

4

Database Interfaces

By: [Arif Basofi](#)

Tujuan

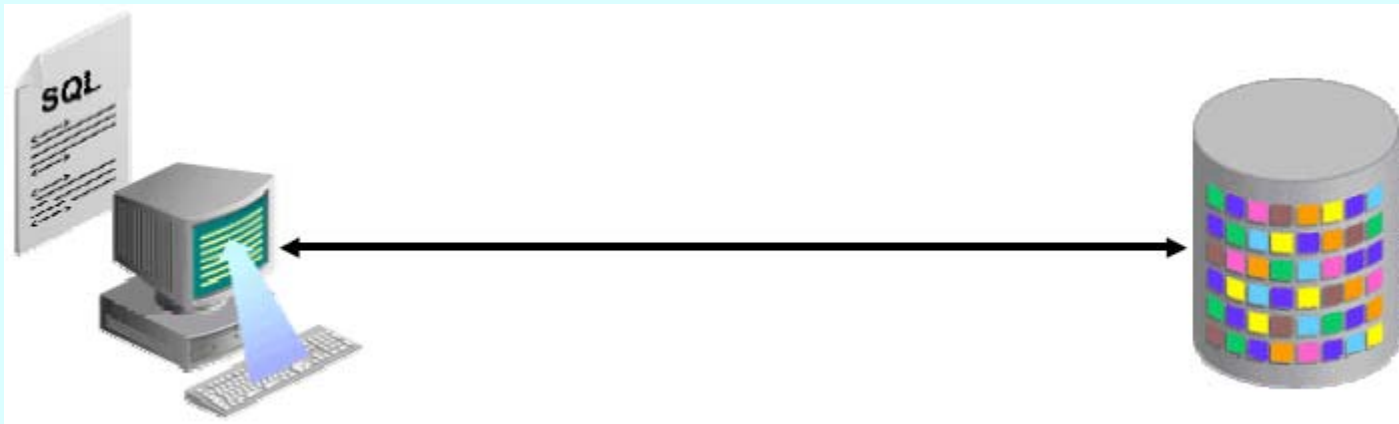
- Menggunakan SQL*Plus dan iSQL*Plus untuk mengakses Database Oracle 10g
- Menjelaskan struktur logika dari sebuah table
- Menggunakan SQL untuk melakukan query, memanipulasi dan mendefinisikan data
- Mengidentifikasi beberapa interface database

Apa itu SQL?

- SQL menyediakan statemen untuk berbagai tugas, mencakup:
 - Query data
 - Operasi DML: insert, update, dan delete baris di dalam suatu tabel.
 - Operasi DDL: create, replace, alter, dan drop object
 - kontrol akses kepada database dan objectnya
- SQL mempersatukan semua tugas yang terdahulu di dalam satu bahasa konsisten.

Menggunakan SQL

- Ada beberapa tool untuk berinteraksi dengan database menggunakan SQL, yaitu :
 - Oracle SQL*Plus and iSQL*Plus
 - Oracle Forms, Reports, and Discoverer (tool data mining)
 - Oracle Enterprise Manager
 - Third-party tools



Melihat SQL dalam Enterprise Manager

The image shows two screenshots from Oracle Enterprise Manager. The top screenshot is a dialog box titled "Create Tablespace". It has three tabs: "General", "Storage", and "Thresholds". The "General" tab is selected, and the "Name" field contains the text "INVENTORY". There are three buttons at the top right: "Show SQL", "Cancel", and "OK". An arrow points from the "Show SQL" button to the bottom screenshot. The bottom screenshot shows the SQL command generated for creating the tablespace. The breadcrumb path is "Database: orcl.us.oracle.com > Tablespaces > Create Tablespace" and it is "Logged in As SYS". The SQL command is as follows:

```
CREATE SMALLFILE TABLESPACE "INVENTORY" DATAFILE
'*/u01/app/oracle/product/10.1.0/oradata/orcl/inv01.dbf' SIZE
100M LOGGING EXTENT MANAGEMENT LOCAL SEGMENT SPACE
MANAGEMENT AUTO
BEGIN DBMS_SERVER_ALERT.SET_THRESHOLD
(9000, NULL, NULL, NULL, NULL, 1, 1, NULL, 5, 'INVENTORY'); END;
```

- Dengan menggunakan **Enterprise Manager (EM)** Database Control memudahkan dalam men-generate objek dengan SQL yang dibentuk.
- Untuk melihat perintah SQL yang dibangun, dapat klik **Show SQL**.

Apa itu SQL*Plus?

- Perintah berbasis Command-line tool.
- Dapat digunakan secara interaktif atau dalam mode batch.
- Dapat dijalankan dari aplikasi yg ter-install pertama kali, atau melalui command prompt.

```
$ sqlplus /nolog
SQL*Plus: Release 10.1.0.2.0 - Production on Tue Feb
17 06:17:14 2004
Copyright (c) 1982, 2004, Oracle. All rights
reserved.
SQL> connect ric
Enter password:
Connected.
SQL> SELECT * FROM dual;

D
-
X
SQL>
```

Apa itu *iSQL*Plus*?

ORACLE[®]
*iSQL*Plus*

Help

Login

* Indicates required field

* Username

* Password

Connect Identifier

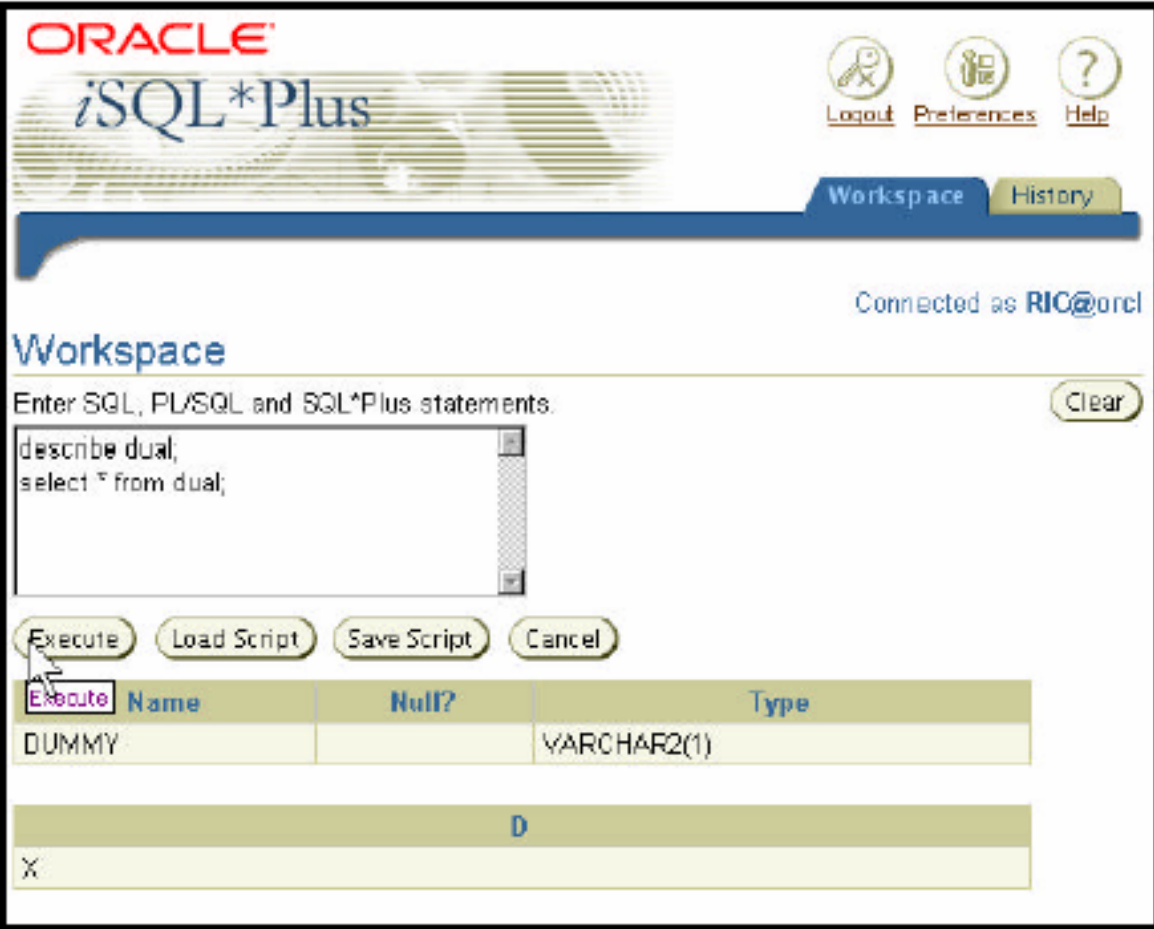
Login

Help

Copyright © 2003, Oracle. All rights reserved.

- ***iSQL*Plus*** adalah sebuah interface berbasis browser untuk mengakses database Oracle.
- ***iSQL*Plus*** adalah sebuah komponen dari produk SQL*Plus.
- ***iSQL*Plus*** memiliki proses **listener** server-side yang harus dinyalakan sebelum terkoneksi dengan sebuah browser.

Menggunakan iSQL*Plus



ORACLE[®]
iSQL*Plus

Logout Preferences Help

Workspace History

Connected as RIC@orcl

Workspace

Enter SQL, PL/SQL and SQL*Plus statements. Clear

```
describe dual;  
select * from dual;
```

Execute Load Script Save Script Cancel

Execute	Name	Null?	Type
	DUMMY		VARCHAR2(1)
		D	
X			

Melihat deskripsi Data

Workspace

Enter SQL, PL/SQL and SQL*Plus statements. Clear

```
desc hr.employees
```

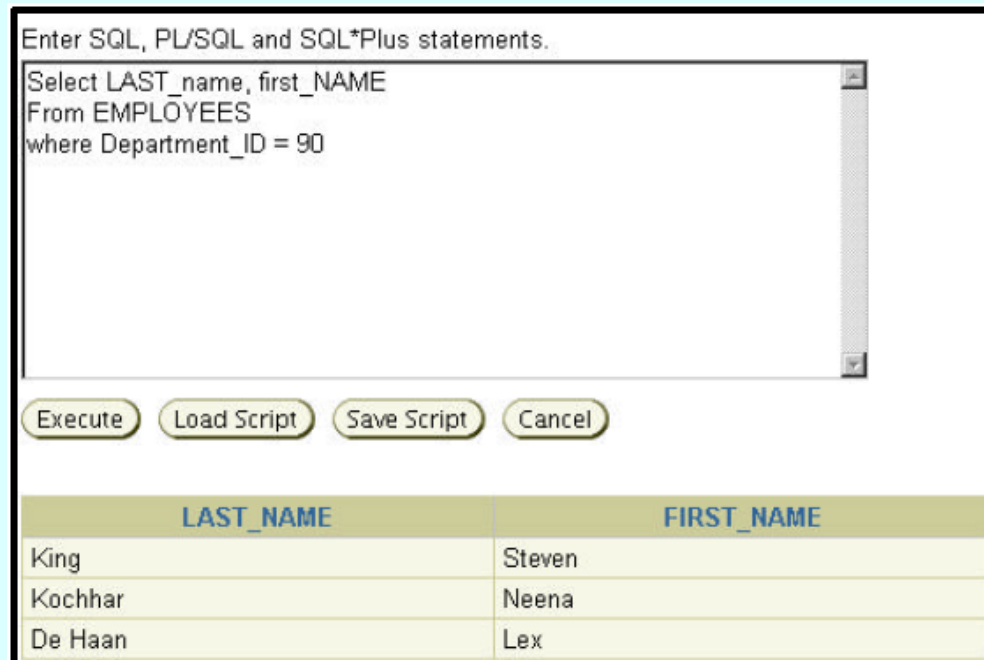
Execute Load Script Save Script Cancel

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

Meng-Query Data

Perintah SELECT mempunyai tiga bagian dasar:

- SELECT List
- FROM clause
- WHERE condition (optional)



The screenshot shows a window titled "Enter SQL, PL/SQL and SQL*Plus statements." with a text area containing the following SQL query:

```
Select LAST_name, first_NAME  
From EMPLOYEES  
where Department_ID = 90
```

Below the text area are four buttons: "Execute", "Load Script", "Save Script", and "Cancel".

Below the buttons is a table with the following data:

LAST_NAME	FIRST_NAME
King	Steven
Kochhar	Neena
De Haan	Lex

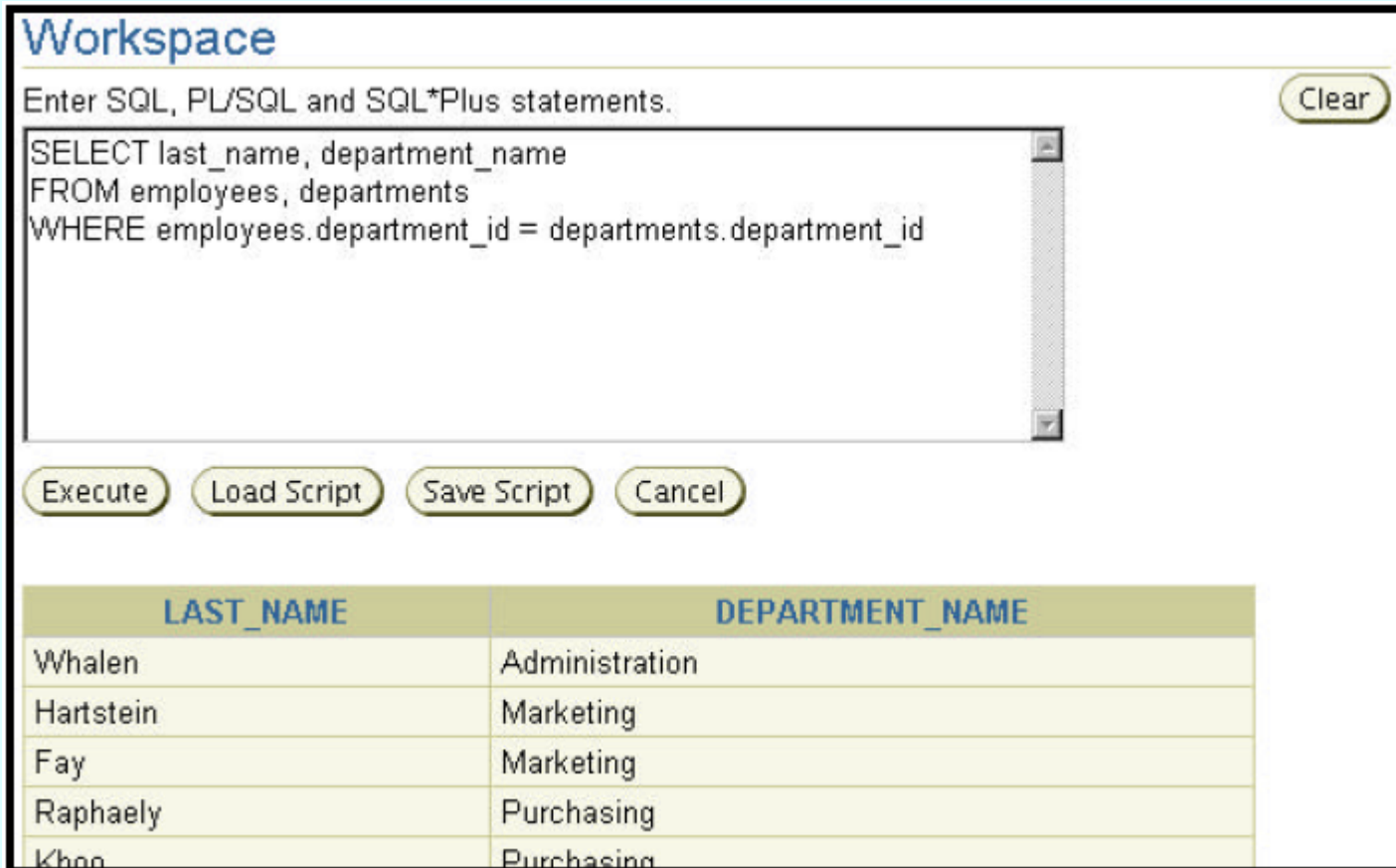
Mengurutkan Data

```
SQL> SELECT last_name, department_id, phone_number  
2 FROM employees  
3 ORDER BY last_name;
```

LAST_NAME	DEPARTMENT_ID	PHONE_NUMBER
Abel	80	011.44.1644.429267
Ande	80	011.44.1346.629268
Atkinson	50	650.124.6234
Austin	60	590.423.4569
Baer	70	515.123.8888
Baida	30	515.127.4563
Banda	80	011.44.1346.729268

Menggabungkan (Join) Tabel

Mendapatkan data dari lebih dari satu table



The screenshot shows a 'Workspace' window with a text area containing the following SQL query:

```
SELECT last_name, department_name  
FROM employees, departments  
WHERE employees.department_id = departments.department_id
```

Below the text area are buttons for 'Execute', 'Load Script', 'Save Script', and 'Cancel'. A 'Clear' button is also present in the top right corner of the workspace.

The results of the query are displayed in a table below the workspace:

LAST_NAME	DEPARTMENT_NAME
Whalen	Administration
Hartstein	Marketing
Fay	Marketing
Raphaely	Purchasing
Khoo	Purchasing

Memanipulasi Data

```
SQL> INSERT INTO employees
  2  (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL, PHONE_NUMBER,
  3  HIRE_DATE, JOB_ID, SALARY, COMMISSION_PCT,
  4  MANAGER_ID, DEPARTMENT_ID)
  5  VALUES
  6  (9999, 'Bob', 'Builder', 'bob@abc.com', NULL, sysdate,
  7  'IT_PROG', NULL, NULL, 100, 90);
```

1 row created.

```
SQL> UPDATE employees SET SALARY=6000
  2  WHERE EMPLOYEE_ID = 9999;
```

1 row updated.

```
SQL> DELETE from employees
  2  WHERE EMPLOYEE_ID = 9999;
```

1 row deleted.

Mendefinisikan Data (DDL)

Enter SQL, PL/SQL and SQL*Plus statements.

```
CREATE TABLE local_temp  
(temp_date DATE,  
hi_temp NUMBER(4,1),  
lo_temp NUMBER(4,1));  
  
ALTER TABLE local_temp ADD (mean_temp NUMBER(4,1));  
  
DROP TABLE local_temp;
```

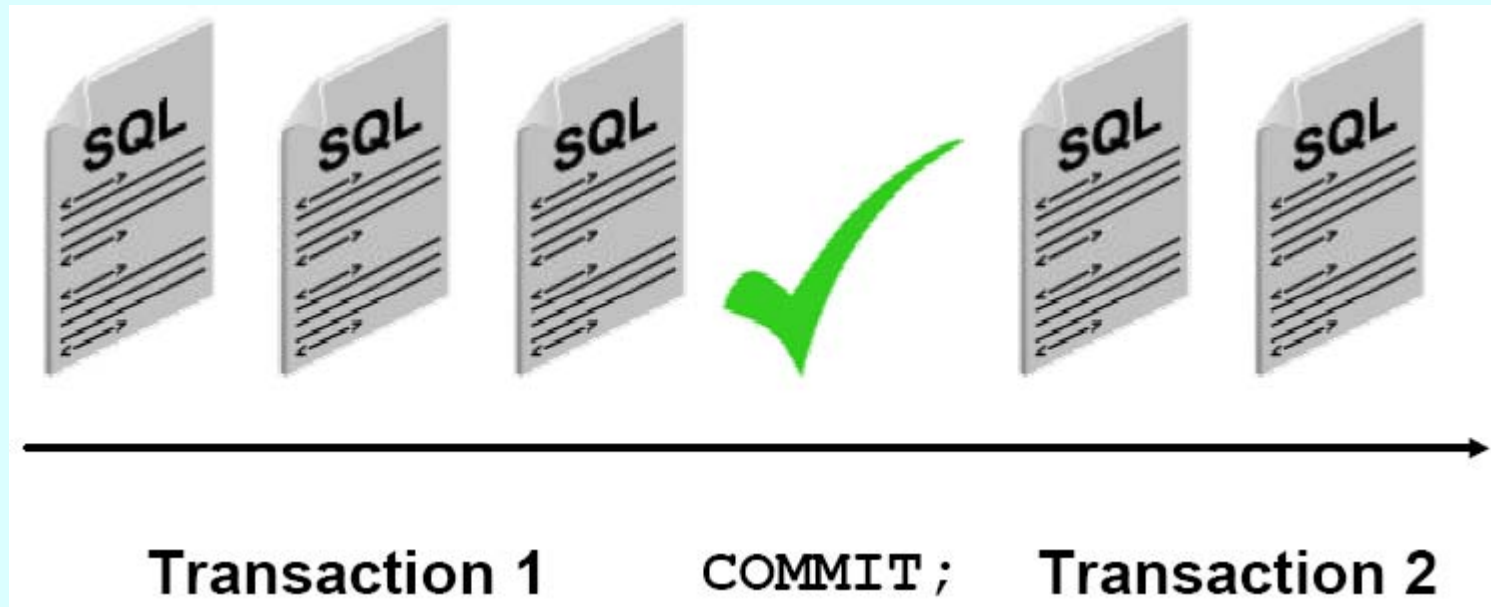
Execute Load Script Save Script Cancel

Table created.

Table altered.

Table dropped.

Overview Transaksi



Perintah untuk mengontrol Transaksi

```
SQL> SELECT * FROM local_temp;  
no rows selected
```

```
SQL> INSERT INTO local_temp VALUES  
  2  (SYSDATE, 76, 58);  
1 row created.
```

```
SQL> SELECT * from local_temp;  
TEMP_DATE      HI_TEMP      LO_TEMP  
-----  
27-OCT-03          76          58
```

```
SQL> ROLLBACK;  
Rollback complete.
```

```
SQL> SELECT * FROM local_temp;  
no rows selected
```

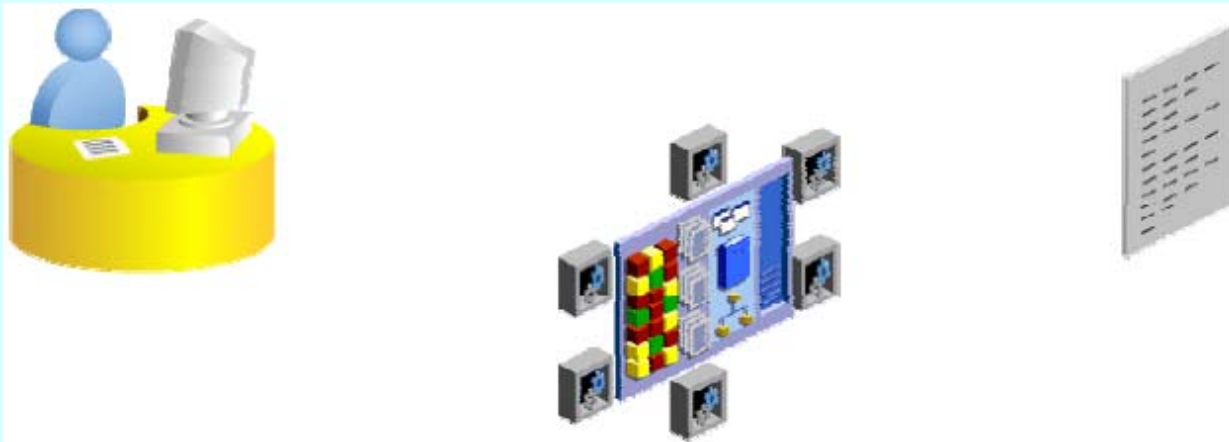

Penguncian Data

Database Oracle 10g secara **otomatis mengunci** data sehingga hanya satu user yang dapat merubah data pada satu waktu.



Kategori Perintah Lainnya

- Session Control Statements:
 - mengelola properti-properti dari user session
- System Control Statements:
 - mengelola properti-properti dari Oracle instance
- Embedded SQL Statements:
 - perintah-perintah SQL yang berada didalam procedural language program



Apakah PL/SQL itu?

PL/SQL adalah sebuah block-structured language yang memperluas SQL dengan:

- Deklarasi:
 - Variabel
 - Konstanta
 - Kursor
- Struktur control:
 - Kontrol kondisi
 - Kontrol perulangan
 - Kontrol sekuensial
- Penanganan error



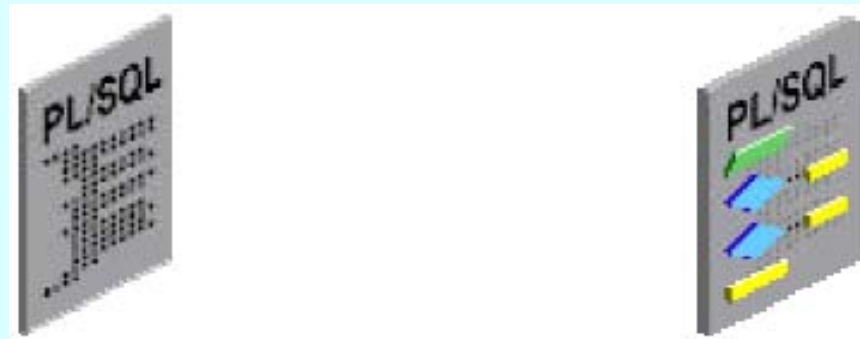
Contoh blok PL/SQL

```
DECLARE
  qty_on_hand NUMBER(5);
BEGIN
  SELECT quantity INTO qty_on_hand FROM
    inventory
  WHERE product = 'TENNIS RACKET'
  FOR UPDATE OF quantity;
  IF qty_on_hand > 0 THEN -- check quantity
    UPDATE inventory SET quantity = quantity - 1
    WHERE product = 'TENNIS RACKET';
    INSERT INTO purchase_record
    VALUES ('Tennis racket purchased', SYSDATE);
  ELSE
    INSERT INTO purchase_record
    VALUES ('Out of tennis rackets', SYSDATE);
  END IF;
  COMMIT;
END;
```

Penggunaan PL/SQL

Blok dari PL/SQL digunakan pada:

- Blok anonymous
- Functions
- Procedures
- Packages
- Triggers
- Object Types



Apa itu Java?

Java adalah standar industri, bahasa pemrograman berorientasi obyek.

Termasuk konsep berikut ini:

- Java Virtual Machine (JVM) yang memungkinkan berjalan di banyak platform.
- Mekanisme pengelolaan penyimpanan secara otomatis.
- Syntax yang tidak jauh berbeda dengan C dan memerlukan pengetikan secara tepat.



Oracle dan Java

A PL/SQL function:

```
FUNCTION balance (acct_id NUMBER) RETURN NUMBER IS
acct_bal NUMBER;
BEGIN
  SELECT bal INTO acct_bal FROM accts
    WHERE acct_no = acct_id;
  RETURN acct_bal;
END;
```

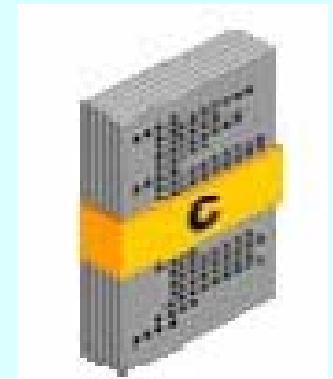
Calling the function with Java:

```
CallableStatement
cstmt = conn.prepareCall("{? = CALL balance(?) }");
cstmt.registerOutParameter(1, Types.FLOAT);
cstmt.setInt(2, acctNo);
cstmt.executeUpdate();
float acctBal = cstmt.getFloat(1);
```

Apa itu OCI?

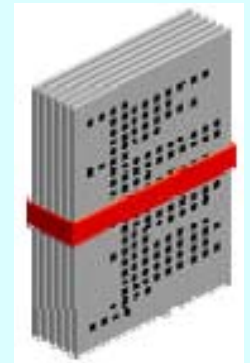
OCI mendukung:

- **Oracle Call Interface (OCI)** mengatur bagaimana agar semua fitur database dapat diakses oleh pembuat aplikasi.
- OCI memungkinkan adanya aplikasi yang berskala dan berperforma tinggi.
- High-level API dan tool-tool menggunakan OCI secara tidak langsung untuk mengakses database.



API Lainnya

- Java Database Connectivity (JDBC)
- Pro*C/C++
- Pro*COBOL
- Pro*Fortran
- Oracle C++ Interface (OCCI)
- Open Database Connectivity (ODBC)
- Oracle Data provider for .Net (ODP .NET)
- Oracle Objects for OLE (OO4O)



Praktikum

1. Konek ke database dengan user HR dengan menggunakan iSQL*Plus.
2. Berikan perintah SQL untuk mendesripsikan table EMPLOYEES, DEPARTMENTS dan LOCATIONS.
3. Tulis query untuk menampilkan last name, phone number dan department id untuk setiap pegawai di table EMPLOYEES.
4. Tulis ulang query sehingga department name ditampilkan sebagai pengganti department_id.
5. Tulis ulang query untuk menampilkan last name, department name dan city untuk setiap pegawai pada table EMPLOYEES.
6. Perhatikan bahwa hasilnya tidak ditampilkan dalam urutan. Tulis ulang query sehingga hasil yang ditampilkanurut secara alfabet pertama berdasarkan city lalu berdasarkan department name dan yang terakhir berdasarkan last name.
7. Tulis perintah SQL yang akan memberikan setiap pegawai di department 80 kenaikan sebesar 10% .
8. Kenaikan 10% yang telah Anda berikan di langkah 7 tidak disetujui oleh manajemen tingakt atas. Batalkan transaksi.
9. Konek ke database sebagai user HR dengan menggunakan SQL*Plus.
10. Tulis query yang menampilkan last name, department name, state dari setiap pegawai di department 30.
11. Buat Implementasi control transaksi meliputi COMMIT, ROLLBACK dan SAVEPOINT.