**-lPL/SQL Ejercicios**

1. EJER\_38: Función que recibe un código (de asignatura) y devuelve cuantas asignaturas hay con ese código.

**create or replace FUNCTION EJER\_38 (COD\_ASIGNATURA NUMBER)**

**RETURN NUMBER**

**IS**

**NUMEROASIGNATURAS NUMBER;**

**BEGIN**

**SELECT COUNT(\*) INTO NUMEROASIGNATURAS**

**FROM ASIGNATURAS**

**WHERE COD=COD\_ASIGNATURA;**

**RETURN NUMEROASIGNATURAS;**

**END;**

1. EJER\_39: Procedimiento para dar de alta una nueva asignatura. Recibe un código de asignatura y un nombre de asignatura y:

* Llama a la función EJER\_38 para comprobar si ya existe ese código de asignatura. Si existe muestra un mensaje de error y no hace nada más.
* Si el código no existe comprueba si ya hay alguna asignatura con el nombre que recibe. Si la hay muestra un mensaje de error y no hace nada más.
* Si todo ha ido bien da de alta una nueva asignatura con el código y nombre recibido.

Utilizar este procedimiento para dar de alta la asignatura 5 con nombre FOL

**create or replace PROCEDURE EJER\_39(COD\_ASIGNATURA NUMBER, NOMBRE\_A VARCHAR2)**

**AS**

**NOMBREASIGNATURA NUMBER;**

**BEGIN**

**IF EJER\_38(COD\_ASIGNATURA)>0 THEN**

**VER('ERROR CODIGO DE ASIGNATURA EXISTENTE');**

**ELSE**

**SELECT COUNT(\*) INTO NOMBREASIGNATURA**

**FROM ASIGNATURAS**

**WHERE NOMBRE=NOMBRE\_A;**

**IF NOMBREASIGNATURA>0 THEN**

**VER('ERROR NOMBRE ASIGNATURA EXISTENTE');**

**ELSE**

**INSERT INTO ASIGNATURAS VALUES(COD\_ASIGNATURA, NOMBRE\_A);**

**END IF;**

**END IF;**

**END;**

1. EJER\_40: Procedimiento para dar de alta una nueva nota. Recibe un dni de alumno, un código de asignatura y una nota.

* Llama a la función EJER\_38 para comprobar si existe ese código de asignatura. Si NO existe muestra un mensaje de error y no hace nada más.
* Si el código existe comprueba si hay algún alumno con ese dni. Si no lo hay muestra un mensaje de error y no hace nada más.
* Comprueba que la nota sea un número entre 0 y 10. Si no lo es muestra un mensaje de error y no hace nada más.
* Comprueba en NOTAS si el alumno ya tiene una nota en la asignatura recibida.
  + Si existe modifica la nota con la nota recibida. (UPDATE)
  + Si no la tiene da de alta la nota. (INSERT)

Ejemplos de ejecución:

Execute EJER\_40 ('55555555P ', 9, 8)

Muestra: No la asignatura

Execute EJER\_40 ('55555555P ', 1, 8)

Muestra: No existe el alumno

Execute EJER\_40 (' 12344345H ', 1, 12)

Muestra: No es una nota válida

Execute EJER\_40 (' 12344345H ', 1, 8)

Modifica la nota que ya existe para ese código y alumno con un 8

Execute EJER\_40 (' 12344345H ', 5, 8)

Da de alta la nota

**create or replace PROCEDURE EJER\_40(PDNI\_ALUMNO VARCHAR2,PCOD\_ASIGNATURA NUMBER, PNOTA NUMBER)**

**AS**

**NUMERODEALUMNOS NUMBER;**

**NOTAEXISTENTE NUMBER;**

**BEGIN**

**IF EJER\_38(PCOD\_ASIGNATURA)=0 THEN**

**VER('ERROR CODIGO DE ASIGNATURA NO EXISTENTE');**

**ELSE**

**SELECT COUNT(\*) INTO NUMERODEALUMNOS**

**FROM ALUMNOS**

**WHERE DNI=PDNI\_ALUMNO;**

**IF NUMERODEALUMNOS=0 THEN**

**VER('ERROR NO EXISTE ESTE ALUMNO');**

**ELSE**

**IF PNOTA NOT BETWEEN 0 AND 10 THEN**

**VER('ERROR NOTA NO ESTA ENTRE 0 Y 10');**

**ELSE**

**SELECT COUNT(\*) INTO NOTAEXISTENTE**

**FROM NOTAS**

**WHERE DNI=PDNI\_ALUMNO**

**AND COD=PCOD\_ASIGNATURA;**

**IF NOTAEXISTENTE>0 THEN**

**UPDATE NOTAS**

**SET NOTA=PNOTA**

**WHERE DNI=PDNI\_ALUMNO**

**AND COD=PCOD\_ASIGNATURA;**

**ELSE**

**INSERT INTO NOTAS VALUES (PDNI\_ALUMNO,PCOD\_ASIGNATURA,PNOTA);**

**END IF;**

**END IF;**

**END IF;**

**END IF;**

**END;**

1. EJER\_41: Procedimiento que recibe un apellido y muestra el salario y fecha de alta del empleado correspondiente. Si no existe el empleado o si existieran varios con el mismo apellido muestra un mensaje indicándolo.

**create or replace PROCEDURE EJER\_41(PAPELLIDO VARCHAR2)**

**AS**

**EMPLEADOEXISTENTE NUMBER;**

**PSALARIO EMPLE.SALARIO%TYPE;**

**PFECHA\_ALTA EMPLE.FECHA\_ALTA%TYPE;**

**BEGIN**

**SELECT SALARIO, FECHA\_ALTA INTO PSALARIO, PFECHA\_ALTA**

**FROM EMPLE**

**WHERE APELLIDO=PAPELLIDO;**

**VER('EL SALARIO Y FECHA DE ALTA DE '||PAPELLIDO||' ES '||PSALARIO||' Y '||PFECHA\_ALTA);**

**EXCEPTION**

**WHEN NO\_DATA\_FOUND THEN**

**VER('NO EXISTE EL EMPLEADO');**

**WHEN TOO\_MANY\_ROWS THEN**

**VER('HAY MAS DE UN EMPLEADO CON ESE APELLIDO, POR AHORA NO PUEDO AYUDARTE');**

**END;**

1. EJER\_42: Procedimiento que recibe un apellido y muestra el número de departamento y el nombre del departamento al que pertenece el empleado. Si no existe el empleado o si existieran varios con el mismo apellido muestra un mensaje indicándolo.

**create or replace PROCEDURE EJER\_41(PAPELLIDO VARCHAR2)**

**AS**

**EMPLEADOEXISTENTE NUMBER;**

**PDEPT\_NO EMPLE.DEPT\_NO%TYPE;**

**PNOMBRE DEPART.DNOMBRE%TYPE;**

**BEGIN**

**SELECT DEPT\_NO INTO PDEPT\_NO**

**FROM EMPLE**

**WHERE APELLIDO=PAPELLIDO;**

**SELECT DNOMBRE INTO PNOMBRE**

**FROM DEPART**

**WHERE DEPT\_NO=PDEPT\_NO;**

**VER('EL DEPARTAMENTO ES '||PDEPT\_NO||' Y EL NOMBRE ES '||PNOMBRE);**

**EXCEPTION**

**WHEN NO\_DATA\_FOUND THEN**

**VER('NO EXISTE EL EMPLEADO');**

**WHEN TOO\_MANY\_ROWS THEN**

**VER('HAY MAS DE UN EMPLEADO CON ESE APELLIDO, POR AHORA NO PUEDO AYUDARTE');**

**END;**

—----------------------------------OTRA FORMA SIN EL EXCEPTION—--------------------------------

**CREATE OR REPLACE PROCEDURE EJER\_42(PAPELLIDO VARCHAR2)**

**AS**

**EMPLEADOEXISTENTE NUMBER;**

**PDEPT\_NO EMPLE.DEPT\_NO%TYPE;**

**PNOMBRE DEPART.DNOMBRE%TYPE;**

**BEGIN**

**SELECT COUNT(\*) INTO EMPLEADOEXISTENTE**

**FROM EMPLE**

**WHERE APELLIDO=PAPELLIDO;**

**IF EMPLEADOEXISTENTE>0 THEN**

**SELECT DEPT\_NO INTO PDEPT\_NO**

**FROM EMPLE**

**WHERE APELLIDO=PAPELLIDO;**

**SELECT DNOMBRE IN**

**TO PNOMBRE**

**FROM DEPART**

**WHERE DEPT\_NO=PDEPT\_NO;**

**VER('EL DEPARTAMENTO ES '||PDEPT\_NO||' Y EL NOMBRE ES '||PNOMBRE);**

**ELSE**

**VER('ERROR NO EXISTE DICHO EMPLEADO');**

**END IF;**

**EXCEPTION**

**WHEN NO\_DATA\_FOUND THEN**

**VER('NO EXISTE EL EMPLEADO');**

**WHEN TOO\_MANY\_ROWS THEN**

**VER('HAY MAS DE UN EMPLEADO CON ESE APELLIDO, POR AHORA NO PUEDO AYUDARTE');**

**END;**

1. EJER\_43: **Función** que recibe un nombre de departamento y devuelve el número de departamento correspondiente o NULL si no existe el departamento.

**EJER\_43 (‘VENTAS’) devuelve 30**

**EJER\_43 (‘PIRIBI’) devuelve NULL**

**CREATE OR REPLACE FUNCTION EJER\_43(NOMBREDEPART VARCHAR2)**

**RETURN NUMBER**

**IS**

**FDEPT\_NO DEPART.DNOMBRE%TYPE;**

**BEGIN**

**SELECT DEPT\_NO INTO FDEPT\_NO**

**FROM DEPART**

**WHERE DNOMBRE=NOMBREDEPART;**

**RETURN FDEPT\_NO;**

**EXCEPTION**

**WHEN NO\_DATA\_FOUND THEN**

**RETURN NULL;**

**END;**

1. EJER\_44: **Procedimiento** que recibe un nombre de departamento y un porcentaje para subir el salario a los empleados de ese departamento.

* Validar que el porcentaje está entre 0 y 100, si no es así se muestra un mensaje de error y no se hace más.
* Si el porcentaje es correcto llamar a la función anterior pasándole el nombre del departamento.
  + Si la función devuelve NULL se da un mensaje de error indicando que no existe el departamento.
  + Si no, subir el salario ese porcentaje a los empleados de ese departamento.

EJER\_44 (‘VENTAS’,120);

Mensaje de que el porcentaje es erróneo

EJER\_44 (‘piribi’,10);

Mensaje de que el departamento no existe

**create or replace PROCEDURE EJER\_44 (PNOMBREDEPT VARCHAR2, PORCENT NUMBER)**

**AS**

**NUMERO NUMBER;**

**BEGIN**

**IF PORCENT NOT BETWEEN 0 AND 100 THEN**

**VER('ERROR PORCENTAJE NO ESTA ENTRE 0 Y 10');**

**ELSE**

**NUMERO:=EJER\_43(PNOMBREDEPT);**

**IF NUMERO IS NULL THEN**

**VER('ERROR EL DEPARTAMENTO NO EXISTE');**

**ELSE**

**UPDATE EMPLE**

**SET SALARIO=SALARIO+(SALARIO\*PORCENT/100)**

**WHERE DEPT\_NO=(SELECT DEPT\_NO**

**FROM DEPART**

**WHERE DNOMBRE=PNOMBREDEPT);**

**END IF;**

**END IF;**

**END;**

1. EJER\_45: Procedimiento que recibe un apellido y si el empleado existe muestra los años que lleva en la empresa. Si no muestra un mensaje de error.

**create or replace PROCEDURE EJER\_45(PAPELLIDO VARCHAR2)**

**AS**

**EEMPLEADO NUMBER;**

**PFECHA DATE;**

**ANOS NUMBER;**

**BEGIN**

**SELECT COUNT(\*) INTO EEMPLEADO**

**FROM EMPLE**

**WHERE APELLIDO=PAPELLIDO;**

**IF EEMPLEADO>0 THEN**

**SELECT FECHA\_ALTA INTO PFECHA**

**FROM EMPLE**

**WHERE APELLIDO=PAPELLIDO;**

**ANOS:=TRUNC((SYSDATE-PFECHA)/365);**

**VER('LLEVA TRABAJANDO CON NOSOTROS ESTOS AÑOS: '||ANOS);**

**ELSE**

**VER('ERROR');**

**END IF;**

**END;**

1. EJER\_46: Procedimiento que recibe un apellido de empleado y un nuevo oficio y modifica a este empleado el oficio. Validar que exista el empleado y si existe mostrar el oficio antiguo y luego modificarlo.

**create or replace PROCEDURE EJER\_46 (PAPELLIDO VARCHAR2, POFICIO VARCHAR2)**

**AS**

**CUANTOS NUMBER;**

**BEGIN**

**SELECT COUNT(\*) INTO CUANTOS**

**FROM EMPLE**

**WHERE APELLIDO=PAPELLIDO;**

**IF CUANTOS>0 THEN**

**UPDATE EMPLE**

**SET OFICIO=POFICIO**

**WHERE APELLIDO = PAPELLIDO;**

**ELSE**

**VER('ERROR');**

**END IF;**

**END;**

1. EJER\_47: Procedimiento que muestra el salario máximo, mínimo y medio de un departamento recibido por parámetro.

**create or replace PROCEDURE EJER\_47 (PNOMBRE VARCHAR2)**

**AS**

**PMAX NUMBER;**

**PMIN NUMBER;**

**PMEDIO NUMBER;**

**BEGIN**

**SELECT MAX(SALARIO) INTO PMAX**

**FROM EMPLE**

**WHERE DEPT\_NO=(SELECT DEPT\_NO FROM DEPART WHERE DNOMBRE=PNOMBRE);**

**SELECT MIN(SALARIO) INTO PMIN**

**FROM EMPLE**

**WHERE DEPT\_NO=(SELECT DEPT\_NO FROM DEPART WHERE DNOMBRE=PNOMBRE);**

**SELECT AVG(SALARIO) INTO PMEDIO**

**FROM EMPLE**

**WHERE DEPT\_NO=(SELECT DEPT\_NO FROM DEPART WHERE DNOMBRE=PNOMBRE);**

**VER('LA MAXIMA ES '||PMAX);**

**VER('LA MINIMA ES '||PMIN);**

**VER('LA MEDIA ES '||PMEDIO);**

**END;**

1. EJER\_48: Función que recibe el nombre de una atracción y devuelve el código de la atracción si existe o nulo si no existe.

**CREATE OR REPLACE FUNCTION EJER\_48 (FNOMBRE VARCHAR2)**

**RETURN CHAR**

**IS**

**FCOD CHAR(4);**

**BEGIN**

**SELECT COD\_ATRACCION INTO FCOD**

**FROM ATRACCIONES**

**WHERE NOM\_ATRACCION=FNOMBRE;**

**EXCEPTION**

**--ZONA TRATAMIENTO EXCEPTION**

**WHEN NO\_DATA\_FOUND THEN**

**--0 FILAS Y HA FALLADO**

**RETURN NULL;**

**END;**

1. EJER\_49: Función que recibe el nombre de una atracción y devuelve el dni del encargado de la zona de dicha atracción o NULL si no existe la atracción.
2. EJER\_50: Procedimiento que recibe el nombre de una atracción y da de alta una avería en este momento para esa atracción. Primero llama a la función EJER\_48 para ver si existe. En el caso de que exista llama a la función EJER\_49 para ver el dni del encargado. Finalmente da de alta la avería con el empleado devuelto.