

Bases de Datos	Supuesto Práctico de Evaluación
UT5: PL/SQL	Fecha: 01/06/2023

### TIPO A

Dadas las tablas **cliente**, **reserva** y **coche** de la base de datos “**cicar.sql**”, se pide realizar en PL/SQL los ejercicios que se detallan en las siguientes líneas:

#### 1. Devoluciones de vehículos (2,5 puntos)

Desarrolle **un procedimiento almacenado** que nos muestre por pantalla información obtenida a partir de las tablas **reserva**, **cliente** y **coche**. Para cada registro de la **consulta** se deberá mostrar la información que se detalla a continuación:

- Mostrar en vez del **código del cliente** su **nombre**.
- Mostrar en vez del **código del coche** su **marca**.
- Mostrar la **fecha de reserva**.
- Mostrar el número de **días reservados**.
- Mostrar la “**fecha\_devolucion**” cuyo valor se ubicará en una **variable** o en un **campo calculado** de la consulta y se obtendrá como la suma de la **fecha de reserva** y el número de **días** reservados (la **fecha** en la que el **cliente** deberá devolver el **coche**). El procedimiento deberá denominarse **PROC\_RESERVA\_DEVUELVE**.

#### 2. Tiempo de alta clientes (2 puntos)

Desarrolle **una función** que dado un **código** de un cliente pasado como parámetro, devuelva los años y los días que lleva registrado en el sistema. La función deberá denominarse **FUN\_CLIENTE\_ALTA**.

#### 3. Bajas de clientes (3 puntos)

Desarrolle **una procedimiento almacenado** que muestre para cada cliente su **código** y la **fecha de su última reserva**. Si el cliente lleva más de **un año** sin reservar vehículos, entonces se deberá eliminar de la **tabla cliente**. Antes de borrar, debemos guardar los datos en **otra tabla** denominada **clientes\_borrados** donde se almacenará el **código** del cliente, el **nombre** del cliente, el **usuario** del sistema que hace la operación y **fecha** de borrado. El procedimiento deberá denominarse **PROC\_BAJA\_CLIENTES**.

#### 4. Pagar reserva con crédito (2.5 puntos)

Crear un disparador que actualice el valor del campo **crédito** del **cliente** al insertar una entrada en la tabla **reserva**. Si el cliente no tiene suficiente crédito, entonces **no cambiará** el valor y se lanzará la correspondiente **excepción impidiendo la inserción** en la tabla **reserva**. El disparador deberá denominarse **DISP\_COBRAR**.

**NOTA:** El importe a pagar por el cliente se calculará como el **precio** del coche al día multiplicado por el número de **días** reservado. **cliente.crédito = cliente.crédito - (coche.preciodía \* reserva.días)**.

**NOTA:** Para facilitar la resolución de los ejercicios, el alumno puede **crear todas aquellas vistas o rutinas auxiliares que considere oportunas**. Compruebe que están las rutinas creadas y activas probando a activarlas/ejecutarlas y desactivarlas. Realice las operaciones SQL oportunas sobre las tablas creadas, de forma que se compruebe la funcionalidad del código creado.