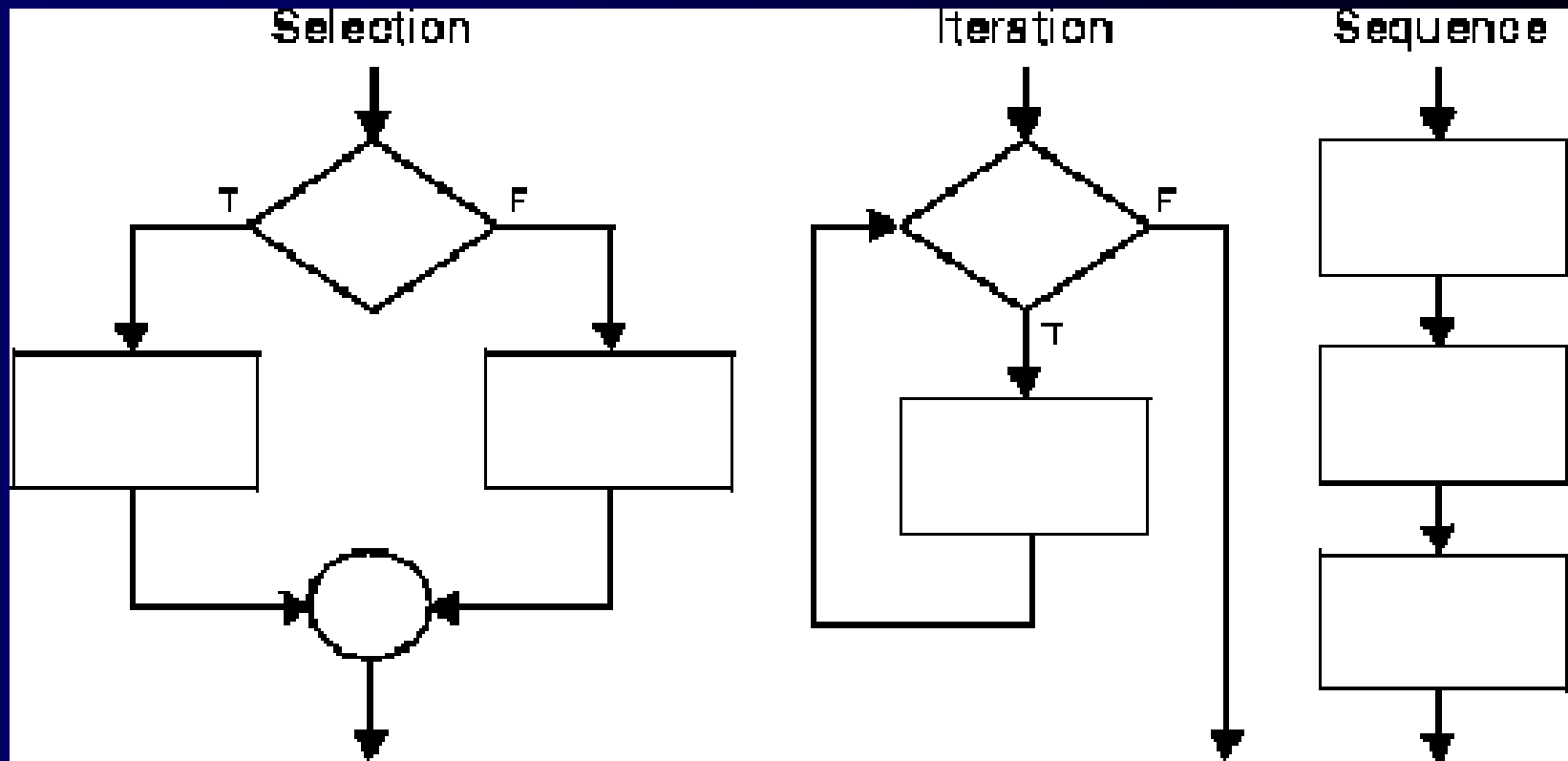


Control de Flujo en Bloques PL/SQL

Estructuras de Control de Flujo



Control Condicional: Sentencias IF

IF condición THEN

 secuencia de sentencias;

[ELSIF condición THEN

 secuencia de sentencias;]

[ELSE

 secuencia de sentencias;]

END IF;

Ejemplo

```
BEGIN
```

```
...
```

```
IF v_atenciones > 50 THEN
```

```
    v_gratificacion := 250;
```

```
ELSIF v_atenciones > 35 THEN
```

```
    v_gratificacion := 150;
```

```
ELSE
```

```
    v_gratificacion := 100;
```

```
END IF;
```

```
INSERT INTO nomina
```

```
    VALUES (v_id_emp, v_gratificacion, ...);
```

```
END;
```

Consejos

- El valor de una expresión booleana puede ser asignado directamente a una variable booleana

```
DECLARE
v_sobregirado BOOLEAN;
BEGIN
v_sobregirado := v_balance_nuevo < v_balance_minimo;
IF v_sobregirado THEN ...
...
END;
```

Consejos (cont.)

- Utilizar la cláusula ELSIF en lugar de sentencias IF anidadas.

```
IF condición1 THEN
    sentencia1;
ELSE
    IF condición2 THEN
        sentencia2;
    END IF;
END IF;
```

```
IF condición1 THEN
    sentencia1;
ELSIF condición2 THEN
    sentencia2;
END IF;
```

Construcción de Condiciones Lógicas

- Cualquier expresión que contenga un valor nulo devuelve NULL como resultado.

```
x := 5;
```

```
y := NULL;
```

```
...
```

```
IF x != y THEN -- da NULL, no TRUE
```

```
  secuencia_de_sentencias; -- no ejecutadas
```

```
END IF;
```

- Una comparación IS NULL devuelve como resultado TRUE o FALSE.

Control Iterativo: Sentencia LOOP

LOOP

 secuencia_de_sentencias;
END LOOP;

EXIT fuerza a que un ciclo termine incondicionalmente

EXIT - WHEN permite que un ciclo termine
 condicionalmente

Ejemplo

LOOP

...

IF v_valuacion_credito < 3 THEN

 EXIT; -- sale del ciclo inmediatamente

END IF;

END LOOP;

-- el control reasume aquí

Ejemplo

LOOP

...

EXIT WHEN v_valuacion_credito < 3;

...

END LOOP;

SELECT ...

IF contador > 100 THEN

EXIT;

END IF;

EXIT WHEN contador > 100;

Control Iterativo: Sentencia WHILE - LOOP

WHILE condición LOOP

 secuencia_de_sentencias;

END LOOP;

LOOP

 EXIT WHEN $x < y$;

 ...

END LOOP;

WHILE $x < y$ LOOP

 ...

END LOOP;

Ejemplos

```
    WHILE total <= 25000 LOOP
...
SELECT sal INTO salario
    FROM emp WHERE ...;
total := total + salario;
END LOOP;
```

LOOP UNTIL o REPEAT UNTIL en PL/SQL

```
    hecho := FALSE;
    WHILE NOT hecho LOOP
secuencia_de_sentencias;
hecho := expresión_booleana;
END LOOP;
```

Control Iterativo: Sentencia FOR - LOOP

```
FOR contador IN [REVERSE] li..lf LOOP  
    secuencia_de_sentencias;  
END LOOP;
```

Esquemas de Iteración:

j IN -5..5

k IN REVERSE primero..ultimo

paso IN 0..TRUNC(alto/bajo) * 2

codigo IN ASCII('A')..ASCII('J')

Ejemplos

```
FOR i IN 1..3 LOOP -- asigna los valores 1,2,3 a i
secuencia_de_sentencias; -- ejecuta tres veces
END LOOP;
```

```
SELECT COUNT(empno) INTO emp_contador
FROM emp;
FOR i IN 1..emp_contador LOOP
```

...

```
END LOOP;
```