



PL/SQL Advanced(project)

Aya Sabry

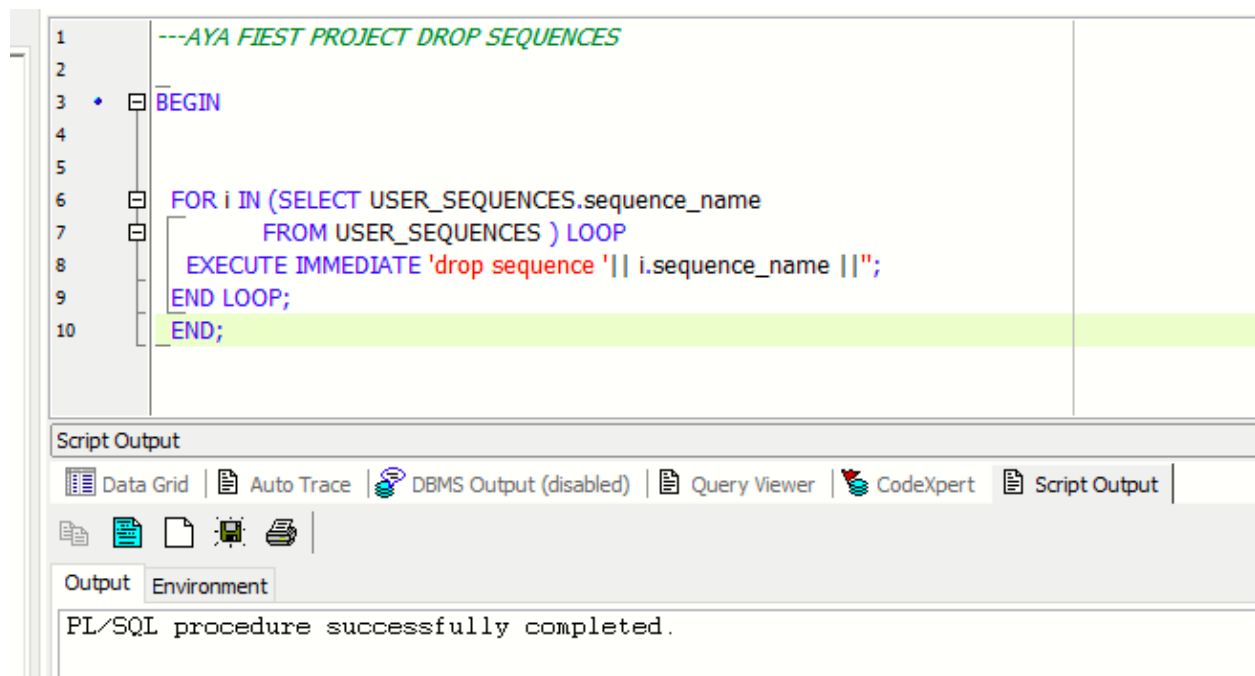
Submitted to:

Eng. Yehia Momtaz

Jan 8, 2023

Task

1-Drop sequences



The screenshot displays the Oracle SQL Developer environment. The main editor window contains a PL/SQL script with the following code:

```
1  ---AYA FIEST PROJECT DROP SEQUENCES
2
3  BEGIN
4
5
6  FOR i IN (SELECT USER_SEQUENCES.sequence_name
7            FROM USER_SEQUENCES ) LOOP
8      EXECUTE IMMEDIATE 'drop sequence ' || i.sequence_name || ";
9  END LOOP;
10 END;
```

The script is executed, and the "Script Output" pane at the bottom shows the message: "PL/SQL procedure successfully completed." The interface includes a toolbar with icons for Data Grid, Auto Trace, DBMS Output (disabled), Query Viewer, CodeXpert, and Script Output. The "Output" tab is currently selected, showing the completion message.

2-

```
set serveroutput on
```

```
-- AYA FIRST PROJECT--
```

```
declare
```

```
    curr_mx number;  
    counter number:=1;  
    sequ_name varchar2 (30);  
    trigger_nam varchar2 (30);
```

```
begin
```

```
    for rec in ( select col_cons.table_name  
                  , col_cons.column_name  
                from user_constraints uc  
                join user_cons_columns col_cons  
                  on col_cons.table_name = uc.table_name  
                  and col_cons.constraint_name = uc.constraint_name  
                join user_tab_columns tc  
                  on tc.table_name = col_cons.table_name  
                  and tc.column_name = col_cons.column_name  
                where uc.constraint_type = 'P'  
                  and tc.data_type = 'NUMBER'  
                )
```

```
loop
```

```
    execute immediate 'Select (max ('|| rec.column_name ||')+1) from ' ||rec.table_name  
    into curr_mx;
```

```
    IF curr_mx is null THEN
```

```
        curr_mx := 0;
```

```
    END IF;
```

```
        sequ_name := counter || '_PKAAA_SEQ';
```

```
        trigger_nam := counter || '_PKA_Trig';
```

```
    counter := counter +1;
```

```
    execute immediate 'CREATE SEQUENCE "' || sequ_name || '"  
    || ' START WITH ' || to_char( curr_mx + 1 ) ||";
```

```
    execute immediate  'CREATE OR REPLACE TRIGGER "' || trigger_nam || '"  
    || ' BEFORE INSERT ON "' || rec.table_name || '"  
    || ' FOR EACH ROW '  
    || ' BEGIN '  
    || ' :new."' || rec.column_name || '" := "' || sequ_name || '".nextval;'  
    || ' END;;
```

```
end loop;
```

end;

3-Show sequences and triggers after run project

The screenshot shows a database IDE interface. The top pane displays a SQL script with the following content:

```
49  
50  
51  
52     end loop;  
53 end;  
54 -----Aya sabry  
55 ► select 'select sequence ' || sequence_name || ';' as sequence_project from user_sequences;
```

The bottom pane, titled "Data Grid", shows the results of the query. It includes a toolbar with icons for "Data Grid", "Auto Trace", "DBMS Output", "Query Viewer", "CodeXpert", and "Script Output". Below the toolbar is a "Cancel" button. The results are displayed in a table with the following data:

SEQUENCE_PROJECT
select sequence 10_PKAAA_SEQ;
select sequence 11_PKAAA_SEQ;
select sequence 12_PKAAA_SEQ;
select sequence 13_PKAAA_SEQ;
select sequence 14_PKAAA_SEQ;
select sequence 15_PKAAA_SEQ;
select sequence 1_PKAAA_SEQ;
select sequence 2_PKAAA_SEQ;
select sequence 3_PKAAA_SEQ;

The bottom of the interface features a standard SQL IDE toolbar with navigation and execution icons.

•	SELECT	
	trigger_name,	
	trigger_type,	
	triggering_event,	
	table_owner,	
	table_name,	
	base_object_type,	
	status,	
	trigger_body	
0	FROM	
1	user_triggers;	
2		

Script Output

Data Grid Auto Trace DBMS Output Query Viewer CodeXpert Script Output

Output Grid 1 Environment

TRIGGER_NAME	TRIGGER_TYPE	TRIGGERING_EVENT	TABLE_OWNER	TABLE_NAME	BASE_OBJECT_TYPE	STATUS	TRIGGER_BODY
13_PKA_Trig	BEFORE EACH ROW	INSERT	HR	TRAINERS	TABLE	ENABLED	BEGIN :new."TR_ID" := "13_PKAAA_SEQ".nextval; END;
12_PKA_Trig	BEFORE EACH ROW	INSERT	HR	REGIONS	TABLE	ENABLED	BEGIN :new."REGION_ID" :=

Output

Grid 1

Environment

TRIGGER_NAME	TRIGGER_TYPE	TRIGGERING_EVENT	TABLE_OWNER	TABLE_NAME	BASE_OBJECT_TYPE	STATUS	TRIGGER_BODY
13_PKA_Trig	BEFORE EACH ROW	INSERT	HR	TRAINERS	TABLE	ENABLED	BEGIN :new."TR_ID" := "13_PKAAA_SEQ".nextval; END;
12_PKA_Trig	BEFORE EACH ROW	INSERT	HR	REGIONS	TABLE	ENABLED	BEGIN :new."REGION_ID" := "12_PKAAA_SEQ".nextval;
11_PKA_Trig	BEFORE EACH ROW	INSERT	HR	LOCATIONS	TABLE	ENABLED	BEGIN :new."LOCATION_ID" := "11_PKAAA_SEQ".nextval
10_PKA_Trig	BEFORE EACH ROW	INSERT	HR	JOB_HISTORY	TABLE	ENABLED	BEGIN :new."EMPLOYEE_ID" := "10_PKAAA_SEQ".nextval
9_PKA_Trig	BEFORE EACH ROW	INSERT	HR	INSTALLMENTS_PAID	TABLE	ENABLED	BEGIN :new."INSTALLMENT_ID" := "9_PKAAA_SEQ".nextv
8_PKA_Trig	BEFORE EACH ROW	INSERT	HR	EMPS2	TABLE	ENABLED	BEGIN :new."EMP_ID" := "8 PKAAA SFO".nextval; END;

4-TEST BY INSERT NEW DEPARTMENT

1 *----AYA SABRY PROJECT*
2 • `select max(department_id) from DEPARTMENTS;`
3
4
5
6
7
8

Data Grid

Data Grid | Auto Trace | DBMS Output | Query Viewer | CodeXpe

Cancel

MAX(DEPARTMENT_ID)
893

1 *----AYA SABRY PROJECT*
2 • `select * from departments where department_id in (select max(department_id) from DEPARTMENTS);`
3
4
5
6
7
8

Data Grid

Data Grid | Auto Trace | DBMS Output | Query Viewer | CodeXpert | Script Output

Cancel

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
893	AI_robot1		3600

<No name>

<No name>

<No name>

<No name>

<No name>

<No name>

1

2

3

4

5

6

7

8

9

10

11

AYA SABRY PROJECT

•

select * from departments where department_id in (select max(department_id) from DEPARTMENTS);

AYA -TEST PROJECT

•

insert into departments(department_name,location_id)

values('AI_compterVision',1800);

▶

select * from departments where department_id in (select max(department_id) from DEPARTMENTS);

Data Grid

Auto Trace

DBMS Output

Query Viewer

CodeXpert

Script Output

Cancel

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
894	AI_compterVision		1800