

Pràctica : Bloc anònim en Oracle

La sentència de bloc anònim PL/SQL és una sentència executable que pot contindre sentències de control PL/SQL i sentències SQL

Per realitzar esta pràctica, utilitzarem la mv Windows 10 amb Oracle **SQL Developer**

En SQL Developer podem guardar un script amb **ctrl-s** o **Archivo-guardar** o

En SQL Developer podem recuperar un script amb **ctrl-o** o **Archivo-abrir** o

En SQL Developer podem executar un script amb **F5** o una sentència amb **F9** o

Activar l'execida => set serveroutput on (F5)



Connecta amb SYSTEM o SYS a la primera pdb

Crea usuari usuari1. Donar permisos a usuari1. Connecta amb **usuari1**

Tasca 1 Declaracions => Sintaxis



Indica quines declaracions donarien error i perquè. Després prova-les en sql-developer

SET SERVEROUTPUT ON

DECLARE

```
primera number:=5.0;
segona number not null;
fixa constant varchar2(20);
cadena varchar2;
valor1 number not null:=4;
valor2 number:=valor1/2;
valor3 number:=valor4+valor2;
valor4 number:= default 5;
cad varchar2(10);
proxim valor1%TYPE;
numero number(4,2):=150.3;
num1, num2 number;
CAD varchar2(12);
cad varchar2(2):='HOLA';
4num number;
vdia sysdate;
v1 varchar2(4) := 4;
v2 number := '6';
v3 number := '6.4';
b1 boolean := TrUe;

BEGIN

    dbms_output.put_line ('La primera variable val ' || primera);
    dbms_output.put_line ('La suma val ' || primera+valor1);
    -- Posa ací més línies per provar TOTS els resultats

END;
```

=====

Mostra en un document tots el errors emsos per ORACLE i com els has arreglat.

Investiga: Quina és la diferència entre CHAR i VARCHAR2 ??

Investiga: Quina és la diferència entre NUMBER i NUMBER(15) ??



Tasca 2

Conversió de dates

El tipus de dades '**DATE**' és molt potent però necessita d'un tractament específic.

Escriu un Bloc anònim que demane (a part del títol, autor i preu d'un llibre) la data de publicació amb una variable de substitució (en char) i inserisca una fila en la taula LLIBRES. (compte amb els camps **datapub** i **datareg**). Els valors de les cadenes s'han de posar en majúscules. Utilitza les funcions necessàries.

=====

Script de creació de taula llibres i de tres files de dades:

```
drop table llibres;

CREATE TABLE llibres(
titol VARCHAR2(60) NOT NULL,
autor varchar2(30) not null,
datapub DATE,
editorial VARCHAR2(30),
edicio VARCHAR2(12),
isbn VARCHAR2(25),
preu number( 6,2),
datareg date,
CONSTRAINT pk_codi PRIMARY KEY(titol,autor)
);
```

Tasca 3

Conversió de dates

El tipus de dades '**DATE**' és molt potent però necessita d'un tractament específic.

Escriu un Bloc anònim que demane una data (es demanarà amb el tipus date) i a continuació mostre per pantalla: l'any, el mes, el dia del mes, el dia de la semana en nombre i en lletra.

Tasca 4

Executa el següent codi i explica els resultats

```
set serveroutput on
declare
    d1 date; d2 date;
begin
    d1:=sysdate;dbms_session.sleep(1);
    d2:=sysdate;dbms_session.sleep(1);
    dbms_output.put_line('d1 val ' || d1);
    dbms_output.put_line('d2 val ' || d2);
    if (d1=d2) then
        dbms_output.put_line('d1 i d2 son iguals');
    else
        dbms_output.put_line('d1 i d2 NO son iguals');
    end if;
end;
```



Butlletí repàs PL/SQL

Utilitza SQL Developer. Connecta amb SYSTEM a la primera pdb (con_id=3)

Crea usuari usuari1 en la primera pdb (con_id=3)

Donar permisos necessaris a usuari1.

Connecta amb **usuari1 a la primera pdb (con_id=3)**

Tasca 1

Crea un bloc anònim que demane dos números (utilitza dos variables de substitució) i diga la suma, la multiplicació, la resta, la divisió dels números.

Tasca 2

Càlcul de la superfície de diverses figures geomètriques. (utilitza tres variables de substitució)

Rectangle	$base * altura$
Quadrat	$base^2$
Triangle	$(base * altura) / 2$
Cercle	$\pi * radi^2$

Tasca 3

Crea un fragment de codi que donat un mes de l'any en número (de 1 a 12), i a continuació, que mostre el nom del mes. (utilitza una variable de substitució)

Tasca 4

Fes un altre fragment que donat un mes (utilitza una variable de substitució), en lloc de mostrar el nom del mes, mostre els dies que té. (Considerem que febrer sempre té 28 dies).

Tasca 5

Un altre fragment que donat un dia de la setmana en número (d'1 a 7) i, a continuació que mostre si el dia introduït és entre setmana o cap de setmana. (utilitza una variable de substitució)

Tasca 6

Un bloc anònim que reba una cadena i la visualitze a l'inrevés. (utilitza una variable de substitució). Transforma el bloc anònim en un procediment.

Executa el procediment utilitzant notació posicional i notació nominal.

Tasca 7

Una fragment de codi que retorne el nombre d'anys complets que hi ha entre dues dates que es passen com a strings. Transforma el bloc anònim en una funció.

Tasca 8

Un bloc anònim que escriga solament caràcters alfabètics substituint qualsevol altre caràcter per blancs a partir d'una cadena que es passarà en una variable de substitució

TIPS: Pots utilitzar funcions predefinides de PL/SQL

	round(n)	trunc(n)	mod(n,m)	floor(n)	Ceil(n)
Length(s)	Lower(s)	Upper(s)	Ascii(s)	Chr(n)	Substr(s,n [,1])
trim(s)	instr(c,s)		initcap(s)	replace(s,s1,s2)	

Data2 - data1 : resultat , num de dies entre les dos dates

Data1 + num : resultat , data de 'num' de dies més que Data1



Butlletí : Crear procediments amb pl/sql

Per realitzar esta pràctica, utilitzarem Oracle **SQL Developer**

Amb l'usuari system, connectat en la primera pdb,

Crear usuari **usuari3** (donar-li contrasenya i permisos de connexió i creació de procediments i funcions)

Dona-li permisos de crear taules i assigna quota (10M) en el tablespace per defecte.

Connecta a la primera pdb amb l'usuari creat (**usuari3**)

Realitza els següents exercicis i practica (prova que funcionen amb exemples)

0.- Crea la taula llibres i ompli-la (5 files)

```
drop table llibres;  
CREATE TABLE llibres(  
codi NUMBER(6) PRIMARY KEY,  
titol VARCHAR2(50) NOT NULL,  
autor VARCHAR2(30),  
editorial VARCHAR2(40),  
preu number(8,2),  
datadalta date );
```

1) *Escriure un bloc anònim PL/SQL que escriga el text 'HOLA'*

```
SET SERVEROUTPUT ON  
BEGIN  
DBMS_OUTPUT.PUT_LINE('HOLA');  
END;
```

2) *Escriure un bloc PL/SQL que compte el nombre de files que hi ha en la taula **llibres**, deposita el resultat en la variable v_num, i visualitza el seu contingut.*

2.1 – *Guardar el bloc en un fitxer anomenat PROG01.SQL en c:\users\oracle\Documents*

3) *Carregar i executar el bloc guardat en l'arxiu PROG01.SQL de c:\users\oracle\Documents*

4) *Escriure un procediment anomenat 'suma2n' que reba dos números i visualitze la seua suma.
Fes servir la funció amb un exemple.*

=====

5) *Codificar un procediment 'inreves' que reba una cadena i la visualitze a l'inrevés.*

Hola > aloH

Fes servir la funció amb un exemple.

6) *Escriure una funció que reba una data i retorne l'any, en número, corresponent a aqueixa data.
Fes servir la funció amb un exemple.*



7) Escriure un bloc PL/SQL que faça ús de la funció anterior. Guarda el bloc en un fitxer prog02.sql



8) Donat el següent procediment, basat en la taula abans creada:

```
CREATE OR REPLACE PROCEDURE alta_llibre (  
v_num llibres.llibreid%TYPE,  
v_titol llibres.titol%TYPE default 'sense titol',  
v_autor llibres.autor%TYPE DEFAULT 'anònim')  
IS  
BEGIN  
INSERT INTO llibres  
VALUES (v_num , v_titol, v_autor);  
END crear_llibre;
```

Detecta els errors i corregix-los (compila i executa primer)

Indicar quins de les següents cridades al procediment són correctes i quins incorrectes, en aquest últim cas escriure la crida correcta usant la notació posicional (en els casos que es puga):

- 1º. crear_llibre;
- 2º. crear_llibre(50);
- 3º. crear_llibre('Hackers');
- 4º. crear_llibre(50,'Hackers');
- 5º. crear_llibre('Hackers', 50);
- 6º. crear_llibre('Hackers', 'McClure');
- 7º. crear_llibre(50, 'Hackers', 'McClure');
- 8º. crear_llibre('Hackers', 50, 'McClure');
- 9º. crear_llibre('McClure', 'Hackers');
- 10º. crear_llibre('McClure', 50);

9) Desenvolupar una funció que retorne el nombre d'anys complets que hi ha entre dues dates que es passen com a arguments.

Fes servir la funció amb un exemple.

10) Escriure una funció que, fent ús de la funció anterior retorne els triennis que hi ha entre dues dates. (Un trienni són tres anys complets).

Fes servir la funció amb un exemple.

11) Codificar un procediment que reba una llista de **fins** a 5 números i visualitze la seua suma.

Fes servir el procediment amb un exemple.

12) Escriure una funció que retorne solament caràcters alfabètics substituint qualsevol altre caràcter per blancs a partir d'una cadena que es passarà en la crida.

Fes servir la funció amb un exemple.



13) Implementar un procediment que reba un import i visualitze el desglossament del canvi en unitats monetàries de 1c, 2c, 5c, 10c, 20c, 50c, 1€, 2€, 5€, 10€, 20€, 50€, 100€, 200€, 500€ en ordre invers al que apareixen ací enumerades.

Fes servir el procediment amb un exemple.

14) Codificar un procediment que permeta esborrar un llibre el número(num_id) del qual es passarà en la cridada al procediment.

Nota: El procediment anterior retornarà el missatge

<< Procedimiento PL/SQL terminado con éxito >> encara que no existisca el número i, per tant, no s'esborre el llibre.

Pots fer que s'informe d'aquesta situació quan es produeixca ?

Fes servir el procediment amb exemple per comprovar que funciona tant si existeix com si no.

15) Escriure un procediment que modifique el títol d'un llibre. El procediment rebrà com a paràmetres el número del llibre i el títol nou.

Nota: L'indicat en la nota de l'exercici anterior es pot aplicar també a aquest.

Fes servir el procediment amb exemple per comprovar que funciona tant si existeix com si no.

16) Utilitzant el **DD** (Diccionari de Dades d'Oracle) Visualitzar tots els procediments i funcions de l'**usuari** (el que s'està utilitzant per fer els problemes del butlletí) emmagatzemats en la base de dades i la seua situació (vàlid o invalid).

Pràctica : Crear procediment emmagatzemat en SQL Developer



Algun punt d'esta pràctica no funciona molt bé ... 😬
Esbrina perquè i fes les modificacions oportunes per a que funcione.

Per realitzar esta pràctica utilitza la .ova facilitada (windowsOracle6).
Totes les contrasenyes son 1234 . Fes servir l'eina **SQL Developer**

1. En la primera PDB de la CDB de l'ORACLE_SID: (<== *mostra quin valor té*)
Amb usuari system :
2. Crear usuari **usuari1** (donar-li contrasenya i permisos de connexió, crear taules, crear procediments)
3. Crear usuari **usuari2** (donar-li contrasenya i permisos de connexió)

Amb usuari usuari1 en pdb1:

4. Crear taula **llibres3**
5. Crear un procediment (nom *amay*) que pose totes les dades de la taula en majúscules
6. Donar-li a usuari2 permís d'execució del procediment creat.
7. Inserir 3 files amb dades en majúscules i minúscules
8. Llistar dades de la taula

Amb usuari usuari2 en pdb1:

9. Executar procediment (execute usuari1.amay;)
10. Explorar resultats (llistar dades) -- (comenta que succeeix i perquè)
11. Inserir 2 files més amb dades en majúscules i minúscules (comenta que succeeix i perquè)
12. Explorar resultats (comenta que succeeix i perquè)
13. Contesta: Com podríem solucionar-ho ??

14. Des d'usuari1, crea un procediment que inserisca llibres de la editorial 'Sintesis' , anomenat *inserirSintesis* , on se li pase com a paràmetre, el nom del llibre i el preu.
15. El procediment buscarà l'últim codi de la taula, l'incrementarà en 1 per a donar de alta el nou registre

16. Dona permís a usuari2 per executar el nou procediment i prova'l des d'usuari2

Documentar el procés

Documentar els errors o dificultats trobades i documentar-les explicant la solució adoptada

Entregar el document en format PDF

Taula **Llibres3** CREATE TABLE llibres3(codi NUMBER(6) PRIMARY KEY, titol VARCHAR2(30) NOT NULL, editorial VARCHAR2(30), preu NUMBER(8,2), datadalta date);