

AED. UT2.Manejo de conectores.

Moisés Antonio Pestano Castro

1. Creación de las tablas (tanto en PostgreSQL como en Oracle)

```
CREATE TABLE Proveedores (  
    Cod_prov varchar(4) PRIMARY KEY,  
    Nombre_prov varchar(30),  
    Direccion varchar(30),  
    Telefono varchar(9),  
    Bonifica DOUBLE PRECISION  
);
```

```
CREATE TABLE Productos (  
    Cod_prod varchar(5) PRIMARY KEY,  
    Nombre_prod varchar(30),  
    Precio DOUBLE PRECISION,  
    Stock int,  
    Cod_prov varchar(4),  
    FOREIGN KEY (Cod_prov) REFERENCES Proveedores(Cod_prov)  
);
```

```
CREATE TABLE pedidos_pendientes (  
    Cod_prov varchar(4),  
    Stock int,  
    fecha_modificacion DATE,  
    FOREIGN KEY (Cod_prov) REFERENCES Proveedores(Cod_prov)  
);
```

2. Creación de procedimiento almacenado

POSTGRESQL

```
CREATE OR REPLACE PROCEDURE ejercicio4(  
    IN codigoProveedor varchar(4),  
    IN porcentaje int  
)  
LANGUAGE plpgsql  
AS $$  
BEGIN  
    UPDATE Proveedores  
    SET Bonifica = Bonifica * (1.0 + (porcentaje::float / 100))  
    WHERE Cod_prov = codigoProveedor;  
END;  
$$  
;
```

ORACLE

```
CREATE OR REPLACE PROCEDURE "EJERCICIO4" (  
    codigoproveedor_ IN VARCHAR2,  
    porcentaje_ IN NUMBER  
) IS  
BEGIN  
    UPDATE Proveedores  
    SET Bonifica = Bonifica * (1.0 + (porcentaje_ / 100))  
    WHERE Cod_prov = codigoproveedor_;  
END "EJERCICIO4";
```

3. Creación de procedimiento almacenado

POSTGRESQL

```
CREATE OR REPLACE FUNCTION funcion_ejercicio5()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.Stock < 10 THEN
        INSERT INTO pedidos_pendientes (Cod_prov, Stock, fecha_modificacion)
        VALUES (NEW.Cod_prov, NEW.Stock, CURRENT_DATE);
    END IF;
    RETURN NEW;
END;
$$
LANGUAGE plpgsql;

CREATE TRIGGER ejercicio5
AFTER UPDATE ON Productos
FOR EACH ROW
EXECUTE FUNCTION funcion_ejercicio5();
```

ORACLE

```
CREATE OR REPLACE TRIGGER ejercicio5
AFTER UPDATE ON Productos
FOR EACH ROW
BEGIN
    IF :NEW.Stock < 10 THEN
        INSERT INTO pedidos_pendientes (Cod_prov, Stock, fecha_modificacion)
        VALUES (:NEW.Cod_prov, :NEW.Stock, SYSDATE);
    END IF;
END;
/
```