

## Practica procedimientos almacenados con parámetros 2

- 1) Cree un procedimiento que se llame INSETAR\_EMPLEADO, el fin de este procedimiento es insertar un nuevo empleado en la tabla EMPLOYEES, recuerde tomar todas las precauciones necesarias para hacer el insert de manera correcta.
- 2) Al momento de ejecutar el procedimiento anterior, debe visualizar desde afuera el ID del empleado.

### DECLARACION DEL SP

```
CREATE OR REPLACE PROCEDURE INSETAR_EMPLEADO (  
    P_FIRST_NAME    IN VARCHAR2,  
    P_LAST_NAME     IN VARCHAR2,  
    P_EMAIL         IN VARCHAR2,  
    P_PHONE_NUMBER  IN VARCHAR2,  
    P_HIRE_DATE     IN DATE,  
    P_JOB_ID        IN VARCHAR2,  
    P_SALARY        IN NUMBER,  
    P_MANAGER_ID    IN NUMBER,  
    P_DEPARTMENT_ID IN NUMBER,  
    ID_EMP          OUT NUMBER  
)  
IS  
    V_DEPARTMENT_EXISTS NUMBER;  
    V_MANAGER_EXISTS   NUMBER;  
    V_JOB_EXISTS       NUMBER;  
    V_NEW_EMPLOYEE_ID  NUMBER;  
  
BEGIN  
    SELECT COALESCE(MAX(EMPLOYEE_ID), 0) + 1 INTO V_NEW_EMPLOYEE_ID FROM EMPLOYEES;
```

Hoja de Trabajo		Generador de Consultas	
		<pre>SELECT COALESCE(MAX(EMPLOYEE_ID), 0) + 1 INTO V_NEW_EMPLOYEE_ID FROM EMPLOYEES;</pre>	
		<pre>-- VALIDAR LA EXISTENCIA DEL DEPARTAMENTO SELECT COUNT(*) INTO V_DEPARTMENT_EXISTS FROM DEPARTMENTS WHERE DEPARTMENT_ID = P_DEPARTMENT_ID;</pre>	
		<pre>IF V_DEPARTMENT_EXISTS = 0 THEN     DBMS_OUTPUT.PUT_LINE('ERROR: DEPARTMENT CON ID '    P_DEPARTMENT_ID    ' NO EXISTE.');</pre>	
		<pre>    ID_EMP := 0;     RETURN; END IF;</pre>	
		<pre>-- VALIDAR LA EXISTENCIA DEL GERENTE IF P_MANAGER_ID IS NOT NULL THEN     SELECT COUNT(*) INTO V_MANAGER_EXISTS     FROM EMPLOYEES     WHERE EMPLOYEE_ID = P_MANAGER_ID;</pre>	
		<pre>IF V_MANAGER_EXISTS = 0 THEN     DBMS_OUTPUT.PUT_LINE('ERROR: MANAGER CON ID '    P_MANAGER_ID    ' NO EXISTE.');</pre>	
		<pre>    ID_EMP := 0;</pre>	

Hoja de Trabajo		Generador de Consultas	
		<pre>    ID_EMP := 0;     RETURN;</pre>	
		<pre>END IF;</pre>	
		<pre>END IF;</pre>	
		<pre>-- VALIDAR LA EXISTENCIA DEL TRABAJO SELECT COUNT(*) INTO V_JOB_EXISTS FROM JOBS WHERE JOB_ID = P_JOB_ID;</pre>	
		<pre>IF V_JOB_EXISTS = 0 THEN     DBMS_OUTPUT.PUT_LINE('ERROR: JOB CON ID '    P_JOB_ID    ' NO EXISTE.');</pre>	
		<pre>    ID_EMP := 0;     RETURN; END IF;</pre>	
		<pre>-- INSERTAR EL NUEVO EMPLEADO INSERT INTO EMPLOYEES (     EMPLOYEE_ID,     FIRST_NAME,     LAST_NAME,     EMAIL,</pre>	

```
Hoja de Trabajo | Generador de Consultas
EMAIL,|
PHONE_NUMBER,
HIRE_DATE,
JOB_ID,
SALARY,
MANAGER_ID,
DEPARTMENT_ID
) VALUES (
V_NEW_EMPLOYEE_ID,
P_FIRST_NAME,
P_LAST_NAME,
P_EMAIL,
P_PHONE_NUMBER,
P_HIRE_DATE,
P_JOB_ID,
P_SALARY,
P_MANAGER_ID,
P_DEPARTMENT_ID
);

COMMIT;
DBMS_OUTPUT.PUT_LINE('INSERCIÓN REALIZADA CORRECTAMENTE!');
```

```
Hoja de Trabajo | Generador de Consultas
DBMS_OUTPUT.PUT_LINE('INSERCIÓN REALIZADA CORRECTAMENTE!');
ID_EMP := V_NEW_EMPLOYEE_ID;

EXCEPTION
WHEN OTHERS THEN
ROLLBACK;
DBMS_OUTPUT.PUT_LINE('NO SE PUDO INSERTAR: ' || SQLERRM);
ID_EMP := 0;

END;
```

LLAMADA

```

SET SERVEROUTPUT ON
DECLARE
  ID_EMP NUMBER ;
BEGIN
  INSETAR_EMPLEADO(
    'NOE',
    'CERON',
    'NOE.CERON@GMAIL.COM',
    '503-7777-0000',
    SYSDATE,
    'SA_REP',
    50000,
    100,
    50,
    ID_EMP
  );
  IF ID_EMP <> 0 THEN
    DBMS_OUTPUT.PUT_LINE('EL NUEVO EMPLEADO TIENE EL SIGUIENTE ID:' || ID_EMP);
  END IF ;
END;

```

SALIDA (HICE PRUEBAS CON VALORES QUE IBAN A INTENTAR VIOLAR LA INTEGRIDAD DE LAS RELACIONES Y SE GENERARON LAS SIGUIENTES SALIDAS)

Tarea terminada en 0.067 segundos

ERROR: MANAGER CON ID 999 NO EXISTE.

Procedimiento PL/SQL terminado correctamente.

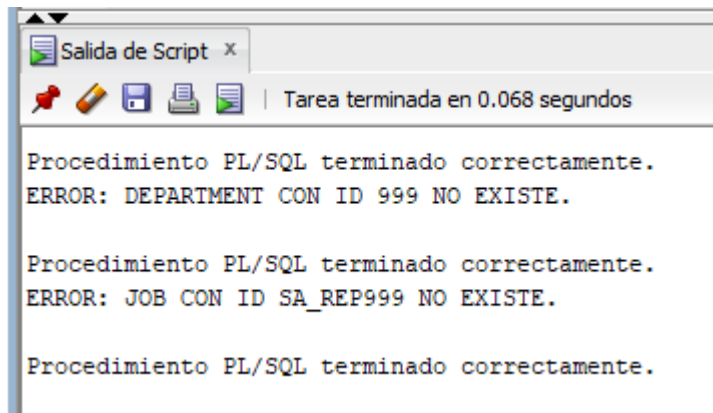
Tarea terminada en 0.066 segundos

ERROR: MANAGER CON ID 999 NO EXISTE.

Procedimiento PL/SQL terminado correctamente.

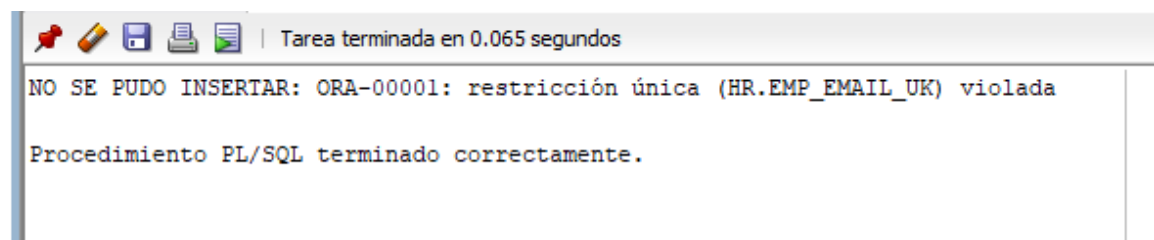
ERROR: DEPARTMENT CON ID 999 NO EXISTE.

Procedimiento PL/SQL terminado correctamente.



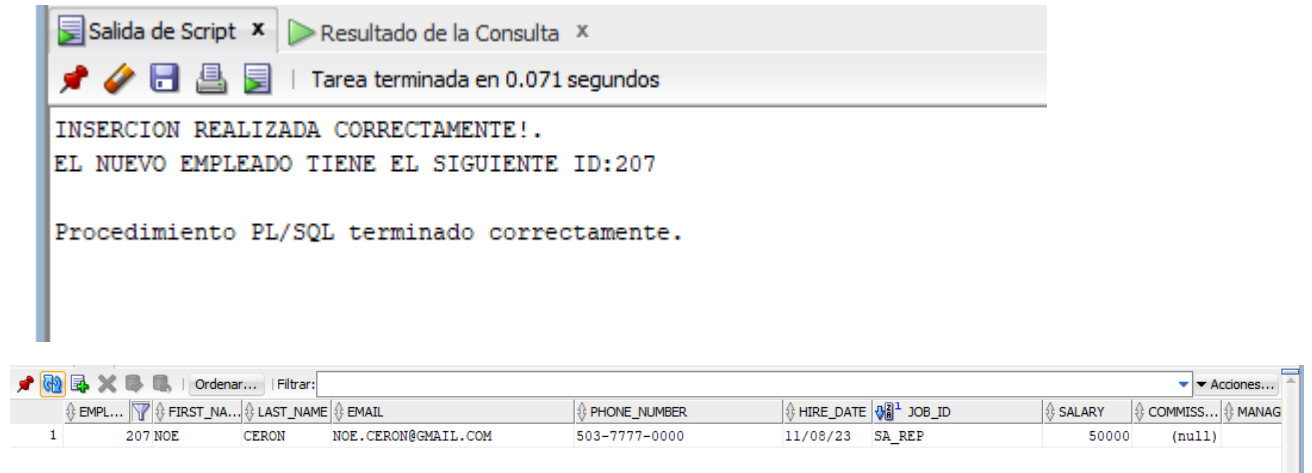
```
Procedimiento PL/SQL terminado correctamente.  
ERROR: DEPARTMENT CON ID 999 NO EXISTE.  
  
Procedimiento PL/SQL terminado correctamente.  
ERROR: JOB CON ID SA_REP999 NO EXISTE.  
  
Procedimiento PL/SQL terminado correctamente.
```

(SE INTENTO AGREGAR EL MISMO EMAIL)



```
NO SE PUDO INSERTAR: ORA-00001: restricción única (HR.EMP_EMAIL_UK) violada  
  
Procedimiento PL/SQL terminado correctamente.
```

USO LOS DATOS ADECUADOS



```
INSERCIÓN REALIZADA CORRECTAMENTE!.  
EL NUEVO EMPLEADO TIENE EL SIGUIENTE ID:207  
  
Procedimiento PL/SQL terminado correctamente.
```

EMPL...	FIRST_NA...	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISS...	MANAG
1	207	NOE	CERON	NOE.CERON@GMAIL.COM	503-7777-0000	11/08/23	SA_REP	50000	(null)