CIPFP Mislata
Centre Integrat Públic
Formació Professional Superior

Chuleta PL/SQL

Mostrar mensaje

```
DBMS_OUTPUT.PUT_LINE('Soy una cadena' || resultado);
```

Sentencias condicionales

```
IF condition_1 THEN
   statements_1
ELSIF condition_2 THEN
   statements_2
[ ELSIF condition_3 THEN
      statements_3
]
...
[  ELSE
      else_statements
]
END IF;
```

```
CASE selector
WHEN selector_value_1 THEN
      statements_1
WHEN selector_value_1 THEN
      statement_2
...
ELSE
      else_statements
END CASE;
```

Bucles

```
LOOP
      EXIT WHEN
condition;
END LOOP;
```

```
FOR var IN
lower_bound ..
upper_bound
LOOP
      statements;
END LOOP;
```

```
WHILE condition
LOOP
      statements;
END LOOP;
```

Excepciones

```
DECLARE
        _ - -
BEGIN
      -- executable section
      ...
      -- exception-handling section
      EXCEPTION
           WHEN e1 THEN
                -- exception_handler1
           WHEN NO_DATA_FOUND THEN
                -- exception_handler1
           WHEN TOO_MANY_ROWS THEN
                --  exception_handler1

           WHEN OTHERS THEN
                --  other_exception_handler
END;
```

Cursores

```
DECLARE
   CURSOR c_cursor IS
SELECT;
   VAR
TABLA.COLUMNA%TYPE;
   BEGIN
   OPEN c_cursor;
   LOOP
      FETCH CUR INTO
      VAR
    EXIT WHEN CUR
%NOTFOUND
   AQUI VA LA COSA
   END LOOP;
CLOSE c_cursor;
END;
```

```
FOR record IN cursor_name
LOOP
     process_record_statements;
END LOOP;
```

Procedimiento

```
CREATE [OR REPLACE ] PROCEDURE procedure_name (parameter_list in
tipo) IS
        [declaration variable tipo]
BEGIN
        [execution insert etc];
EXCEPTION
        [exception handler];
END;
```

Función

```
CREATE [OR REPLACE] FUNCTION function_name (parameter_list in
    tipo) RETURN return_type
IS
    [variable tipo]
BEGIN
    [executable section]
    Return variable;
[EXCEPTION]
    [exception-handling section]
END;
```

Trigger

```
CREATE [OR REPLACE] TRIGGER trigger_name
{BEFORE | AFTER } triggering_event ON table_name
[FOR EACH ROW]
DECLARE
    declaration variable tipo
BEGIN
    executable statements
    = :NEW.columna
EXCEPTION
    exception_handling statements
END;
```