

## **Laporan Praktikum Basis Data Lanjut**



Dosen Pengampu: Muhammad Naufal Luthfi, S.T., M.Kom

AIL: Rian Ardi Wibowo, S.Tr.Kom

Identitas Mahasiswa:

Nama Lengkap: Nayla Saffana Chalisa

NIM: 2457301112

Kelas: 2 SIB

Pertemuan ke - 13

Hari/Tanggal: Rabu, 3 Desember 2025

Program Studi Sistem Informasi Politeknik Caltex Riau

No	Perintah SQL	Hasil
1	<pre> CREATE OR REPLACE TRIGGER tShowDataAgents BEFORE INSERT ON agents FOR EACH ROW BEGIN dbms_output.put_line('Agent Code: '    :NEW.AGENT_CODE); dbms_output.put_line('Agent Name: '    :NEW.AGENT_NAME); dbms_output.put_line('COMISSION : '    :NEW.COMISSION); END; </pre>	<p>Error at line 7: PLS-00049: bad bind variable 'NEW.COMISSION'</p> <pre> 5. dbms_output.put_line('Agent Code: '    6. :NEW.AGENT_CODE); 7. dbms_output.put_line('Agent Name: '    8. :NEW.AGENT_NAME); 9. dbms_output.put_line('COMISSION : '    </pre> <p>Trigger dibuat untuk menampilkan informasi agents ke output sebelum data baru di insert ke tabel agents.</p> <p>Perbaikan:</p> <pre> 12 -- perbaikan 13 CREATE OR REPLACE TRIGGER tShowDataAgents 14 BEFORE INSERT ON agents 15 FOR EACH ROW 16 BEGIN 17 dbms_output.put_line('Agent Code: '    18 :NEW.AGENT_CODE); 19 dbms_output.put_line('Agent Name: '    20 :NEW.AGENT_NAME); 21 dbms_output.put_line('COMISSION : '    22 :NEW.COMISSION); 23 END; </pre> <p>Results Explain Describe Saved SQL History</p> <p>Trigger created. 0.09 seconds</p>
2	<pre> CREATE OR REPLACE TRIGGER tINSCust AFTER INSERT ON customer FOR EACH ROW BEGIN DBMS_OUTPUT.PUT_LINE('Dari Trigger tINSCust'); DBMS_OUTPUT.PUT_LINE('Cust Code : '  :NEW.CUST_CODE); DBMS_OUTPUT.PUT_LINE('Cust NAME : '  :NEW.CUST_NAME); END; </pre>	<p>Trigger created.</p> <p>0.07 seconds</p> <p>Insert dengan sebuah data pada table customer:</p> <pre> 36 --Insert sebuah data pada table customer 37 CREATE OR REPLACE TRIGGER tINSCust 38 AFTER INSERT ON customer 39 FOR EACH ROW 40 BEGIN 41 DBMS_OUTPUT.PUT_LINE('Dari Trigger tINSCust'); 42 DBMS_OUTPUT.PUT_LINE('Cust Code : '    :NEW.CUST_CODE); 43 DBMS_OUTPUT.PUT_LINE('Cust NAME : '    :NEW.CUST_NAME); 44 END; </pre> <p>Results Explain Describe Saved SQL History</p> <p>Trigger created.</p>
3	<pre> CREATE OR REPLACE TRIGGER tDELCust BEFORE DELETE ON customer FOR EACH ROW BEGIN DELETE FROM ORDERS WHERE CUST_CODE = :OLD.CUST_CODE; END; </pre>	<p>Trigger created.</p> <p>0.06 seconds</p>

4	<pre>CREATE TABLE histCustomer( idhistC number primary key, tanggal timestamp, cust_code varchar2(6), cust_name varchar2(40), cust_city char(35), working_area varchar2(35), cust_country varchar2(50), grade number, opening_amt number, receive_amt number, payment_amt number, outstanding_amt number, phone_no varchar2(17), agent_code char(6) );</pre>	<p><b>Table created.</b></p> <p><b>0.07 seconds</b></p> <p>Tujuan: untuk membuat table histCustomer yang berfungsi sebagai tabel histori untuk menyimpan riwayat perubahan data customer.</p>
5	<pre>CREATE SEQUENCE idHisDelCu_seq START WITH 1;</pre>	<p><b>Sequence created.</b></p> <p><b>0.02 seconds</b></p>
6	<pre>CREATE OR REPLACE TRIGGER tDELCust BEFORE DELETE ON customer FOR EACH ROW BEGIN INSERT into histCustomer values( idHisDelCu_seq.NEXTVAL, localtimestamp, :OLD.CUST_CODE, :OLD.CUST_NAME, :OLD.CUST_CITY, :OLD.WORKING_AREA, :OLD.CUST_COUNTRY, :OLD.GRADE, :OLD.OPENING_AMT, :OLD.RECEIVE_AMT, :OLD.PAYMENT_AMT, :OLD.OUTSTANDING_AMT, :OLD.PHONE_NO, :OLD.AGENT_CODE ); END;</pre>	<p><b>Trigger created.</b></p> <p><b>0.07 seconds</b></p>

7	DELETE FROM CUSTOMER WHERE CUST_CODE = 'C00100';	<pre>0 row(s) deleted.</pre> <p>0.04 seconds</p> <p>Hasil yang terjadi pada tabel customer:</p>
8	CREATE TABLE histOrders(     idhistO number,     tanggal timestamp,     order_num decimal(6,0),     ord_amount decimal (12,2),     advance_amount     decimal(12,2),     ord_date date,     cust_code varchar(6),     agent_code char(6),     ord_description varchar(60),     constraint pk_histOrder     primary     key(idhistO) );	<pre>Table created.</pre> <p>0.09 seconds</p> <p>Perbedaan antara query no 4 dan no 8:  Tabel histCustomer dibuat untuk mencatat riwayat perubahan data pelanggan/customer, sedangkan tabel histOrders untuk mencatat histori pesanan. histCustomer mencatat semua atribut customer, dan histOrders lebih fokus pada data tansaksi pesanan saja.</p>
9	CREATE SEQUENCE idHisDelOr_seq START WITH 1;	<pre>Sequence created.</pre> <p>0.03 seconds</p>
10	CREATE OR REPLACE TRIGGER tDELOrder BEFORE DELETE ON orders FOR EACH ROW BEGIN INSERT into histOrders values ( idHisDelOr_seq.NEXTVAL, localtimestamp, :OLD.ORD_NUM, :OLD.ORD_AMOUNT, :OLD.ADVANCE_AMOUNT, :OLD.ORD_DATE, :OLD.CUST_CODE,	<pre>Trigger created.</pre> <p>0.08 seconds</p>

	<pre>:OLD.AGENT_CODE, :OLD.ORD_DESCRIPTION ); END;</pre>	
11	<pre>DELETE FROM orders WHERE ORD_NUM='200100';</pre>	<pre>1 row(s) deleted.  0.08 seconds</pre> <p>Table orders berfungsi sebagai sistem transaksi aktif yang ringan dan cepat untuk operasional, sementara histOrders berfungsi sebagai sistem arsip yang menyimpan setiap perubahan order secara lengkap dengan timestamp. Jika sebuah pesanan di edit di orders, maka akan tampil nilai baru, sedangkan di tabel histOrders akan tercatat data lama beserta waktu kapan perubahan dilakukan.</p>
12	<pre>CREATE OR REPLACE TRIGGER tDELcust BEFORE DELETE ON customer FOR EACH ROW BEGIN INSERT into histCustomer values ( idHisDelCu_seq.NEXTVAL, localtimestamp, :OLD.CUST_CODE, :OLD.CUST_NAME, :OLD.CUST_CITY, :OLD.WORKING_AREA, :OLD.CUST_COUNTRY, :OLD.GRADE, :OLD.OPENING_AMT, :OLD.RECEIVE_AMT, :OLD.PAYMENT_AMT, :OLD.OUTSTANDING_AMT, :OLD.PHONE_NO, :OLD.AGENT_CODE ); DELETE FROM orders WHERE CUST_CODE =</pre>	<pre>Trigger created.  0.05 seconds</pre>

	<pre>:OLD.CUST_CODE; END;</pre>	
13	<pre>DELETE FROM customer WHERE CUST_CODE='C00007';</pre>	<p>1 row(s) deleted.</p> <p>0.13 seconds</p> <p>Hasil yang terjadi pada tabel:  Tabel customer dan orders adalah tabel operasional, customer menyimpan informasi semua pelanggan, dan orders berisi semua pesanan aktif.  Tabel histCustomer dan histOrders adalah sistem pencatatan otomatis yang setiap kali ada perubahan di tabel customer, sistem langsung menyimpan data lamanya ke histCustomer. Ketika data pelanggan di update di tabel customer, maka di tabel customer akan menampilkan alamat barunya, sedangkan di histCustomer akan tersimpan record baru dengan alamat lama dan waktu kapan perubahan dilakukan, begitu juga di tabel orders dan histOrders.</p>
14	<pre>CREATE OR REPLACE TRIGGER t BEFORE INSERT OR UPDATE OF AGENT_CODE, AGENT_NAME OR DELETE ON AGENTS FOR EACH ROW BEGIN CASE WHEN INSERTING THEN DBMS_OUTPUT.PUT_LINE('Input Data Pada Tabel AGENTS'); WHEN UPDATING('AGENT_CODE') THEN DBMS_OUTPUT.PUT_LINE('Peruba han Data Pada Kolom AGENT_CODE dari Tabel AGENTS'); WHEN UPDATING('AGENT_NAME')</pre>	<p>Trigger created.</p> <p>0.07 seconds</p> <p>Fungsi utama dari trigger ini adalah untuk memberikan transparansi dan konfirmasi real time untuk semua operasi perubahan data pada tabel agents, sehingga pengguna tau apa yang terjadi dengan dta mereka setiap saat.</p>

```

187 THEN
DBMS_OUTPUT.PUT_LINE('Perubahan Data
Pada Kolom AGENT_NAME :
'||:OLD.AGENT_NAME ||
menjadi :
'||:NEW.AGENT_NAME);
188 WHEN DELETING THEN
DBMS_OUTPUT.PUT_LINE('Penghapusan Data');
189 END CASE;
190 END;

```

### Tugas praktikum

- Buatlah sebuah table untuk menampung data yang dihapus dari table agents!

```

188 CREATE TABLE histAgents (
189   idhist    NUMBER PRIMARY KEY,
190   tanggal   TIMESTAMP,
191   agent_code CHAR(6),
192   agent_name VARCHAR2(40),
193   working_area VARCHAR2(35),
194   commission NUMBER(10,2),
195   phone_no   VARCHAR2(15),
196   country    VARCHAR2(25)
197 );

```

Results Explain Describe Saved SQL History

Table created.

0.05 seconds

```

198
199  CREATE SEQUENCE idHisDelAg_seq START WITH 1 INCREMENT BY 1;

```

Results Explain Describe Saved SQL History

Sequence created.

0.02 seconds

Table: HISTAGENTS					
Column	Data Type	Length	Precision	Scale	Nullable
IDHIST	NUMBER	22	-	-	No
TANGGAL	TIMESTAMP(6)	11	-	6	Yes
AGENT_CODE	CHAR	6	-	-	Yes
AGENT_NAME	VARCHAR2	40	-	-	Yes
WORKING_AREA	VARCHAR2	35	-	-	Yes
COMMISSION	NUMBER	22	10	2	Yes
PHONE_NO	VARCHAR2	15	-	-	Yes
COUNTRY	VARCHAR2	25	-	-	Yes

2. Buatlah sebuah trigger yang akan memindahkan data yang dihapus dari table agents ke table no-1! (Perhatikan relasi dari table agents)

```

202 -- Membuat Trigger tDELAagents
203 CREATE OR REPLACE TRIGGER tDELAagents
204 BEFORE DELETE ON agents
205 FOR EACH ROW
206 BEGIN
207   INSERT into histAgents values (
208     idhisDelAg_seq.NEXTVAL,
209     localtimestamp,
210     :OLD.AGENT_CODE,
211     :OLD.AGENT_NAME,
212     :OLD.WORKING_AREA,
213     :OLD.COMMISSION,
214     :OLD.PHONE_NO,
215     :OLD.COUNTRY
216   );
217 END;
218

```

**Results** Explain Describe Saved SQL History

Trigger created.

0.06 seconds

228 SELECT \* FROM agents ORDER BY agent\_code;

**Results** Explain Describe Saved SQL History

AGENT_CODE	AGENT_NAME	WORKING_AREA	COMMISSION	PHONE_NO	COUNTRY
1234	Test	Test	.75	0000000	Test
A001	Subbarao	Bangalore	.26	077-12346674	INDIA
A002	Mukeshi	Mumbai	.21	029-12358964	INDIA
A003	Alex	London	.23	075-12458969	ENGLAND
A004	Ivan	Toronto	.27	008-22544166	CANADA
A005	Anderson	Brisban	.25	045-2147739	AUSTRALIA
A006	McDen	London	.27	078-22255988	ENGLAND
A007	Ramasundar	Bangalore	.28	077-25814763	INDIA
A008	Alford	New York	.22	044-25874365	USA
A009	Benjamin	Hampshair	.2	008-22536178	ENGLAND