

Bloque anónimo

```
[ DECLARE
    <declaraciones> ]
BEGIN
    <órdenes>
[ EXCEPTION
    <gestión de excepciones> ]
END;
```

Declaración de variables mediante atributos

Nombre_variable nombre_objeto%TYPE

Nombre_variable nombre_objeto%ROWTYPE

Excepciones

NO_DATA_FOUND

TOO_MANY_ROWS

Comandos para ver resultados de programas en pantalla

SET SERVEROUTPUT ON

DBMS_OUTPUT.PUT_LINE('texto');

Estructuras PLSQL

Alternativa simple	Alternativa doble	Alternativa múltiple
<pre>IF <condición> THEN Instrucciones; END IF;</pre>	<pre>IF <condición> THEN Instrucciones1; ELSE Instrucciones2; END IF;</pre>	<pre>IF <condición1> THEN Instrucciones1; ELSIF <condición2> THEN Instrucciones2; ELSIF <condición3> THEN Instrucciones3; ... [ELSE Instruccionesotras;] END IF;</pre>
Alternativa múltiple con CASE de comprobación		Alternativa múltiple con CASE de búsqueda
<pre>CASE expresión WHEN <valor1> THEN Instrucciones1; WHEN <valor2> THEN Instrucciones2; ... [ELSE Instruccionesotras;] END CASE;</pre>		<pre>CASE WHEN <condición1> THEN Instrucciones1; WHEN <condición2> THEN Instrucciones2; ... [ELSE Instruccionesotras;] END CASE;</pre>

BUCLES:

Iterar	
<pre>LOOP Instrucciones; EXIT [WHEN <condición>]; Instrucciones; END LOOP;</pre>	<p>Bucle que se repetirá indefinidamente hasta que encuentre EXIT sin condición o hasta que se cumpla la condición asociada a la clausula EXIT WHEN. Es una condición de salida.</p>
Mientras	
<pre>WHILE <condición> LOOP Instrucciones; END LOOP;</pre>	<p>Se evalúa la condición y, si se cumple, se ejecutarán las instrucciones del bucle. El bucle se seguirá ejecutando mientras se cumpla la condición. Es una condición de continuación.</p>
Para	
<pre>FOR <variablecontrol> IN <valorInicio> .. <valorFinal> LOOP Instrucciones; END LOOP;</pre>	