

Introducere in PL/SQL

Motivatie PL/SQL

- Structured Query Language (SQL) este limbajul standard pentru manipularea bazelor de date relationale

- Sa consideram urmatoarea cerere:

```
SELECT first_name, department_id, salary  
FROM employees;
```

- Cerere: In functie de fiecare department si de salariul angajatiilor sa se acorde un bonus.

Imagine generala PL/SQL

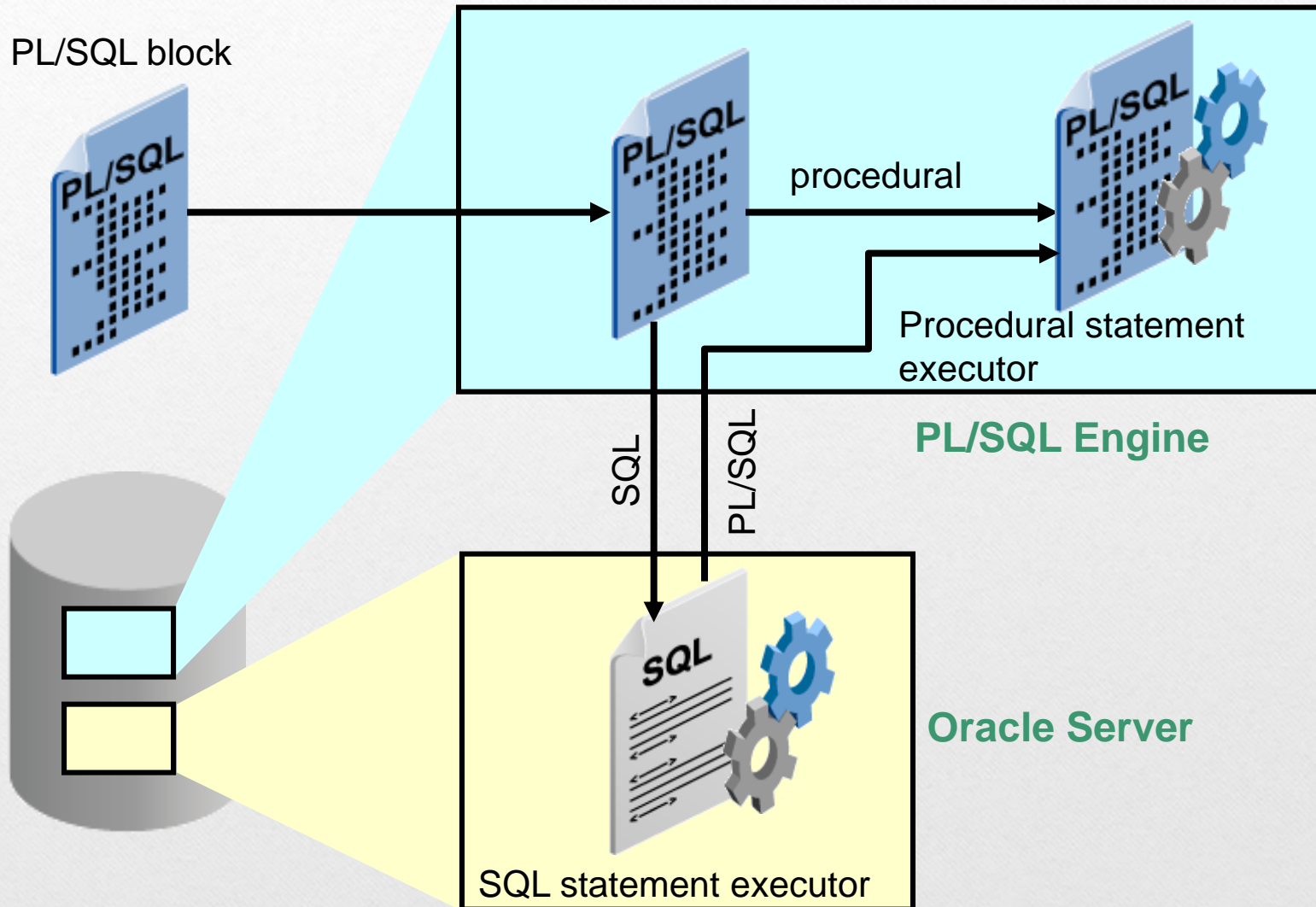
- PL/SQL:
 - Poate fi definit ca “Procedural Language extension to SQL”
 - Este limbajul procedural standard Oracle pentru manipularea bazelor de date relationale
 - Integreaza facilitatile unui limbaj procedural peste SQL



Generalitati PL/SQL

- PL/SQL:
 - Se caracterizeaza printr-o structura de bloc in care sunt integrate comenzile ce urmeaza a fi prelucrate. Mentenanta codului este mult mai usoara pe o astfel de structura.
 - Oferă facilitățile unui limbaj procedural:
 - Variabile, constante și tipuri de date
 - Structuri pentru adăugarea de condiții și pentru controlul execuției (case, if, for, loop, while)
 - Oferă facilități pentru stocarea și reutilizarea codului.

Procesarea blocurilor PL/SQL

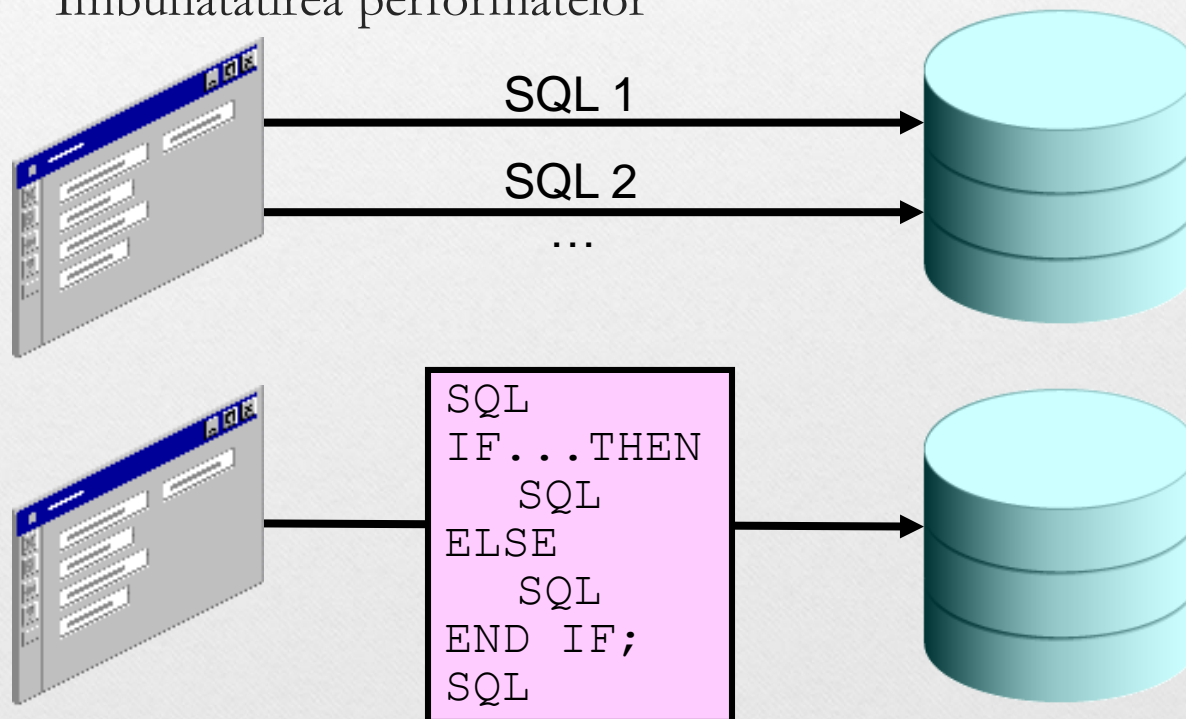


Procesarea blocurilor PL/SQL

- Bloc PL/SQL:
 - In general un bloc PL/SQL poate sa contina sintaxa specifica unui limbaj procedural si comenzi SQL
 - Executiei blocului PL/SQL se intrerupe pentru executia comenzilor SQL
 - O comanda SQL poate sa contina/sa implice executia unui bloc PL/SQL
 - In ambele situatii se asteapta finalizarea contextului apelat si eventual utilizarea valorilor returnate in contextual apelant

Avantjele utilizarii blocurilor PL/SQL

- Utilizarea comenzilor specifice unui limbaj procedural
 - SQL= **ce sa faca** vs. PL/SQL=**ce sa faca + cum sa faca**
- Imbunatatirea performatelor



Avantjele utilizarii blocurilor PL/SQL

- Modularizarea codului
- Integrarea unor utilitare/programe externe
- Portabilitate
- Tratarea erorilor
- PL/SQL utilizeaza aceleasi tipuri de date din SQL (cu mici extensii) si comenzi specifice SQL

Structura blocului PL/SQL

- DECLARE (optional)
 - Variabile, cursoare, exceptii, tipuri de date locale
- BEGIN (mandatory)
 - Comenzi SQL
 - Sintaxa PL/SQL
- EXCEPTION (optional)
 - Ce actiuni sa intreprinda cand are loc o exceptie
- END; (mandatory)



Tipuri de blocuri PL/SQL

- Anonymous

```
[DECLARE]

BEGIN
    --statements

[EXCEPTION]

END;
```

Procedure

```
PROCEDURE name
IS
BEGIN
    --statements

[EXCEPTION]

END;
```

Function

```
FUNCTION name
RETURN datatype
IS
BEGIN
    --statements
    RETURN value;
[EXCEPTION]

END;
```


Tipuri de blocuri PL/SQL

- Un program PL/SQL poate fi alcatuit din mai multe blocuri care pot fi independente sau incuibarite
- Pentru a defini un program pot fi utilizate 3 tipuri de blocuri:
 - **Anonime** sunt blocuri care nu primesc nume; se declar la un anumit moment si nu sunt stocate (se declara si se compileaza de fiecare data cand sunt utilizate)
 - **Proceduri** - blocuri stocate care primesc nume
 - **Functii** - blocuri stocate care au nume + trebuie sa returneze
- **Nota:** subprogramele pot fi reutilizate in diferite contexte;

Utilizare blocurilor PL/SQL

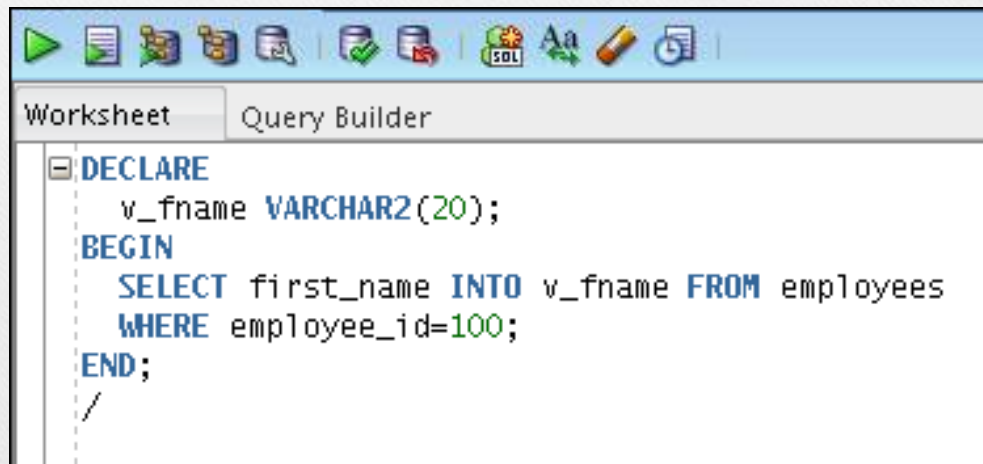
Tools Constructs
Anonymous blocks
Application procedures or functions
Application packages
Application triggers
Object types



Database Server Constructs
Anonymous blocks
Stored procedures or functions
Stored packages
Database triggers
Object types

Structura unui bloc anonim

- Definirea unui bloc anonim in SQL Developer:

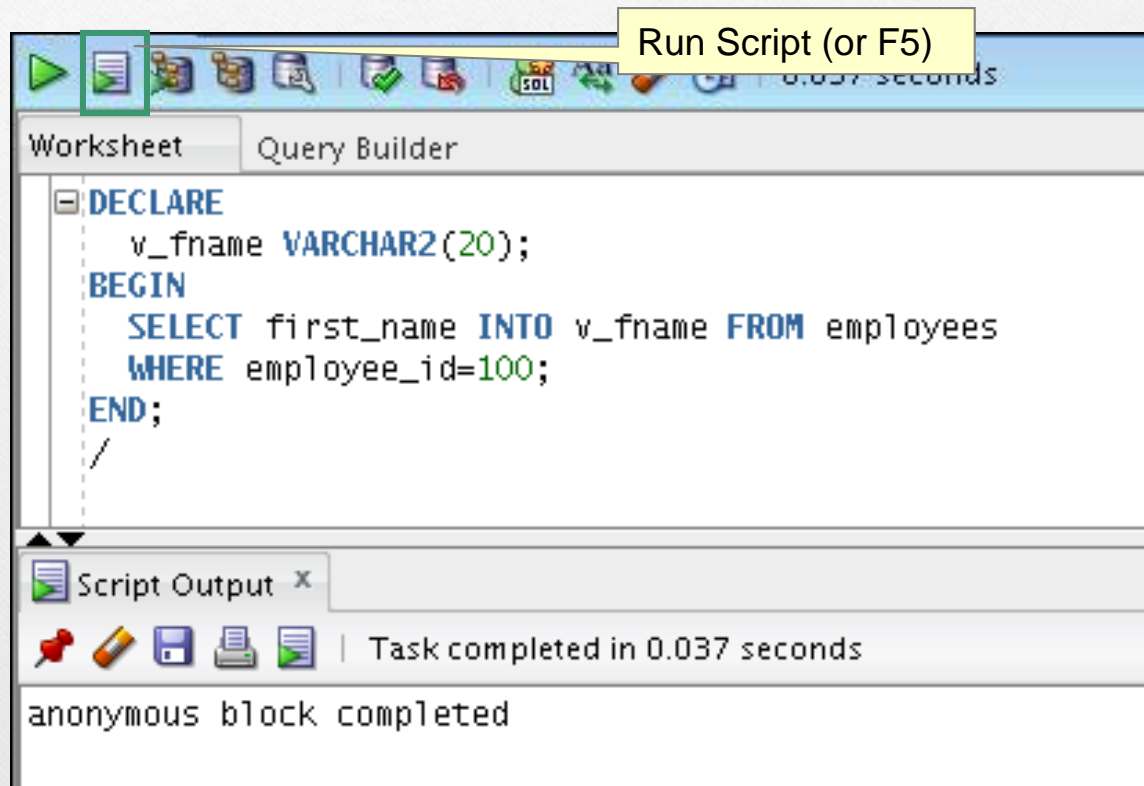


The screenshot shows the SQL Developer Query Builder window. The toolbar at the top includes icons for running queries, saving, and other database operations. The window has two tabs: 'Worksheet' and 'Query Builder'. The 'Query Builder' tab is active, displaying the following SQL code in a text editor:

```
DECLARE
  v_fname VARCHAR2(20);
BEGIN
  SELECT first_name INTO v_fname FROM employees
  WHERE employee_id=100;
END;
/
```

Executia unui bloc anonim

- Selectati butonul Run Script (F5) pentru a executa blocul anonim:



Afisarea rezultatului rularii unui bloc PL/SQL

1. Pentru a permite afisarea rezultatului rularii unui bloc PL/SQL se adauga urmatoarea comanda inaintea blocului:

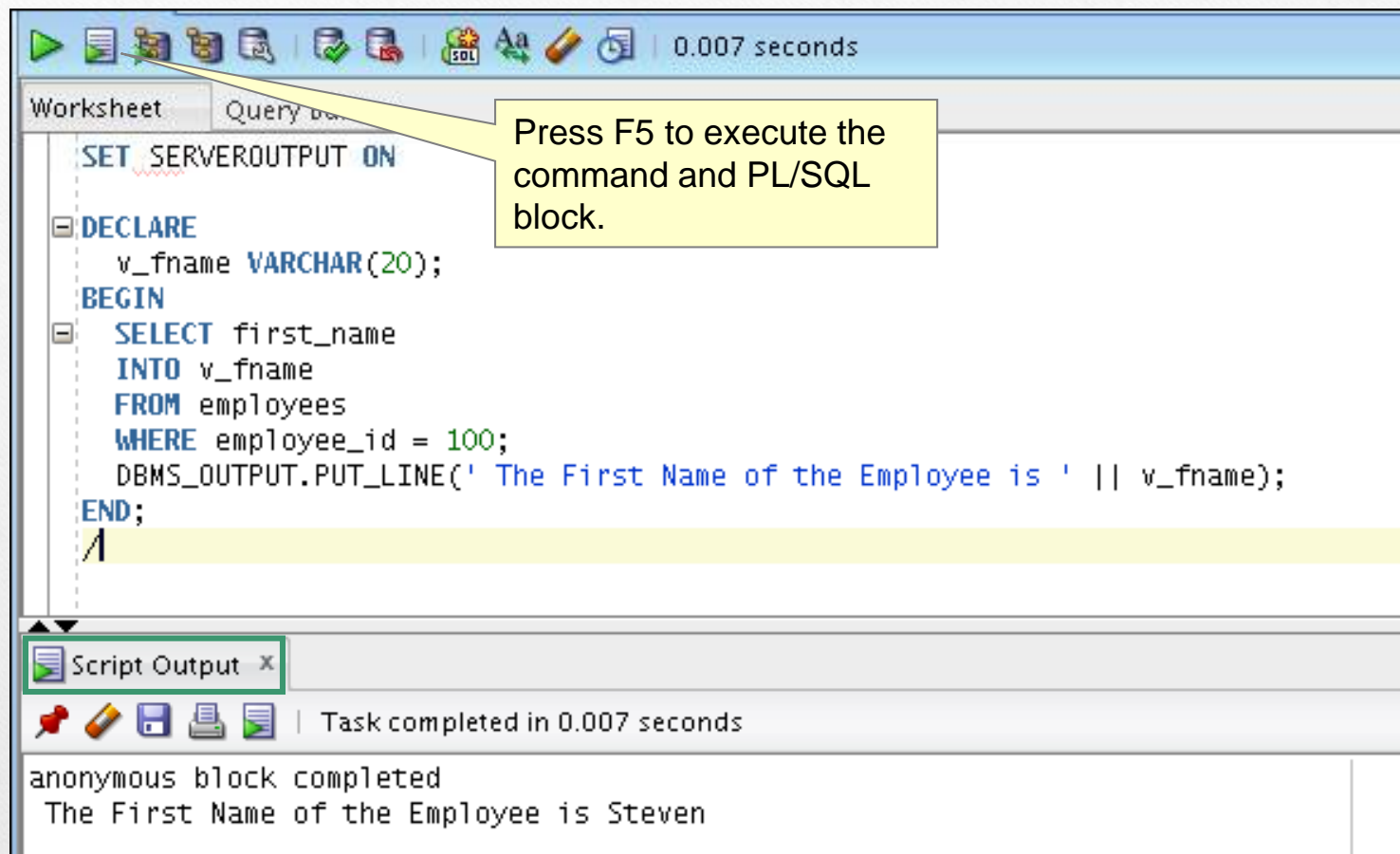
```
SET SERVEROUTPUT ON
```

2. Pentru a afisa un anumit mesaj folositi urmatoarea procedura din pachetul DBMS_OUTPUT:

- DBMS_OUTPUT.PUT_LINE

```
DBMS_OUTPUT.PUT_LINE(' The First Name of the  
Employee is ' || v_fname);  
...
```

Rezultatul rularii unui bloc PL/SQL



The screenshot displays the SQL Developer interface. The main window shows a PL/SQL block in the 'Worksheet' tab. The code is as follows:

```
SET SERVEROUTPUT ON  
  
DECLARE  
    v_fname VARCHAR(20);  
BEGIN  
    SELECT first_name  
    INTO v_fname  
    FROM employees  
    WHERE employee_id = 100;  
    DBMS_OUTPUT.PUT_LINE(' The First Name of the Employee is ' || v_fname);  
END;
```

A yellow callout box points to the 'Run' button (a green play icon) in the toolbar, with the text: "Press F5 to execute the command and PL/SQL block."

Below the code editor, the 'Script Output' window is visible, showing the results of the execution:

```
Task completed in 0.007 seconds  
anonymous block completed  
The First Name of the Employee is Steven
```


Quiz

- Un bloc PL/SQL *trebuie* sa contina urmatoarele 3 sectiuni:
 - **Declarativa**, care incepe cu DECLARE si se continua pana la partea executabila
 - **Executabila**, este marcata prin BEGIN si se termina cu END
 - **Tratarea erorilor**, este marcata prin cuvantul cheie EXCEPTION si este inclusa in partea executabila
- a. Adevarat
- b. Fals

Exercitii (I)

Ce subpuncte definesc un bloc PL/SQL valid?

a. BEGIN

END;

b. DECLARE

v_amount INTEGER(10);

END;

c. DECLARE

BEGIN

END;

d. DECLARE

v_amount INTEGER(10);

BEGIN

DBMS_OUTPUT.PUT_LINE(v_amount);

END;

Exercitii (II)

1. Definiti un bloc anonim care afiseaza mesajul “*Hello World.*”

Exercitii pregatitoare curs 3:

1. Definiti un bloc anonim care citeste de la tastatura un cod de angajat si ii afiseaza numele.
2. Definiti un bloc anonim care citeste de la tastatura un nume de angajat si afiseaza salariul.