



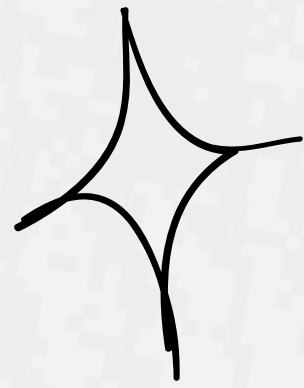
INDIVIDUAL PRESENTATION WORKSHOP

BASIS DATA LANJUT

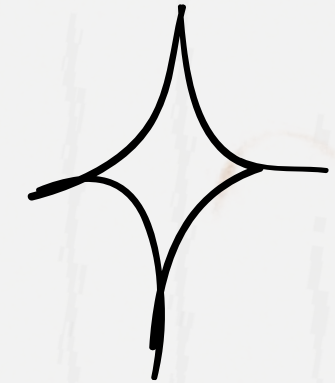


By Rifqy Wafianerdza





Contents



1

- INDEX & VIEW

2

- SEQUENCE & SYNONYM

3

- PL/SQL & PROCEDURE

4

- FUNCTION & CURSOR

5

- EXCEPTION & TRIGGER

Tabel-Tabel Yang Digunakan

1. Tabel Pengaju

Table: PENGAJU ?						
Column	Data Type	Length	Precision	Scale	Nullable	
PENGAJU_ID	NUMBER	22	10	0	No	
PENGAJU_NAMA	VARCHAR2	255	-	-	No	
PENGAJU_EMAIL	VARCHAR2	255	-	-	No	
PENGAJU_JK	VARCHAR2	255	-	-	No	
PENGAJU_PASS	VARCHAR2	255	-	-	No	
PENGAJU_NOTLP	VARCHAR2	20	-	-	No	
PENGAJU_JENIS	VARCHAR2	255	-	-	No	

PENGAJU_ID	PENGAJU_NAMA	PENGAJU_EMAIL	PENGAJU_JK	PENGAJU_PASS	PENGAJU_NOTLP	PENGAJU_JENIS
1	asda	rifqy23ti@pcr.ac.id	Laki-Laki	ac46eb486151ace237901a3ead8fde11	8988667	Dosen
2	tes	tes@pcr.ac.id	Perempuan	28b662d883b6d76fd96e4ddc5e9ba780	085764532	Mahasiswa
3	April	april@pcr.ac.id	Perempuan	f96f2dbc1aae0b2f99a7fe92eeb4ae39	0897654321	Mahasiswa
4	lala	lala@mahasiswa.pcr.ac.id	perempuan	c632a6e0a5238669aa0a4cc8ce56944e	0896432156	mahasiswa
5	dudung	dudung@pcr.ac.id	laki-laki	2ce5a10835c9f1fe4fb1ff13ef2b3331	08923451768	dosen
6	yahya	yahya@pcr.ac.id	laki-laki	b2c00340dcbe97a4e555a3554b514618	08976434161	dosen
7	kola	kola@mahasiswa.pcr.ac.id	perempuan	1d6d6277a96687d1e42d0ce030f4d901	0896571234	mahasiswa
8	abdul	abdul@pcr.ac.id	Laki-laki	550a141f12de6341fba65b0ad0433500	089778658850	mahasiswa
9	ipul	ipul@pcr.ac.id	Laki-laki	15de21c670ae7c3f6f31f37029303c9	089766307712	Dosen
10	Marcopolo	Marc@pcr.ac.id	Laki-laki	f1c1592588411002af340cbaedd6fc33	089766453320	Mahasiswa
11	ww	123	Laki-Laki	202cb962ac59075b964b07152d234b70	213213	Mahasiswa
12	Rifqy	qiqi	Laki-Laki	202cb962ac59075b964b07152d234b70	08123123	Dosen
13	waw	waw@pcr.ac.id	Laki-Laki	202cb962ac59075b964b07152d234b70	081233212	Dosen
14	mbeh	mbeh@pcr.ac.id	Perempuan	4d9375201582469ec8d0ceefd8206720	98123456	Mahasiswa
15	ucok	ucok@pcr.ac.id	Laki-Laki	202cb962ac59075b964b07152d234b70	086788766545	Mahasiswa

Tabel-Tabel Yang Digunakan

2. Tabel Staf

Table: STAF ?					
Column	Data Type	Length	Precision	Scale	Nullable
STAF_ID	NUMBER	22	10	0	No
STAF_NAMA	VARCHAR2	255	-	-	No
STAF_EMAIL	VARCHAR2	255	-	-	No
STAF_PASS	VARCHAR2	255	-	-	No
STAF_JK	VARCHAR2	255	-	-	No
STAF_NOTLP	VARCHAR2	19	-	-	No
1 - 6					

STAF_ID	STAF_NAMA	STAF_EMAIL	STAF_PASS	STAF_JK	STAF_NOTLP
1	Arimby	arimby@pcr.ac.id	e39af9e398b8b92c18daf92ba8e0219d	perempuan	0896543212
2	rifqy	rifqyy@pcr.ac.id	11e54fe99bba3b25cd6e9fdec6185a0	laki-laki	08958764321
3	fikri	fikri@pcr.ac.id	945af650d38ceca4fabd1e2bd7e6b95c	laki-laki	08956432178
4	auliaa	auliaa@pcr.ac.id	5380801648c66572f802b651ac050b13	perempuan	089621764826
5	nada	nada@pcr.ac.id	c31a275423c901d0d1652e5480b9f326	perempuan	089234152678
6	zukri	zukri@pcr.ac.id	3923ff327cbe785dfd4af053b8f080d5	laki-laki	08978654342
7	wulan	wulan@pcr.ac.id	f9b72936ce620fe0188178f7917a5482	perempuan	08987654321
8	mulu	mulu@pcr.ac.id	0d3637d35390b030855f2be0f1d81b9c	laki-laki	08923789651
9	loli	loli@pcr.ac.id	994375777fd2ba1b34cf3485c075b8f3	perempuan	08974536218
10	tomi	tomi@pcr.ac.id	27aaccaacc113b102e5887b14aa8023d	laki-laki	089743982176
11	asep	123@pcr	202cb962ac59075b964b07152d234b70	Laki-Laki	08214321
12	Rahma	rahma@pcr.ac.id	61708596e7bf9a6085e4feaa051335a3	Perempuan	0897654321
13	kiki	kiki@pcr.ac.id	202cb962ac59075b964b07152d234b70	Laki-Laki	09888756786

13 rows returned in 0.04 seconds [Download](#)

3. Tabel Editor

EDITOR_ID	EDITOR_NAMA	EDITOR_EMAIL	EDITOR_PASS	EDITOR_JK	EDITOR_NOTLP	EDITOR_STATUS_PENGERJAAN	STAF_ID
1	tari	tari@pcr.ac.id	5024e30a5f3d993c4ccd197b00c964f9	perempuan	08962134578	Tidak Ada	-
2	bayu	bayu@pcr.ac.id	68276205066ae8264fade33422af399f	laki-laki	08965123478	Ada	13
3	mola	mola@pcr.ac.id	9214e8d196d5c370b31544b028e6983f	perempuan	089671354286	Tidak Ada	-
4	banu	banu@pcr.ac.id	b38f3b8ab934a1f783be94158b628dec	laki-laki	08976895412	Tidak Ada	-
5	tami	tami@pcr.ac.id	5ba0be6cc0da555ff45a49d870fd57b3	perempuan	089231876543	Tidak Ada	-
6	tuti	tuti@pcr.ac.id	e5f8d13d731f580dceb74c4572d3f7a9	perempuan	08963214321	Tidak Ada	-
7	bagus	bagus@pcr.ac.id	2c9a7bf011b0eafee6ee48b2b2b55f9f	laki-laki	08934125678	Tidak Ada	-
8	harto	harto@pcr.ac.id	5195106402898cc44b77302267e6c2e5	laki-laki	0897614253	Tidak Ada	-
9	lana	lana@pcr.ac.id	e4551d969a0edee78507dd433b9c916c	perempuan	089617389276	Ada	11
10	arsel	arsel@pcr.ac.id	32054be821bf15540dc523c61745eaea	laki-laki	08934567893	Tidak Ada	-
10 rows returned in 0.02 seconds Download							

4. Tabel Buku

10 rows returned in 0.01 seconds [Download](#)



Index & View

Index

Sebuah Objek pada database yang berfungsi untuk menyimpan index dari data pada tabel agar proses eksekusi dapat berjalan lebih cepat

Jenis Index:

- Primary Index (dibuat berdasarkan primary key).
- Secondary Index (dibuat secara manual dan tidak menggunakan primary key).

View

Sebuah Objek pada database yang berfungsi sebagai tabel virtual (hanya dapat menampilkan data, tidak untuk menyimpan data) dan dapat di eksekusi

Contoh Index

1. Membuat Index

```
CREATE INDEX buku_judul  
on BUKU(buku_judul)
```

Output

```
Index created.
```

2. Melihat Index

```
SELECT index_name,  
index_type, status,  
table_name FROM  
all_indexes  
WHERE table_name = 'BUKU'
```

Output

INDEX_NAME	INDEX_TYPE	STATUS	TABLE_NAME
SYS_C00162324284	NORMAL	VALID	BUKU
BUKU_JUDUL	NORMAL	VALID	BUKU

Contoh View

1. Membuat View

```
CREATE VIEW buku_judul_c AS  
SELECT buku_judul FROM BUKU  
ORDER BY buku_judul asc
```

Output

View created.

2. Mengeksekusi View

```
SELECT * FROM  
buku_judul_c
```

Output

BUKU_JUDUL
Cahaya Kampus
Dampak nilai
Gelarku
KOKI
Kita dan Kota
Kutemukan



Sequence & Synonym

Sequence

Objek yang menghasilkan nilai numerik yang unik dan berurutan. Objek ini digunakan untuk menghasilkan nilai otomatis bagi kolom yang memerlukan nilai unik, misalnya kolom ID dalam tabel.

Synonym

Digunakan untuk penamaan alternatif pada objek seperti tabel, view, sequence, prosedur, dan lainnya. Synonym dapat bersifat public dan private. Synonym public dapat diakses oleh semua pengguna, sementara synonym private hanya dapat diakses oleh pemiliknya.

Contoh Sequence

1. Membuat Sequence

```
CREATE SEQUENCE generateId  
  INCREMENT BY 1  
  START WITH 100  
  MAXVALUE 150  
  CYCLE  
  NOCACHE
```

Output

```
Sequence created.
```

2. Menghapus Sequence

```
DROP SEQUENCE generateId
```

Output

```
Sequence dropped.
```


Contoh Sequence

3. Menggunakan Sequence Untuk Insert Data

```
INSERT INTO BUKU  
VALUES(generateId.nextVal, 'Bagaikan Matahari', '2013', 'Nunung',  
'Menunggu Diproses', 1, 2, 3, 'https://google.com')
```

Output

```
1 row(s) inserted.
```

```
SELECT * FROM buku where buku_judul = 'Bagaikan Matahari'
```

BUKU_ID	BUKU_JUDUL	BUKU_TAHUN	BUKU_PENULIS	BUKU_STATUS	PENGAJU_ID	STAF_ID	EDITOR_ID	BUKU_DOKUMEN
100	Bagaikan Matahari	2013	Nunung	Menunggu Diproses	1	2	3	https://google.com

Contoh Synonym

1. Membuat Synonym

```
CREATE SYNONYM books  
FOR buku
```

Output

```
Statement processed.
```

2. Melihat Synonym

```
SELECT * FROM books
```

Output

ID	BUKU_ID	BUKU_JUDUL	BUKU_TAHUN	BUKU_PENULIS	BUKU_STATUS	PENGAJUAN
1	1	Cahaya Kampus	2022	Aulia	Sedang Diproses	1
2	2	Wahai	2023	kii	Sedang Diproses	1
3	3	Kita dan Kota	2023	april lia	Menunggu Diproses	3
4	4	Kutemukan	2021	sani	Sedang Diajukan	4



PL/SQL & Procedure

PL/SQL

Penggabungan antara bahasa pemrograman prosedural (PL) dan sintaks Structured Query Language (SQL). Jika SQL berfokus pada bagaimana mendapatkan hasil dari database, PL/SQL menjalankan proses dan alur logika seperti kondisi dan perulangan

Procedure

Merupakan subprogram atau bagian dari program yang menjalankan satu atau lebih tugas tertentu. Sebuah procedure dapat dibuat pada tingkat skema (Standalone), dalam blok PL/SQL, atau sebagai bagian dari package. Struktur: Declarative Part (Deklarasi), Executable Part (Bagian yang dieksekusi), dan Exception Handling (Menangkap Error).

Contoh PL/SQL

1. Membuat Variabel

Output

```
DECLARE
  i buku.buku_judul%TYPE;
BEGIN
  i := 'Cahaya Kampus';
  dbms_output.put_line('Buku ini berjudul : '||i);
END
```

Buku ini berjudul : Cahaya Kampus

2. Melakukan While - Looping

Output

```
DECLARE
  i NUMBER := 1;
BEGIN
  WHILE i <= 5 LOOP
    dbms_output.put_line(i);
    I := i+1;
  END LOOP;
END;
```

1
2
3
4
5

Contoh Procedure

1. Membuat Procedure Standalone

Output

```
CREATE PROCEDURE perpustakaan  
AS  
BEGIN  
    dbms_output.put_line('Selamat datang di perpustakaan');  
END;
```

Procedure created.

2. Menjalankan Procedure Standalone

Output

```
BEGIN  
    perpustakaan;  
END;
```

Selamat datang di perpustakaan



Function & Cursor

Function

Sama halnya dengan procedure, function digunakan sebagai bagian atau subproses dari sebuah program PL/SQL. Bedanya adalah function mengembalikan nilai, procedure tidak.

Cursor

Digunakan sebagai penunjuk area konteks pada program PL/SQL yang menandai baris yang dieksekusi oleh SQL. Jenis:

- Implisit Cursor (Cursor langsung pada PL/SQL)
- Eksplisit Cursor (Cursor yang ditulis diluar PL/SQL)

Contoh Function

1. Membuat Function

```
CREATE FUNCTION welcome  
|   RETURN VARCHAR2  
AS BEGIN  
|   RETURN('Selamat Datang di Indomaret!');  
END;
```

Output

Function created.

0.04 seconds

2. Menjalankan Function

```
BEGIN  
|   dbms_output.put_line(welcome);  
END;
```

Output

Selamat Datang di Indomaret!

Statement processed.

0.01 seconds

Contoh Cursor

1. Contoh Penggunaan Cursor Eksplisit

```
DECLARE
  v_buku_judul buku.buku_judul%TYPE;
  v_buku_status buku.buku_status%TYPE;
  CURSOR c_buku IS
    SELECT buku_judul, buku_status
    FROM buku
    WHERE buku_status = 'Selesai';
BEGIN
  OPEN c_buku;
  LOOP
    FETCH c_buku INTO v_buku_judul, v_buku_status;
    EXIT WHEN c_buku%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('Judul: ' || v_buku_judul || ', Status: ' || v_buku_status);
  END LOOP;
  CLOSE c_buku;
END;
```

Output

```
Judul: Cahaya Kampus, Status: Selesai
Judul: Wahai, Status: Selesai
Judul: Kutemukan, Status: Selesai
Judul: Tas Merah, Status: Selesai
Judul: KOKI, Status: Selesai
Judul: Loncengan, Status: Selesai
Judul: Gelarku, Status: Selesai
Judul: Dampak nilai, Status: Selesai
```


Contoh Cursor

2. Contoh Penggunaan Cursor Implisit

```
DECLARE
  jumBaris number (2);
BEGIN
  UPDATE buku SET buku_judul = 'Dikala Malam'
  WHERE buku_judul = 'asdasd';
  if sql%notfound then
    DBMS_OUTPUT.PUT_LINE('Tidak ada Data');
  elsif sql%found then
    jumBaris := sql%rowcount;
    DBMS_OUTPUT.PUT_LINE('Data sebanyak ' || jumBaris || ' berhasil diubah!');
  end if;
END;
```

Output

Tidak ada Data

1 row(s) updated.



Trigger & Exception Handling

Trigger

Merupakan proses yang akan dieksekusi ketika terjadi suatu perubahan atau event. Event yang tersebut adalah Query DML, DDL, System Event dan User Event

Exception

Merupakan kondisi kesalahan yang terjadi saat PL/SQL dieksekusi. Terbagi menjadi 2:

- System Defined Exception
- User Defined Exception

Contoh Trigger

1. Membuat Trigger

```
CREATE TRIGGER tINSERTBuku
AFTER INSERT ON buku
FOR EACH ROW
BEGIN
    dbms_output.put_line('Buku baru dengan id: '
        || :NEW.buku_id || ' telah dimasukkan!');
END;
```

Output

Trigger created.

0.05 seconds

2. Memicu Trigger Saat Insert Data Buku

```
INSERT INTO BUKU VALUES(18, 'Bagaikan Bulan', '2014', 'Nanang',
    'Menunggu Diproses', 2, 4, 3, 'https://google.com')
```

Output

Buku baru dengan id: 18 telah dimasukkan!

1 row(s) inserted.

Contoh Exception

1. Contoh Penggunaan Exception

```
DECLARE
    v_buku_id buku.buku_id%type := 18;
    v_buku_judul buku.buku_judul%type;
BEGIN
    SELECT buku_id, buku_judul
    INTO v_buku_id, v_buku_judul
    FROM buku WHERE buku_id = v_buku_id;
    dbms_output.put_line('ID Buku: ' || v_buku_id);
    dbms_output.put_line('Judul Buku: ' || v_buku_judul);
EXCEPTION
    WHEN no_data_found THEN
        dbms_output.put_line('Tidak ada Buku dengan ID 18!');
    WHEN others THEN
        dbms_output.put_line('Terjadi Error!');
END
```

Output

```
Tidak ada Buku dengan ID 18!
Statement processed.
```


Thanks!

