

Hoja 4: Ejercicios colecciones y paquetes

1. Dada la consulta que muestra el id_categoria y la descripción de la tabla CATEGORIAS, escribe un bloque pl/sql que cargue en un varray el id de la categoría y la descripción mostrando por pantalla el contenido de dicho varray.
2. Reescribe el ejercicio anterior usando una tabla anidada en lugar de un varray. Ten en cuenta que no es necesario inicializar a NULL varios elementos, sino inicializar con una lista vacía y, después, podemos crear nuevos elementos en el bucle de carga usando el método EXTEND.
3. Reescribe el ejercicio 1 usando una tabla indexada y los atributos disponibles para recorrer la tabla. En este caso hay que mostrar en pantalla el id y la descripción.
4. Escribir un paquete completo para gestionar las categorías. El paquete se llamará ges_cat y deberá incluir al menos los siguientes programas:
 - Insertar_nueva_categoria: permite insertar una categoría nueva. El procedimiento recibe la descripción y le asigna el id siguiente que esté libre.
 - Borrar_categoria: permite borrar una categoría. El procedimiento recibirá el id_categoria y la eliminará.
 - Visualizar_datos: visualizará los datos de una categoría cuyo id se le pasará en la llamada. Además, deberá decirnos cuantos elementos hay de cada categoría.
 - Visualizar_datos: versión sobrecargada del procedimiento anterior que en lugar del id de la categoría recibirá la descripción. Realizará una llamada a la función buscar_por_descrip que se indica en el apartado siguiente.
 - buscar_por_descrip: función local al paquete. Recibe la descripción de la categoría y devuelve el id de la misma.

5. Escribir un paquete completo para gestionar los empleados. El paquete se llamará `gest_emple` e incluirá, al menos, los siguientes subprogramas:
 - `insertar_nuevo_emple`. Recibirá todos los datos necesarios para insertar un nuevo empleado, excepto el `id_empleado` que se le asignará el siguiente al último.
 - `modificar_dir_emple`
 - `modificar_ciudad_emple`. Se deberá verificar si la ciudad nueva pertenece a las poblaciones y se debe actualizar el `cod_postal` y el `código_area` poniendo en el mismo el `cod_prov` correspondiente de la tabla poblaciones.
 - `visualizar_datos_emple`.
 - `visualizar_datos_emple`. Versión sobrecargada del procedimiento que recibirá el nombre y los apellidos del empleado.
 - `buscar_emple_por_nombre`. Función local que recibe el nombre y los apellidos y devuelve el `id_empleado`.
6. Escribir un paquete completo para gestionar los productos. El paquete se llamará `gest_productos` e incluirá, al menos, los siguientes subprogramas:
 - `nuevo_producto`. Deberá recibir como parámetro todos los datos necesarios y se le asignará como nuevo número de producto el siguiente al mayor que hay actualmente. Deberá actualizar también la tabla `producto proveedores`, añadiendo una decena más al último `id` del repartidor actual, el `precio_por_mayor` será un 25% más que el más barato y los días de envío serán 3.
 - `modificar_precio_venta`.
 - `modificar_stock`
 - `visualizar_datos_productos`. También se incluirá una versión sobrecargada del procedimiento que recibirá el nombre del producto.
 - `buscar_producto_por_nombre`. Función local que recibe el nombre y devuelve el `numero_producto`.