

Bases de Datos 1

Laboratorio 7

- Fabián Bustos
- Ian Murillo

Evidencia Laboratorio 7

Ejercicio 1:

Investigue los parámetros del procedimiento `dbms_scheduler.create_job`, lístelos e indique qué significa cada uno.

1. `Job_name`: Identificador del JOB.
2. `Job_Type`: Tipo del JOB generado (PLSQL_BLOCK, STORED PROCEDURE, EXECUTABLE).
3. `Job_action`: especifica la acción a realizar.
4. `number_of_arguments` IN PLS_INTEGER AUTO: numero de argumentos a recibir para realizar la acción.
5. `start_date` IN TIMESTAMP WITH TIME ZONE DEFAULT NULL: Fecha en la cual se ejecutará el JOB.
6. `end_date` IN TIMESTAMP WITH TIME ZONE DEFAULT NULL: Fecha en la cual terminará el JOB.
7. `job_priority` IN NUMBER: Prioridad del JOB (1 alta, 5 baja) 3 por defecto.
8. `enabled` IN BOOLEAN DEFAULT FALSE: Especifica si el trabajo se habilita o no al ser creado.
9. `auto_drop` IN BOOLEAN DEFAULT TRUE: Si es TRUE el trabajo será borrado después de ser ejecutado o ser deshabilitado.

Ejercicio 2:

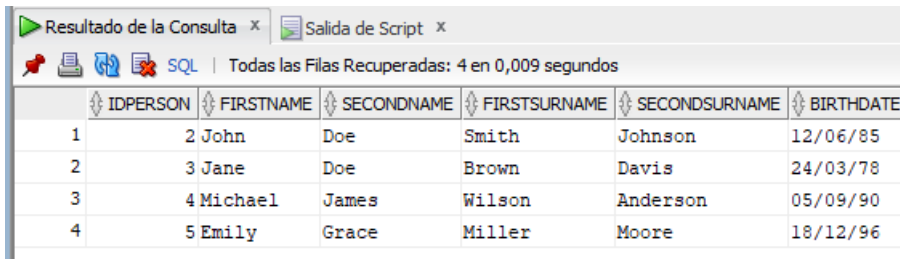
Cree un paquete que contenga un procedimiento que registre empleados en la tabla empleado.

```
14 CREATE OR REPLACE PACKAGE BODY createPerson
15 AS
16 Procedure insertPerson
17 (pcFirstName IN VARCHAR2, pcSecondName IN VARCHAR2, pcFirstSurname IN VARCHAR2, pcSecondSurname IN VARCHAR2, pcBirthDate IN DATE)
18 AS
19 BEGIN
20     INSERT INTO person(idPerson, firstName, secondName, firstSurname, secondSurname, birthDate)
21     VALUES (person_seq.NEXTVAL, pcFirstName, pcSecondName, pcFirstSurname, pcSecondSurname, pcBirthDate);
22     COMMIT;
23 END;
24 END createPerson;
```

Se prueba su funcionamiento con el siguiente script, cambiando los parámetros de entrada a elección:

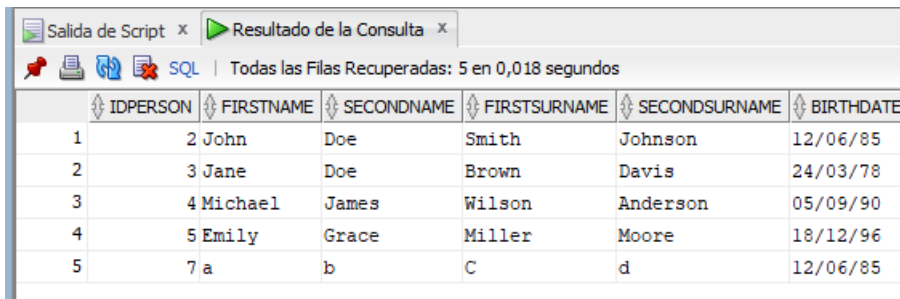
```
26 BEGIN
27 createPerson.insertPerson('a','b','C','d',TO_DATE('1985-06-12','YYYY-MM-DD'));
28
29 END;
```

La tabla comienza con estos datos:



IDPERSON	FIRSTNAME	SECONDA	FIRSTSURNAME	SECONDSURNAME	BIRTHDATE
1	2 John	Doe	Smith	Johnson	12/06/85
2	3 Jane	Doe	Brown	Davis	24/03/78
3	4 Michael	James	Wilson	Anderson	05/09/90
4	5 Emily	Grace	Miller	Moore	18/12/96

Y se incluye el nuevo dato llamado desde el package:



IDPERSON	FIRSTNAME	SECONDA	FIRSTSURNAME	SECONDSURNAME	BIRTHDATE
1	2 John	Doe	Smith	Johnson	12/06/85
2	3 Jane	Doe	Brown	Davis	24/03/78
3	4 Michael	James	Wilson	Anderson	05/09/90
4	5 Emily	Grace	Miller	Moore	18/12/96
5	7 a	b	C	d	12/06/85

Ejercicio 3:

Cree un test para invocar al procedimiento como un job y ejecútelo.

a. ¿Qué significa “A01”?

Es el nombre que tendrá el job creado

b. ¿Se puede cambiar “A01”?

Sí, después de creado el job se puede cambiar el “A01”

c. ¿Qué implicación tendría el cambio del punto b. en el código del procedimiento a ejecutar en el job?

No cambiaría el nombre de un Job previamente creado como A01, crearía un Job nuevo con otro nombre que haría el mismo procedimiento que el job anterior

d. ¿Qué significa freq=secondly?

Significa que el procedimiento del job se realizará cada segundo dentro del rango de tiempo que se le dio.

e. ¿Cómo se le envían parámetros a un procedimiento que se ejecuta desde un job?

Se utiliza la función `reset_job_argument_value` del `DBMS_Scheduler`.

f. ¿Qué implicación tiene que end_date esté en NULL?

Implica que el JOB se ejecutará indefinidamente hasta que se desactive o elimine manualmente.

g. ¿Qué significa SYSTIMESTAMP?

Retorna la fecha del sistema, incluyendo segundos fraccionales y zona horaria del sistema donde se aloja la base de datos.

Ejercicio 4:

Realice la siguiente consulta y valide que su job se está ejecutando. Ajuste el select a los valores que usted tiene (owner, job_name). Evidencie con una imagen.

Ejercicio 5:

Programe un sql para demostrar que se está incrementando la cantidad de empleados en la tabla y demuéstrelo con una captura de pantalla

Ejercicio 6:

Investigue cómo matar un job. Ejecute la sentencia y agregue una imagen con la evidencia.