

# Unidad 5

SQL Avanzado

Parte 2b

**UTPL**

*Material basado en instructivos Oracle*

# Agenda

Procedimientos almacenados y triggers



# Esquema de ejemplo “Libros2020”

```
libros (idlibro, titulo, numpaginas, anio, nroedicion)
autores (idautor, nombreautor, numlibros)
autorias (idlibro, idautor, orden)
```

- Restricciones de integridad adicionales
  - Un mismo libro no puede tener dos autores en el mismo orden
  - autores.numlibros no puede ser negativo
  - El numpaginas de libros debe ser entre 50 y 2000
  - El nroedicion de libros debe ser de 1 en adelante
  - El año del libro no puede ser inferior a 1800
  - numlibros de autores es un campo calculado. Corresponde al total de autorías de cada autor

# **ANONYMOUS PL/SQL BLOCKS**



# Block Structure for Anonymous PL/SQL Blocks

**DECLARE**      (optional)

    Declare PL/SQL objects to be used  
    within this block

**BEGIN**      (mandatory)

    Define the executable statements

**EXCEPTION** (optional)

    Define the actions that take place if  
    an error or exception arises

**END;**      (mandatory)



# Ejemplo 1

## Registro de un libro en la base de datos de biblioteca

```
DECLARE
    l_bfile BFILE;
    l_blob BLOB;
BEGIN
    INSERT INTO libros VALUES (1, '9788490908746', 'Java 8',
                               'Ejercicios prácticos Java', 340,
                               'ES', 8, 2017, 1, 5, EMPTY_BLOB())
    RETURN imgPortada INTO l_blob;

    l_bfile := BFILENAME('IMAGES', '9788490908746.jpeg');
    DBMS_LOB.fileopen(l_bfile, Dbms_Lob.File_ Readonly);
    DBMS_LOB.loadfromfile(l_blob,l_bfile,DBMS_LOB.getlength(l_bfile));
    DBMS_LOB.fileclose(l_bfile);

    INSERT INTO ejemplares VALUES (1, 1, 'P1A', 'D');
    INSERT INTO ejemplares VALUES (2, 1, 'P1A', 'D');

    INSERT INTO autores (idAutor, nombreAutor, pais, numLibros)
    VALUES (1, 'Luis Joyanes', 'España', 1);
    INSERT INTO autores (idAutor, nombreAutor, pais, numLibros)
    VALUES (2, 'Carlos Coronel', 'México', 1);

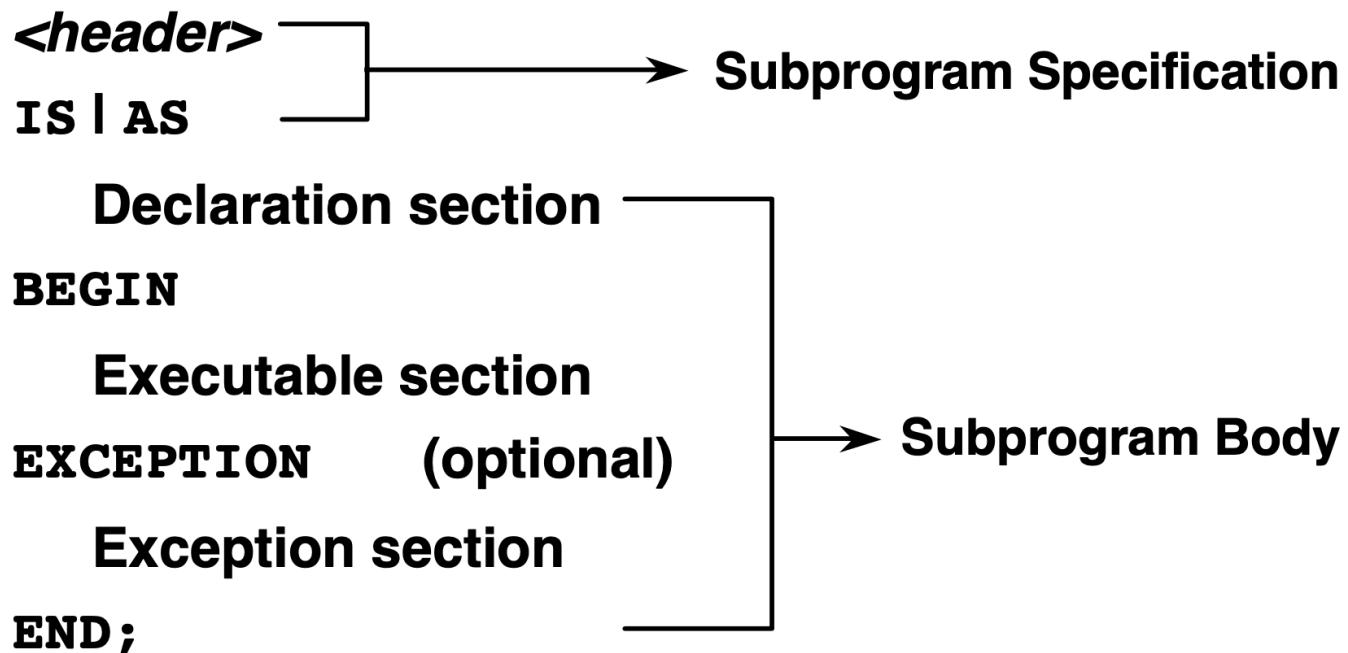
    INSERT INTO autorias (idLibro, idAutor, orden) VALUES (1, 1, 1);
    INSERT INTO autorias (idLibro, idAutor, orden) VALUES (1, 2, 2);
    COMMIT;

EXCEPTION WHEN OTHERS THEN
    ROLLBACK;
END;
/
```

# **STORED PROCEDURES PL/SQL**



# Block Structure for PL/SQL Subprograms



# Syntax for Creating Procedures

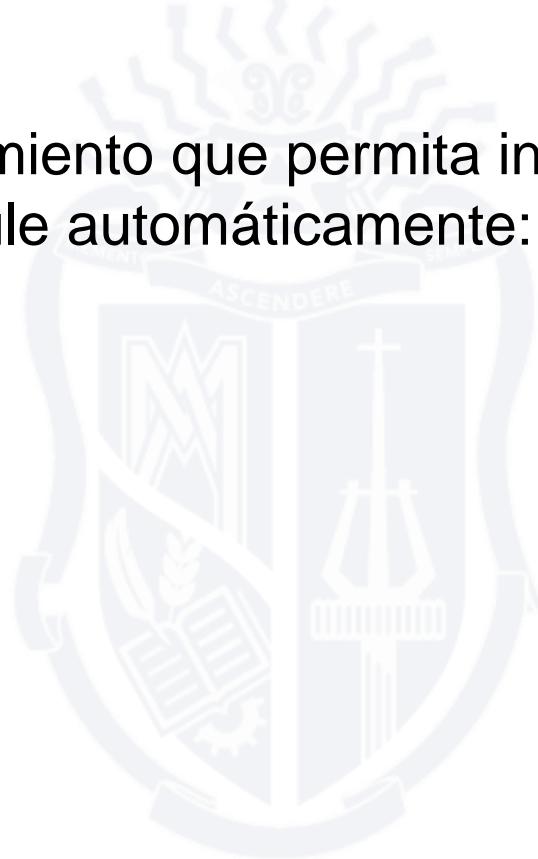
```
CREATE [OR REPLACE] PROCEDURE procedure_name
[ (parameter1 [mode1] datatype1,
  parameter2 [mode2] datatype2,
  . . .)]
IS|AS
PL/SQL Block;
```

- The REPLACE option indicates that if the procedure exists, it will be dropped and replaced with the new version created by the statement.
- PL/SQL block starts with either BEGIN or the declaration of local variables and ends with either END or END *procedure\_name*.



## Ejemplo 2

Construir un procedimiento que permita insertar **autorías** tal que al **orden** lo calcule automáticamente:



# **STORED FUNCTIONS PL/SQL**



# Syntax for Creating Functions

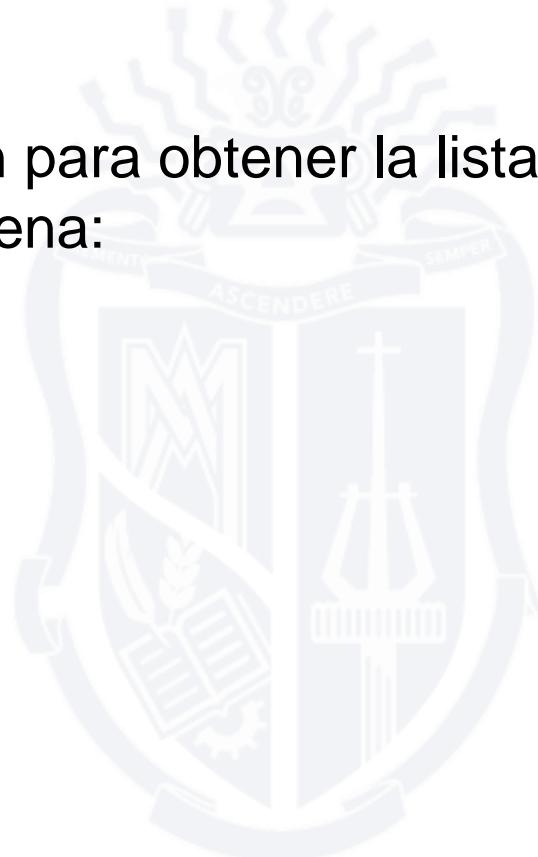
```
CREATE [OR REPLACE] FUNCTION function_name
  [(parameter1 [mode1] datatype1,
    parameter2 [mode2] datatype2,
    . . .)]
RETURN datatype
IS|AS
PL/SQL Block;
```

- The PL/SQL block must have at least one RETURN statement.



## Ejemplo 3

Construir una función para obtener la lista de autores de un libro en una sola cadena:





# TRIGGERS PL/SQL



# Types of Triggers

A trigger:

- Is a PL/SQL block or a PL/SQL procedure associated with a table, view, schema, or the database
- Executes implicitly whenever a particular event takes place
- Can be either:
  - Application trigger: Fires whenever an event occurs with a particular application
  - Database trigger: Fires whenever a data event (such as DML) or system event (such as logon or shutdown) occurs on a schema or database

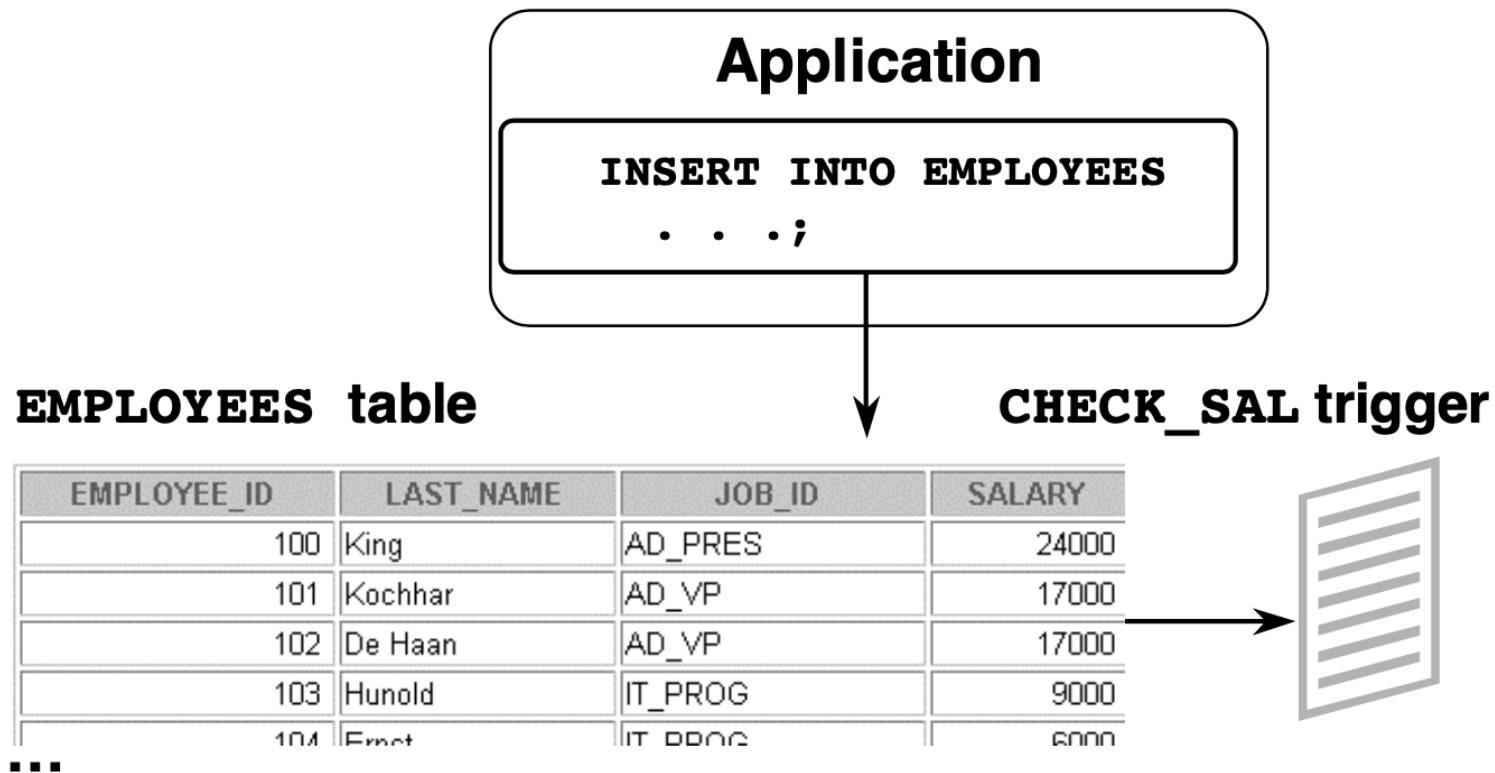


# Guidelines for Designing Triggers

- Design triggers to:
  - Perform related actions
  - Centralize global operations
- Do not design triggers:
  - Where functionality is already built into the Oracle server
  - That duplicate other triggers
- Create stored procedures and invoke them in a trigger, if the PL/SQL code is very lengthy.
- The excessive use of triggers can result in complex interdependencies, which may be difficult to maintain in large applications.



# Database Trigger: Example



# Creating DML Triggers

A triggering statement contains:

- Trigger timing
  - For table: BEFORE, AFTER
  - For view: INSTEAD OF
- Triggering event: INSERT, UPDATE, or DELETE
- Table name: On table, view
- Trigger type: Row or statement
- WHEN clause: Restricting condition
- Trigger body: PL/SQL block



# Syntax for Creating DML Statement Triggers

```
CREATE [OR REPLACE] TRIGGER trigger_name
    timing
        event1 [OR event2 OR event3]
        ON table_name
trigger_body
```

- Note: Trigger names must be unique with respect to other triggers in the same schema.



# Creating DML Statement Triggers

Example:

```
CREATE OR REPLACE TRIGGER secure_emp
BEFORE INSERT ON employees
BEGIN
    IF (TO_CHAR(SYSDATE, 'DY') IN ('SAT', 'SUN')) OR
        (TO_CHAR(SYSDATE, 'HH24:MI')
            NOT BETWEEN '08:00' AND '18:00')
    THEN RAISE_APPLICATION_ERROR (-20500, 'You may
                                    insert into EMPLOYEES table only
                                    during business hours.');
    END IF;
END;
/
```



# Syntax for Creating a DML Row Trigger

```
CREATE [OR REPLACE] TRIGGER trigger_name
  timing
    event1 [OR event2 OR event3]
    ON table_name
    [REFERENCING OLD AS old / NEW AS new]
FOR EACH ROW
  [WHEN (condition)]
trigger_body
```

# Creating DML Row Triggers

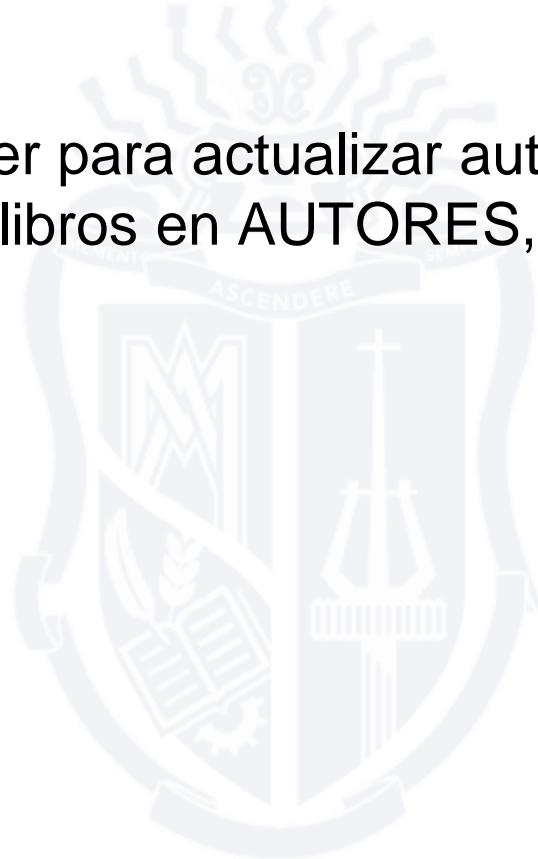
Example:

```
CREATE OR REPLACE TRIGGER restrict_salary
    BEFORE INSERT OR UPDATE OF salary ON employees
    FOR EACH ROW
    BEGIN
        IF NOT (:NEW.job_id IN ('AD_PRES', 'AD_VP'))
            AND :NEW.salary > 15000
        THEN
            RAISE_APPLICATION_ERROR (-20202, 'Employee
                cannot earn this amount');
        END IF;
    END;
/
```

Entre 20000  
y 20999

# Ejemplo 4

Implementar un trigger para actualizar automáticamente el campo derivado numlibros en AUTORES, cuando se agregue una autoría:



# Ejemplo 5

Suponga que existe una restricción que dice que un libro no puede tener más de 3 autores. Implementar un trigger que lo controle:

