this List of Values.

No Lists of Values subscribe to this List of Values.

Source: Lists of Values Query

Query

```

1 select CUST_LAST_NAME || ' ; ' || CUST_FIRST_NAME as display_value, CUSTOMER_ID as return_value
2   from OEHR_CUSTOMERS
3  order by 1

```

List of Values

Lists of Values can be static, hard coded display values and return values, or dynamic, based on a query. Lists of Values can also be based on a PL/SQL function that returns a query allowing the dynamic query to be dynamically constructed.

Usage

This List of Values is not used.

Ilustración 52: Configuración de una LOV más avanzada concatenando campos

4.2 Migas de pan (*Breadcrumbs*)

Cuando navegamos en una aplicación web se puede mostrar en pantalla los pasos seguidos para llegar a la página actual. Además si se crean hiperenlaces, se permite la navegación hacia atrás.

Por ejemplo, el propio APEX utiliza esto continuamente, mostrando los pasos dados en la parte superior de la pantalla. En la siguiente figura, sabemos que estamos situados en la ventana *Entries*, y que si pulsamos sobre *Breadcrumbs* vamos un paso hacia atrás, y si pulsamos en *Shared Components*, dos pasos hacia atrás, en el historial de pasos dados en la aplicación.

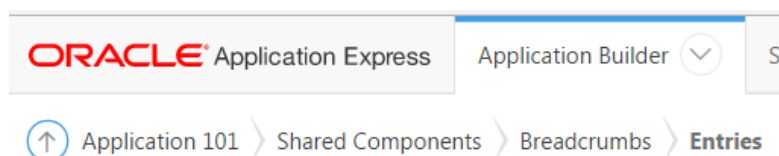


Ilustración 53: Ejemplo de miga de pan en APEX

Estos elementos en pantalla se denominan **migas de pan** (*Breadcrumbs*) y en APEX se pueden crear, configurar y asociar a las páginas. De esta forma se muestra al usuario las migas de pan, facilitando su navegación hacia atrás siguiendo el histórico de sus pasos.

Las migas de pan se gestionan desde el menú de navegación, dentro de *Shared Components*:



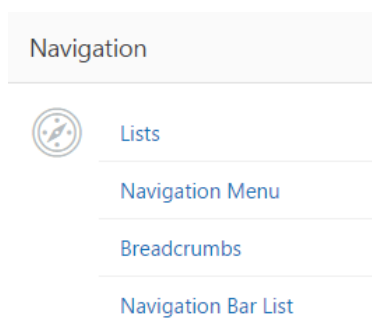


Ilustración 54: Menú de navegación

En primer lugar se muestran las migas de pan ya creadas:

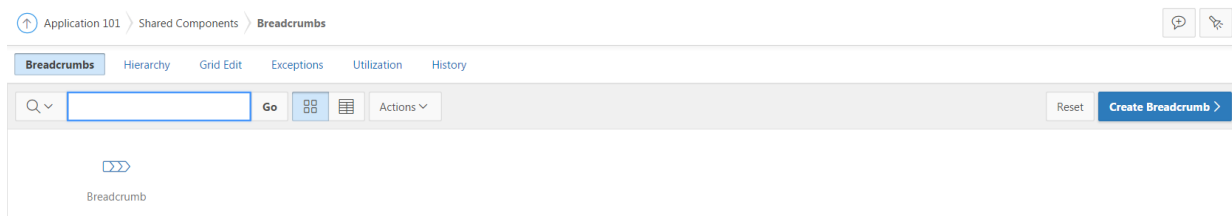


Ilustración 55: Migas de pan creadas

Dentro de la miga de pan, se permite crear una navegación jerárquica con cualquier número de niveles. Se crea una miga de pan (*breadcrumb*) y luego entradas (*breadcrumb entries*), con su correspondiente nombre. Cada entrada se asocia a un página, e indica su padre dentro de la jerarquía, permitiendo la navegación hacia "arriba".

Seleccionando la miga de pan (en el ejemplo *Breadcrumb*) podemos visualizar y editar las entradas asociadas, pudiendo modificar la navegación entre páginas.

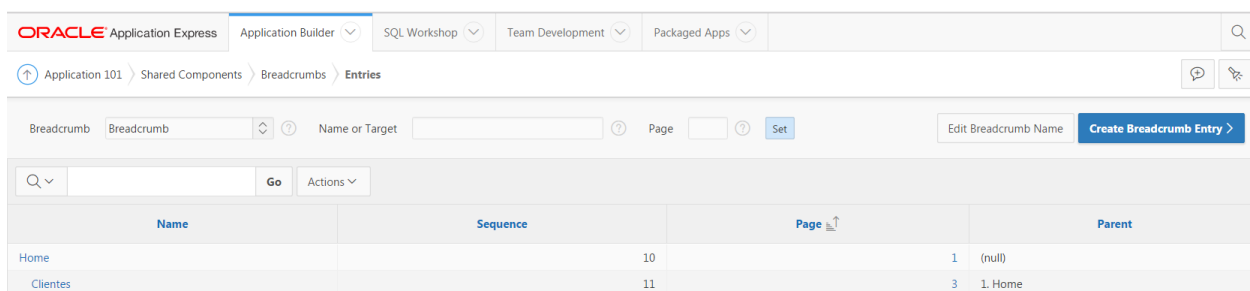


Ilustración 56: Edición de entradas en una miga de pan

Las migas de pan, una vez creadas, se suelen incluir en regiones adicionales en las páginas. En los ejemplos previos, no se han creado, por motivos de brevedad, entradas en la miga de pan por defecto (*Breadcrumb*) que ya teníamos creada, pero se puede (debe) probar a crear nuevas páginas, construyendo el camino de navegación que se requiera.

Es conveniente tener dibujado el mapa de páginas que vamos a construir, para intentar introducir ya inicialmente las migas de pan y su navegación. Aunque posteriormente, sí que se pueden editar y cambiar dichas entradas, es más fácil hacerlo en la creación inicial de páginas.

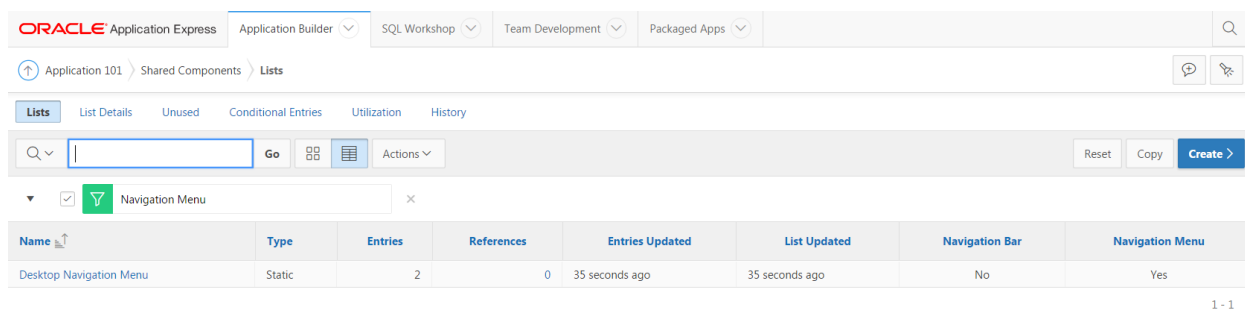


4.3 Menú de navegación

Para movernos entre las distintas opciones de una aplicación, una forma habitual es utilizar menús. En APEX también se cuenta con este elemento, para configurar la navegación entre páginas. Para ello se crea un menú de navegación inicial (nuestra página *Home* tenía creada una entrada en dicho menú).

Para ver la configuración actual, seleccionamos *Navigation Menu* en el menú *Navigation*, dentro de *Shared Components*.

Se nos mostrará la lista de menús, en concreto nuestro menú de navegación de escritorio.

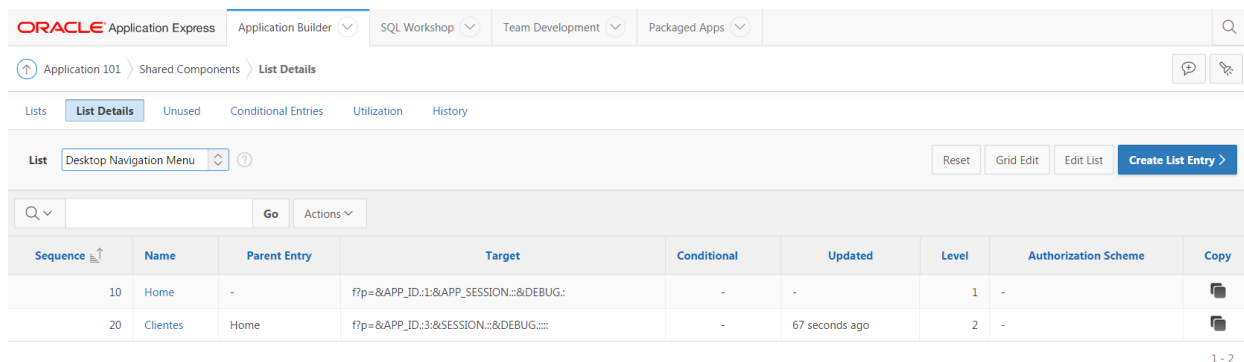


The screenshot shows the Oracle APEX interface for managing Shared Components. The breadcrumb trail is Application 101 > Shared Components > Lists. The 'Lists' tab is active, showing a table of navigation menus. The table has columns: Name, Type, Entries, References, Entries Updated, List Updated, Navigation Bar, and Navigation Menu. One entry is visible: Desktop Navigation Menu, which is Static, has 2 entries, and is the Navigation Menu for the application.

Name	Type	Entries	References	Entries Updated	List Updated	Navigation Bar	Navigation Menu
Desktop Navigation Menu	Static	2	0	35 seconds ago	35 seconds ago	No	Yes

Ilustración 57: Menús de navegación

Si seleccionamos *List Details* veremos las distintas entradas de menú. En el ejemplo, podemos ver que tenemos dos.



The screenshot shows the 'List Details' view for the 'Desktop Navigation Menu'. It displays a table of menu entries with columns: Sequence, Name, Parent Entry, Target, Conditional, Updated, Level, Authorization Scheme, and Copy. Two entries are listed: 'Home' (Sequence 10) and 'Clientes' (Sequence 20). The 'Clientes' entry is the current selection.

Sequence	Name	Parent Entry	Target	Conditional	Updated	Level	Authorization Scheme	Copy
10	Home	-	f?p=&APP_ID.:1:&APP_SESSION.:&DEBUG.:	-	-	1	-	
20	Clientes	Home	f?p=&APP_ID.:3:&SESSION.:&DEBUG.::	-	67 seconds ago	2	-	

Ilustración 58: Entradas de menú

Si pulsamos sobre la entrada de *Clientes*, podemos editar su configuración. Actualmente tendríamos una configuración como esta:



Ilustración 59: Edición de una entrada de menú

Como podemos observar, se indica la entrada padre *Home* (de la que cuelga en este caso) siendo un submenú dentro de dicha entrada (para visualizar *Clientes* debemos pulsar en la flecha hacia abajo, pudiendo ocultarse). Cuando el usuario seleccione dicha entrada de menú (*Clientes*) se le redirige a la página 3 como *Target* (recordemos que en APEX se utilizan los números de página, aunque no sea la solución ideal). En pantalla veríamos algo similar a la siguiente ilustración:

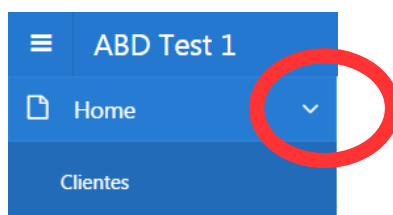


Ilustración 60: Menú y submenú

Si queremos subir de nivel o sacar esta entrada al mismo nivel que *Home*, en el campo *Parent List Entry* lo dejamos en valor -- *No Parent List Item* --. De esta forma indicaríamos que la entrada va a un nivel raíz, en el menú. Ahora se visualizarían siempre las dos entradas, tal y como se muestra a continuación.

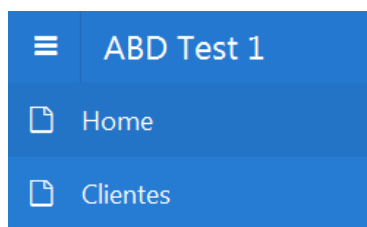


Ilustración 61: Menú con todas las entradas al mismo nivel

Al igual que nos ocurre con las migas de pan, es una cuestión de diseño y personal, el decidir cómo vamos a organizar los menús para nuestra aplicación.



5. Anexo

5.1 Exportación e importación en APEX de aplicaciones

Previamente a la exportación de una aplicación se deben configurar lo que se denominan *Supporting Objects*.

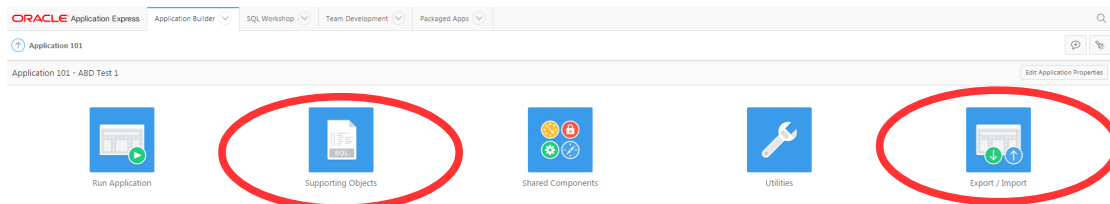


Ilustración 62: Barra de herramientas de acceso a Supporting Objects de una aplicación

Una vez seleccionado se visualizan las opciones adicionales de exportación de objetos de la base de datos:

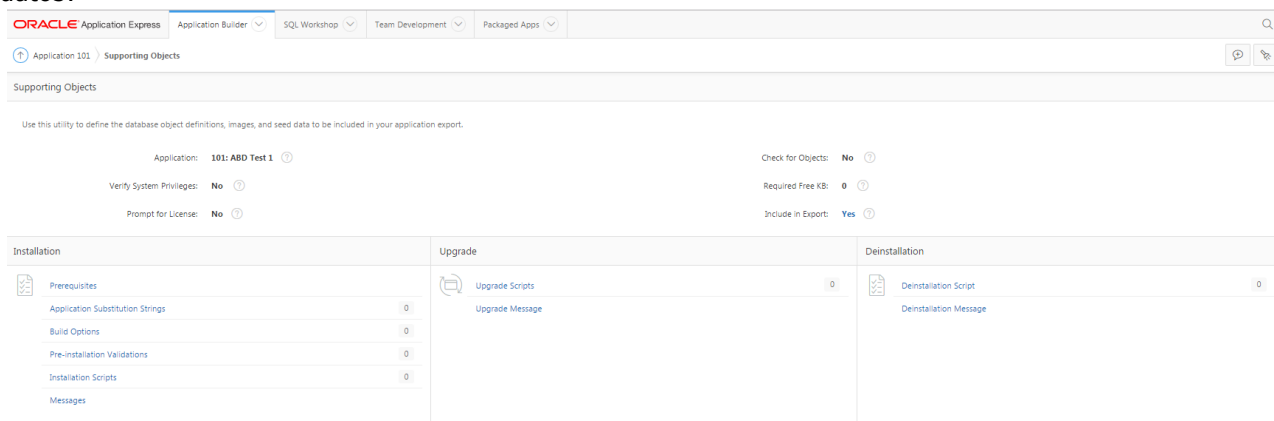


Ilustración 63: Ventana principal de personalización de Supporting Objects

Supporting Objects permite incluir **las definiciones de objetos de la base de datos, imágenes, ficheros y datos a cargar** para ser incluidos posteriormente en la exportación. Incluyendo scripts SQL para la instalación, actualización o desinstalación.

Por otro lado permite personalizar cuestiones típicas en un instalador, como análisis de prerequisites, licencias, etc. junto con los mensajes a mostrar al usuario. **Aunque en esta asignatura no se realizarán labores tan complejas, es conveniente conocer que existen estas opciones, por si se sigue trabajando en el futuro con esta herramienta.**

Para la exportación de la aplicación creada a partir de *Application Builder* se puede realizar desde esta misma ventana, en el botón de la derecha *Export*, mostrándose las siguientes opciones de exportación.



Ilustración 64: Opciones de exportación

O bien desde el botón de *Import/Export* (ver Ilustración 62) que nos permite elegir entre importar o exportar una aplicación.

Muy importante: si queremos que se incluyan los elementos definidos previamente en *Supporting Objects*, marcamos su desplegable con *Yes*.

Como resultado de la exportación, se genera un script `.sql`⁶ con la **aplicación**, que posteriormente puede ser importado en otro servidor APEX que tenga un *workspace* equivalente. Es importante señalar que **estamos exportando aplicaciones, NO un workspace**, como se hizo en la sesión previa de prácticas.

Para importar la aplicación se selecciona *Import*, y se selecciona el fichero `.sql` anteriormente exportado. Se asignará un nuevo ID para no pisar la aplicación actual, en caso de que hubiera coincidencia.

Este proceso de exportación es el que se debe seguir a la hora de entregar la aplicación web solicitada en las entregas.

6 Es un fichero de texto, por lo que se puede ver su contenido con cualquier editor, pero no se sugiere su modificación, puesto que el fichero podría quedar dañado para su posterior importación en APEX.



Licencia

Autores: Jesús Maudes & Raúl Marticorena & Mario Martínez

Área de Lenguajes y Sistemas Informáticos

Departamento de Ingeniería Civil

Escuela Politécnica Superior

UNIVERSIDAD DE BURGOS

2020



Este obra está bajo una licencia Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Unported. No se permite un uso comercial de esta obra ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula esta obra original

Licencia disponible en <http://creativecommons.org/licenses/by-nc-sa/4.0/>

