



# Estructuras de Condición en Bloques PL/SQL

**MDY3131**

# Experiencia de Aprendizaje y Competencia Asociada



ESCUELA DE  
INFORMÁTICA Y  
TELECOMUNICACIONES

Experiencia	Nombre	Unidad de Competencia Especialidad – Nivel de la Competencia de Empleabilidad
Nº 1	Construyendo Bloques Anónimos PL/SQL Simples	Desarrolla operaciones sobre la base de datos que permitan administrar los objetos de la misma de acuerdo a requerimientos de usuario y buenas prácticas de la industria.
		Resolución de Problemas (N1)

- Identificar los usos y tipos de Estructuras de Control en PL/SQL.
- Cómo construir Estructuras de Selección o Condicionales con la sentencia IF Simple.
- Cómo construir Estructuras de Selección o Condicionales con la sentencia IF-THEN-ELSE.
- Cómo construir Estructuras de Selección o Condicionales con la sentencia IF-THEN-ELSIF.
- Cómo utilizar sentencias CASE y expresiones CASE.



## Estructuras de Selección

1. Los bloques PL/SQL de los ejemplos usan tablas del esquema HR de la Base de Datos Oracle y algunas tablas nuevas. Por esta razón, en las sentencias DML (INSERT, UPDATE y DELETE) de los bloques se usan copias de esas tablas del esquema HR para no realizar los cambios de datos en las tablas originales.
2. De acuerdo con esto, antes de que Ud. ejecute los ejemplos debe crear las tablas necesarias con el archivo [script\\_crea\\_tablas\\_ejemplos.sql](#)

# Control de Flujo de Ejecución

IF...  
THEN...  
END IF;

IF...  
THEN...  
ELSE...  
END IF;

IF...  
THEN...  
ELSIF...  
THEN...  
END IF;

IF...  
THEN...  
ELSIF...  
THEN...  
ELSE...  
END IF;

CASE  
WHEN... THEN..  
WHEN... THEN..  
WHEN... THEN..  
ELSE  
END CASE;

loop

for

while

# Estructuras de Selección: Sentencia IF Simple

Permite realizar acciones de forma selectiva basado en la evaluación de condiciones.

- Sintaxis

```
IF condición THEN  
  sentencias;  
[ELSIF condición THEN  
  sentencias;  
[ELSE  
  sentencias;  
END IF;
```



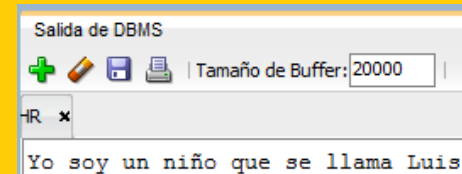
# Estructuras de Selección: Sentencia IF Simple

- Ejemplo:

```
DECLARE
  v_mi_edad number(2) := 31;
BEGIN
  IF v_mi_edad < 11 THEN
    DBMS_OUTPUT.PUT_LINE('Yo soy un niño');
  END IF;
END;
```

- Ejemplo:

```
DECLARE
  v_mi_edad    number(2) := 10;
  v_mi_nombre  varchar2(20) := 'Luis';
BEGIN
  IF v_mi_edad < 11 AND v_mi_nombre = 'Luis' THEN
    DBMS_OUTPUT.PUT_LINE('Yo soy un niño que se llama Luis');
  END IF;
END;
```



# Estructuras de Selección: Sentencia IF Simple

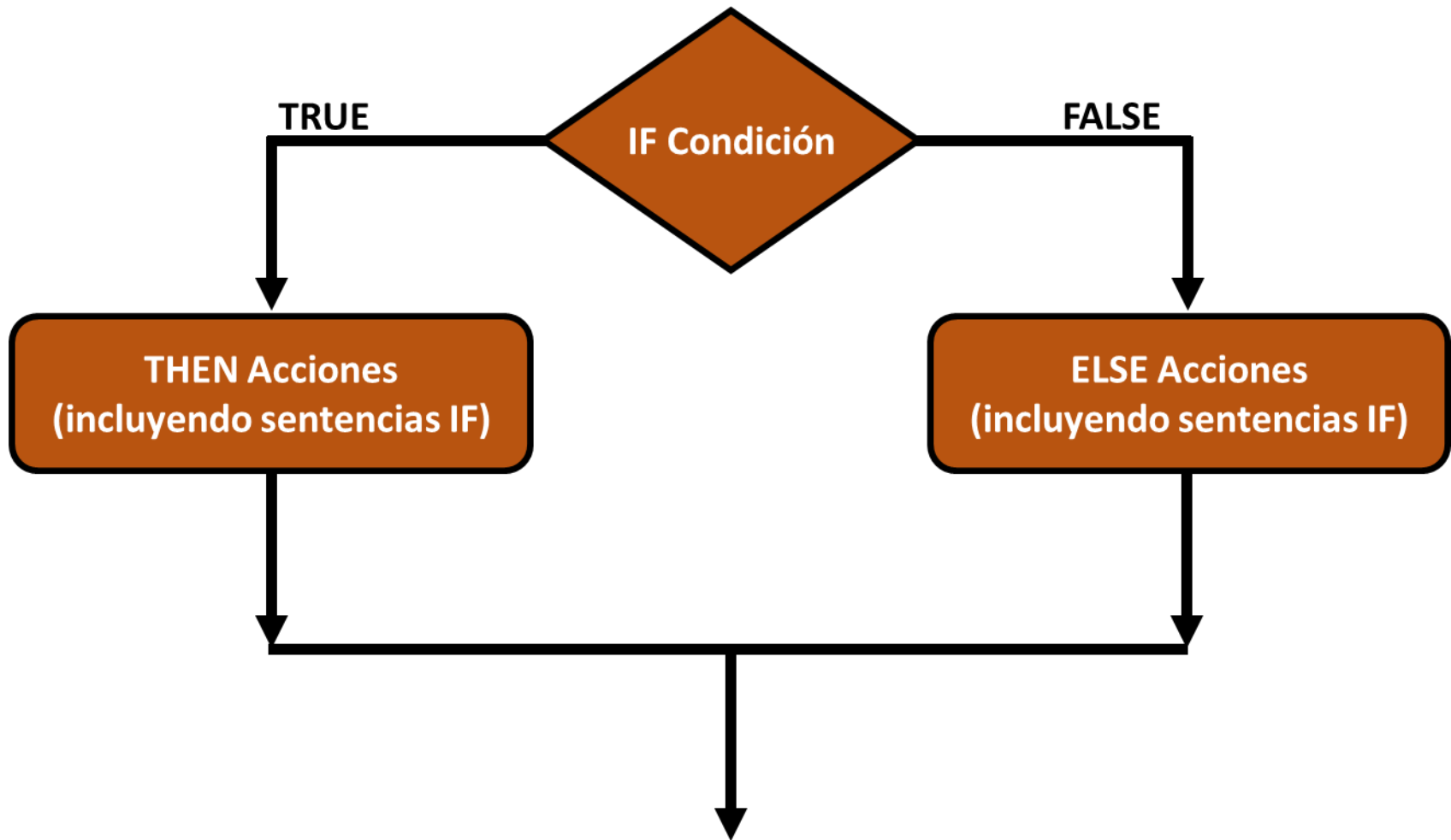
- Ejemplo:

```
VAR b_id_emp NUMBER
EXEC :b_id_emp:=150
DECLARE
v_salario NUMBER(8);
BEGIN
  SELECT salary
    INTO v_salario
  FROM empleados
  WHERE employee_id=:b_id_emp;
  IF v_salario < 5000 THEN
    UPDATE empleados
      SET salary=salary*1.30
    WHERE employee_id=:b_id_emp;
  END IF;
COMMIT;
END;
```

Salario del empleado 150 es de 10000



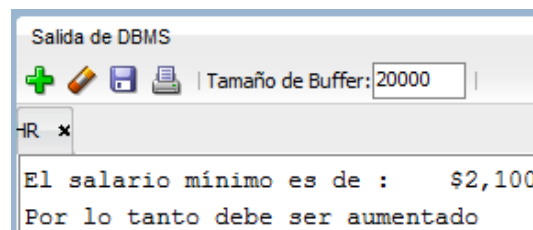
# Estructuras de Selección: Sentencia IF-THEN-ELSE



# Estructuras de Selección: Sentencia IF THEN ELSE

- Ejemplo:

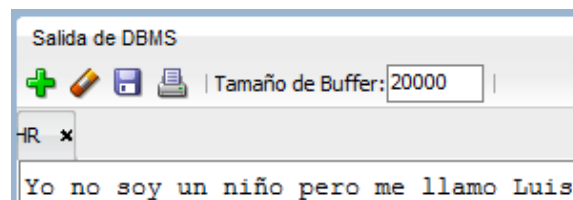
```
DECLARE
  v_salario_min  NUMBER(4);
BEGIN
  SELECT MIN(SALARY)
    INTO v_salario_min
    FROM employees;
  DBMS_OUTPUT.PUT_LINE('El salario mínimo es de : ' || TO_CHAR(v_salario_min,'$999,999'));
  IF v_salario_min > 3000 THEN
    DBMS_OUTPUT.PUT_LINE('Por lo tanto está dentro del rango normal');
  ELSE
    DBMS_OUTPUT.PUT_LINE('Por lo tanto debe ser aumentado');
  END IF;
END;
```



# Estructuras de Selección: Sentencia IF THEN ELSE

- Ejemplo:

```
DECLARE
  v_mi_edad number(2) := 30;
  v_mi_nombre varchar2(20) := 'Luis';
BEGIN
  IF v_mi_edad < 11 AND v_mi_nombre = 'Luis' THEN
    DBMS_OUTPUT.PUT_LINE('Yo soy un niño que se llama Luis');
  ELSE
    DBMS_OUTPUT.PUT_LINE('Yo no soy un niño pero me llamo Luis');
  END IF;
END;
```



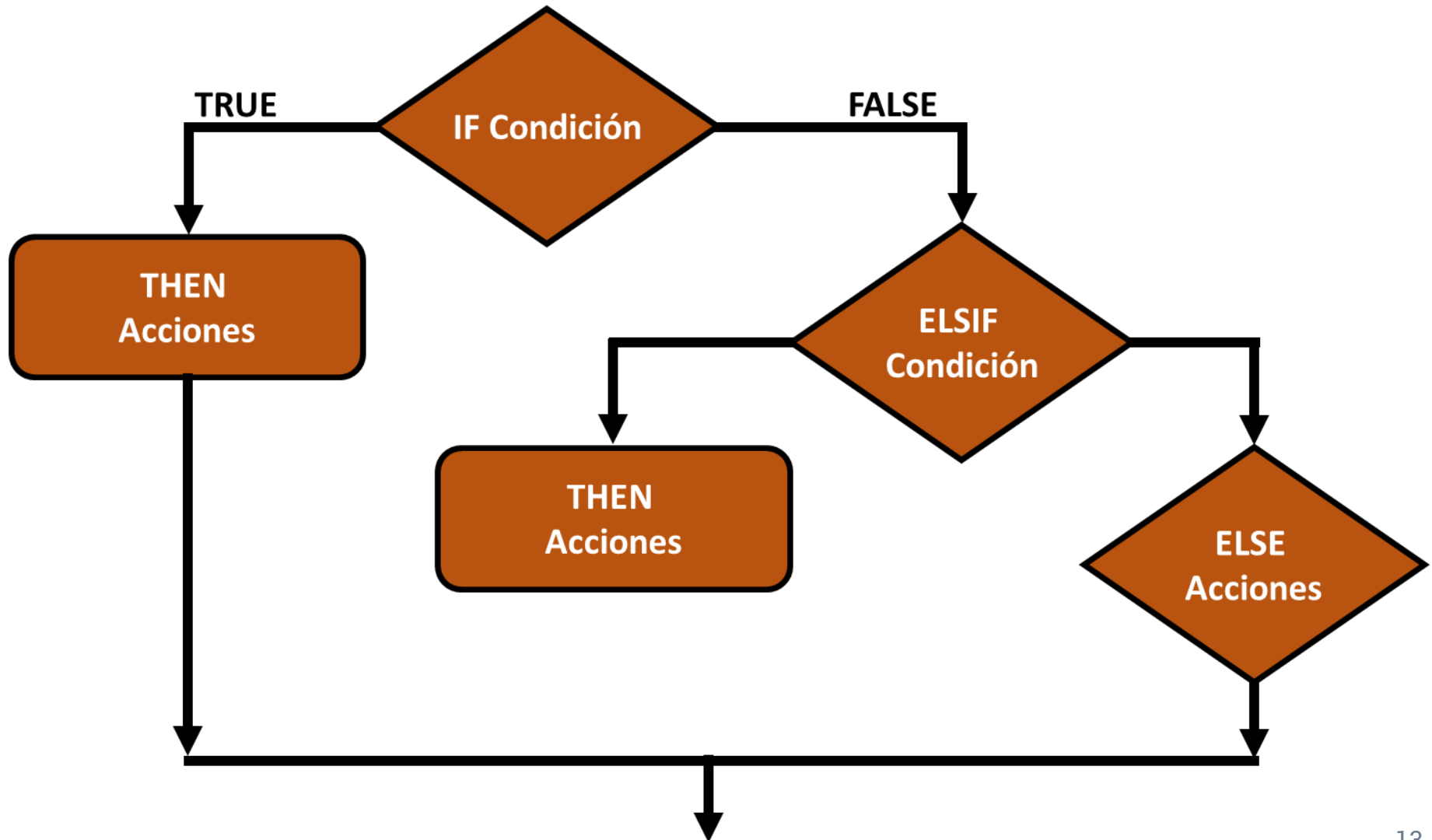
# Estructuras de Selección: Sentencia IF THEN ELSE

- Ejemplo:

```
VAR b_id_emp NUMBER
EXEC :b_id_emp:=150
DECLARE
v_salario NUMBER(8);
BEGIN
  SELECT salary
  INTO v_salario
  FROM employees
  WHERE employee_id=:b_id_emp;
  IF v_salario < 5000 THEN
    UPDATE empleados
    SET salary=salary*1.30
    WHERE employee_id=:b_id_emp;
  ELSE
    UPDATE empleados
    SET salary=salary*1.10
    WHERE employee_id=:b_id_emp;
  END IF;
  COMMIT;
END;
```

Salario del empleado 150 es de 10000

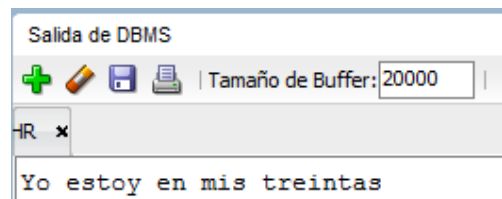
# Estructuras de Selección: Sentencia IF-THEN-ELSIF-ELSE



# Estructuras de Selección: Sentencia IF-THEN-ELSIF-ELSE

- Ejemplo:

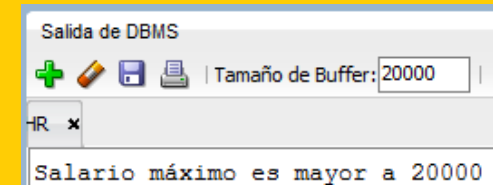
```
DECLARE
v_mi_edad number(2) :=31;
BEGIN
  IF v_mi_edad < 20 THEN
    DBMS_OUTPUT.PUT_LINE('Yo soy un niño');
  ELSIF v_mi_edad < 30 THEN
    DBMS_OUTPUT.PUT_LINE('Yo estoy en mis veintes');
  ELSIF v_mi_edad < 40 THEN
    DBMS_OUTPUT.PUT_LINE('Yo estoy en mis treintas');
  ELSE
    DBMS_OUTPUT.PUT_LINE('Yo siempre seré joven');
  END IF;
END;
```



# Estructuras de Selección: Sentencia IF-THEN-ELSIF-ELSE

- Ejemplo:

```
DECLARE
v_sal_max NUMBER(5);
BEGIN
  SELECT MAX(salary)
  INTO v_sal_max
  FROM employees;
  IF v_sal_max < 5000 THEN
    DBMS_OUTPUT.PUT_LINE('Salario máximo menor a 5000');
  ELSIF v_sal_max < 10000 THEN
    DBMS_OUTPUT.PUT_LINE('Salario máximo menor a 10000 y mayor = a 5000');
  ELSIF v_sal_max < 15000 THEN
    DBMS_OUTPUT.PUT_LINE('Salario máximo menor a 15000 y mayor = a 10000');
  ELSIF v_sal_max < 20000 THEN
    DBMS_OUTPUT.PUT_LINE('Salario máximo menor a 20000 y mayor = a 15000');
  ELSE
    DBMS_OUTPUT.PUT_LINE('Salario máximo es mayor a 20000');
  END IF;
END;
```





# Estructuras de Selección: Sentencia IF-THEN-ELSIF-ELSE

- Ejemplo:

```
VAR b_id_emp NUMBER
EXEC :b_id_emp:=100
DECLARE
v_salario NUMBER(8);
BEGIN
  SELECT salary
  INTO v_salario
  FROM employees
  WHERE employee_id=:b_id_emp;
  IF v_salario < 5000 THEN
    UPDATE empleados
    SET salary=salary*1.40
    WHERE employee_id=:b_id_emp;
  ELSIF v_salario BETWEEN 5000 AND 18000 THEN
    UPDATE empleados
    SET salary=salary*1.20
    WHERE employee_id=:b_id_emp;
  END IF;
  COMMIT;
END;
```

Salario del empleado 100 es de 24000

# Valores Nulos en Sentencias IF

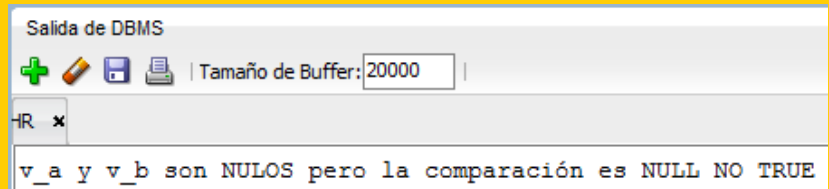
Las comparaciones  
simples que  
involucran nulos  
siempre retornan  
NULL

Para preguntar si  
una variable no  
posee un valor  
NULO se debe usar  
el operador lógico  
NOT (NOT NULL)

Para preguntar si  
una variable posee  
un valor NULO se  
debe usar IS NULL

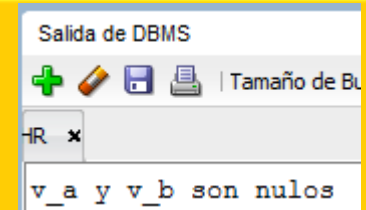
- Ejemplo:

```
DECLARE
v_a NUMBER:= NULL;
v_b NUMBER:= NULL;
BEGIN
  IF v_a = v_b THEN
    DBMS_OUTPUT.PUT_LINE('v_a es igual a v_b');
  ELSE
    DBMS_OUTPUT.PUT_LINE('v_a y v_b son NULOS pero la comparación es NULL NO TRUE');
  END IF;
END;
```



- Ejemplo:

```
DECLARE
v_a number:= NULL;
v_b number:=NULL;
BEGIN
  IF v_a IS NOT NULL AND v_b IS NOT NULL THEN
    DBMS_OUTPUT.PUT_LINE('v_a y v_b no son nulos');
  ELSIF
    v_a IS NULL AND v_b IS NULL THEN
    DBMS_OUTPUT.PUT_LINE('v_a y v_b son nulos');
  END IF;
END;
```





# CASE: Expresiones y Sentencias

La expresión  
condicional CASE  
permite evaluar una  
expresión y retornar  
un valor.  
Debe finalizar con END

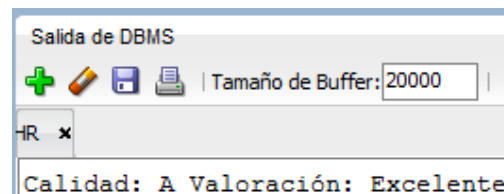
Para retornar el  
resultado, la expresión  
CASE utiliza un selector  
cuyo valor se utiliza  
para devolver una de  
varias alternativas

- Sintaxis:

```
CASE [ selector ]  
  WHEN expresión1 THEN resultado1  
  WHEN expresión2 THEN resultado2  
  ...  
  WHEN expresiónN THEN resultadoN  
  [ ELSE resultadoN+1 ]  
END;
```

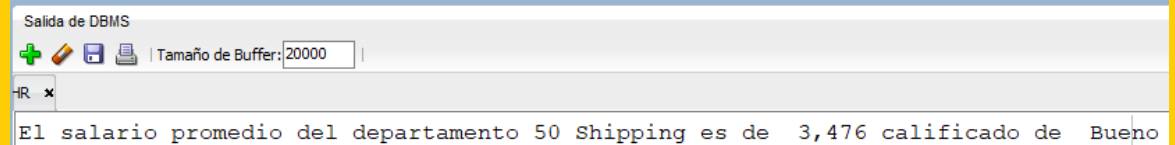
- Ejemplo:

```
DECLARE
v_calidad    VARCHAR2(1) := 'A';
v_valoracion VARCHAR2 (20);
BEGIN
  v_valoracion :=
    CASE v_calidad
      WHEN 'A' THEN ' Excelente'
      WHEN 'B' THEN ' Muy bueno'
      WHEN 'C' THEN ' Bueno'
      ELSE 'No existe calidad'
    END;
  DBMS_OUTPUT.PUT_LINE ('Calidad: ' || v_calidad || ' Valoración: ' || v_valoracion);
END;
```



- Ejemplo:

```
DECLARE
v_salario_prom NUMBER(5);
v_nombre_dep VARCHAR2(30);
v_valoracion VARCHAR2(15);
BEGIN
    SELECT ROUND(AVG(emp.salary)), dep.department_name INTO v_salario_prom, v_nombre_dep
    FROM employees emp JOIN departments dep
        ON(emp.department_id = dep.department_id)
    WHERE emp.department_id = 50
    GROUP BY dep.department_name ;
    v_valoracion :=
    CASE
        WHEN v_salario_prom > 5000 THEN 'Excelente'
        WHEN v_salario_prom > 3000 THEN 'Bueno'
        ELSE 'Regular'
    END;
    DBMS_OUTPUT.PUT_LINE ('El salario promedio del departamento 50 ' || v_nombre_dep || ' es de '
    || TO_CHAR(v_salario_prom, '9,999') || ' calificado de ' || v_valoracion);
END;
```





- Ejemplo:

```
VAR b_id_emp NUMBER
VAR b_id_emp NUMBER
EXEC :b_id_emp:=100
DECLARE
v_salario NUMBER(8);
v_porc_reajuste NUMBER(3,2);
BEGIN
  SELECT salary
    INTO v_salario
  FROM employees
  WHERE employee_id=:b_id_emp;
  v_porc_reajuste:=
CASE
  WHEN v_salario BETWEEN 0 AND 5000 THEN 0.40
  WHEN v_salario BETWEEN 5000 AND 12000 THEN 0.20
  WHEN v_salario BETWEEN 12001 AND 18000 THEN 0.10
  ELSE 0
END;
  UPDATE empleados
    SET salary=salary + ROUND((salary*v_porc_reajuste))
  WHERE employee_id=:b_id_emp;
END;
```

**Salario del empleado 100 es de 24000**

Se puede incluir  
sentencias utilizando  
CASE. Se evalúa la  
condición y realiza una  
acción.  
Finaliza con END CASE

La sentencia puede ser  
un bloque completo  
PL/SQL

- Sintaxis:

```
CASE [ selector ]  
  WHEN expresión1 THEN sentencia(s)1  
  WHEN expresión2 THEN sentencia(s)2  
  ...  
  WHEN expresiónN THEN sentencia(s)N  
  ELSE sentencia(s)  
END CASE;
```

- Ejemplo:

```
DECLARE
v_sal_prom  NUMBER(5);
BEGIN
  SELECT ROUND(AVG(salary))
  INTO v_sal_prom
  FROM employees;
  CASE v_sal_prom
    WHEN 15000 THEN
      UPDATE empleados
        SET salary = salary * 1.25
        WHERE salary < v_sal_prom;
    ELSE
      UPDATE empleados
        SET salary = salary * 1.05
        WHERE salary < v_sal_prom;
  END CASE;
END;
```

El salario promedio es de 6462

- Ejemplo:

```
DECLARE
v_sal_prom  NUMBER(5);
BEGIN
  SELECT ROUND(AVG(salary))
  INTO v_sal_prom
  FROM employees;
  CASE
    WHEN v_sal_prom < 5000 THEN
      UPDATE empleados
        SET salary = salary * 1.25
        WHERE salary < v_sal_prom;
    WHEN v_sal_prom < 7000 THEN
      UPDATE empleados
        SET salary = salary * 1.10
        WHERE salary < v_sal_prom;
    ELSE
      DBMS_OUTPUT.PUT_LINE('No corresponde aumento de salario');
  END CASE;
END;
```

El salario promedio es de 6462

- Ejemplo:

```
DECLARE
v_salario NUMBER(8);
BEGIN
  SELECT salary
  INTO v_salario
  FROM employees
  WHERE employee_id=100;
  CASE
    WHEN v_salario < 5000 THEN
      UPDATE empleados
        SET salary=salary*1.30
        WHERE employee_id=100;
    ELSE
      UPDATE empleados
        SET salary=salary*1.10
        WHERE employee_id=100;
  END CASE;
END;
```

Salario del empleado 100 es de 24000

- Se explicó los usos y tipos de Estructuras de Control en PL/SQL.
- Se explicó cómo construir Estructuras de Selección o Condicionales con la sentencia IF Simple.
- Se explicó cómo construir Estructuras de Selección o Condicionales con la sentencia IF-THEN-ELSE.
- Se explicó cómo construir Estructuras de Selección o Condicionales con la sentencia IF-THEN-ELSIF.
- Se explicó cómo utilizar sentencias CASE y expresiones CASE.