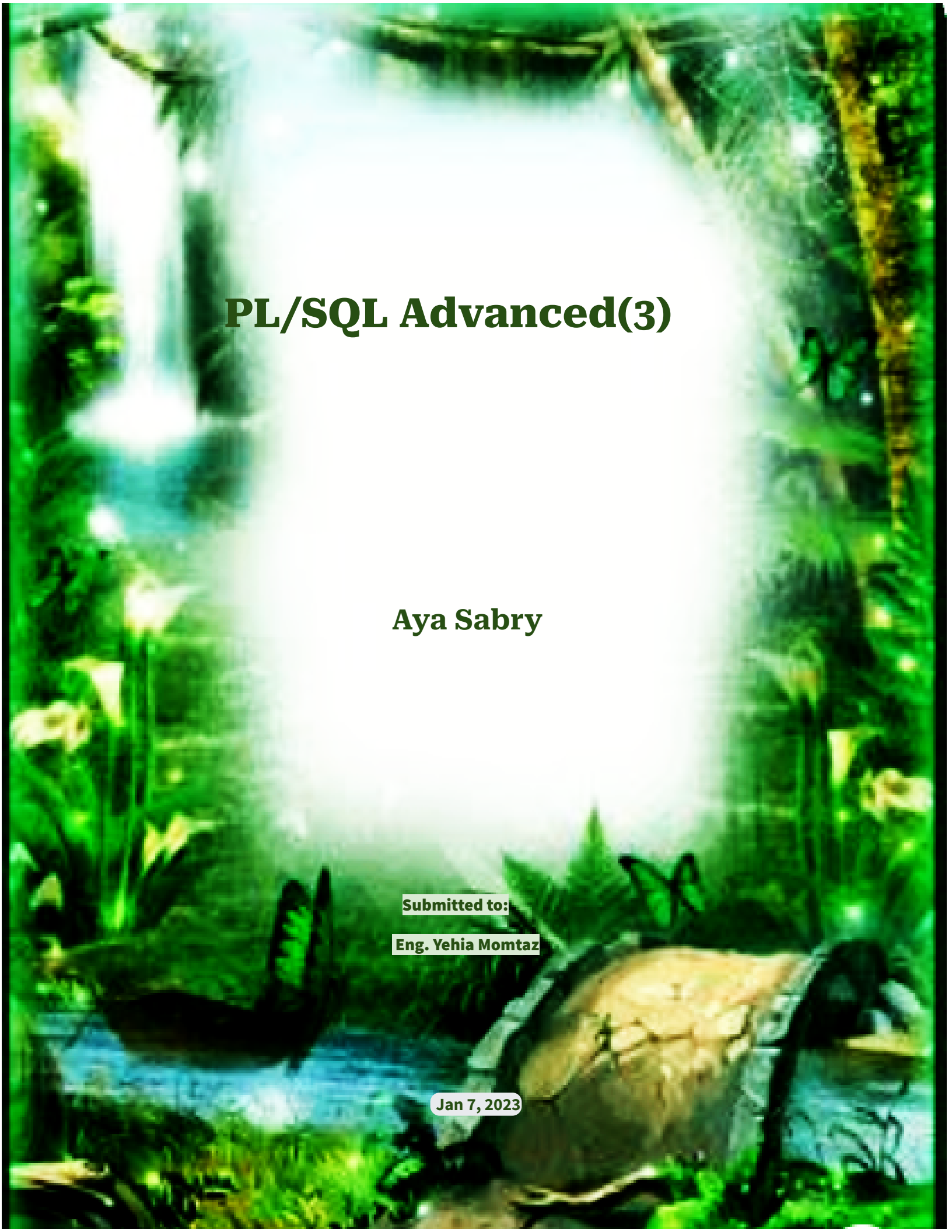


PL/SQL Advanced(3)

Aya Sabry

Submitted to:

Eng. Yehia Momtaz

Jan 7, 2023

Task 3

Database Triggers Lab

1. **Create trigger to audit the user updates in the employees for only salary column**
   Create new Table emp_audit
   With columns
   employee_id number(4), user_name varchar2(100), upd_time date , old_sal number(8, 2) , new_sal number(8, 2)

```
set serveroutput on

---Aya lab 3 p(1)



CREATE TABLE emp_audit
(
employee_id number(4), user_name varchar2(100), upd_time date , old_sal number(8, 2) , new_sal number(8, 2)

);
```

```
set serveroutput on

---Aya lab 3 p(1)


create or replace trigger emp_audit_trig
after update of salary on employees
for each row

begin
insert into emp_audit (employee_id , user_name, upd_time, old_sal, new_sal)
values (:old.employee_id, user, sysdate, :old.salary,:new.salary);
end;
```

Script Output

Data Grid | Auto Trace | DBMS Output (disabled) | Query Viewer | CodeXpert | Script Outp

Output | Environment

```
Trigger created.
```

2. **The rows in the JOBS table store a minimum and maximum salary allowed for different JOB_ID values. You are asked to write code to ensure that employees' salaries fall in the range allowed for their job type**.

```
set serveroutput on

---Aya lab 3 p(2)
create or replace trigger audit_salary
before update or insert of salary on employees
for each row
declare
v_min_salary employees.salary%type;
v_max_salary employees.salary%type;
v_new_salary employees.salary%type;
begin

v_new_salary := :new.salary;
select min_salary, max_salary
into v_min_salary,v_max_salary
from jobs
where job_id = :new.job_id;
if v_new_salary not between  v_min_salary and  v_max_salary then
raise_application_error(-20020,'Stop this process The salary is out of range');
end if;
end;
```

Script Output

Data Grid | Auto Trace | DBMS Output (disabled) | Query Viewer | CodeXpert | Script Output

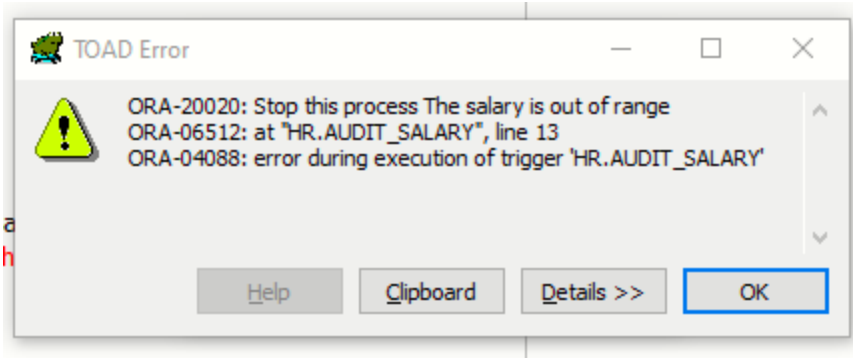
Output | Environment

```
Trigger created.
```

Example

update employees
set salary =55000

where employee_id=105;



3. **Create instead of trigger on view "emps_vu" to update** control_jobs_sal **table and store Min/Max Salary for the jobs after each insert / update on the view.[change jobs scale]**

Use this code for the view

```
create or replace view emps_vu
as select * from employees;
```

create table

```
create table control_jobs_sal
as select job_id, max_salary, min_salary
from jobs;
```

```
---Aya lab 3 p(3)
create or replace trigger emps_vu_trig
instead of insert or update on emps_vu
for each row
declare
vnewsalary employees.salary%type;vminsalary employees.salary%type;vmaxsalary employees.salary%type;
begin
vnewsalary:= :new.salary;
select max_salary, min_salary
into vmaxsalary,vminsalary
from control_jobs_sal
where job_id= :new.job_id;
if vnewsalary< vminsalary then
   update control_jobs_sal
   set min_salary=vnewsalary
   where job_id=:new.job_id;
elsif vnewsalary>vmaxsalary then
   update control_jobs_sal
   set max_salary=vnewsalary
   where job_id=:new.job_id;
end if;
end;
```

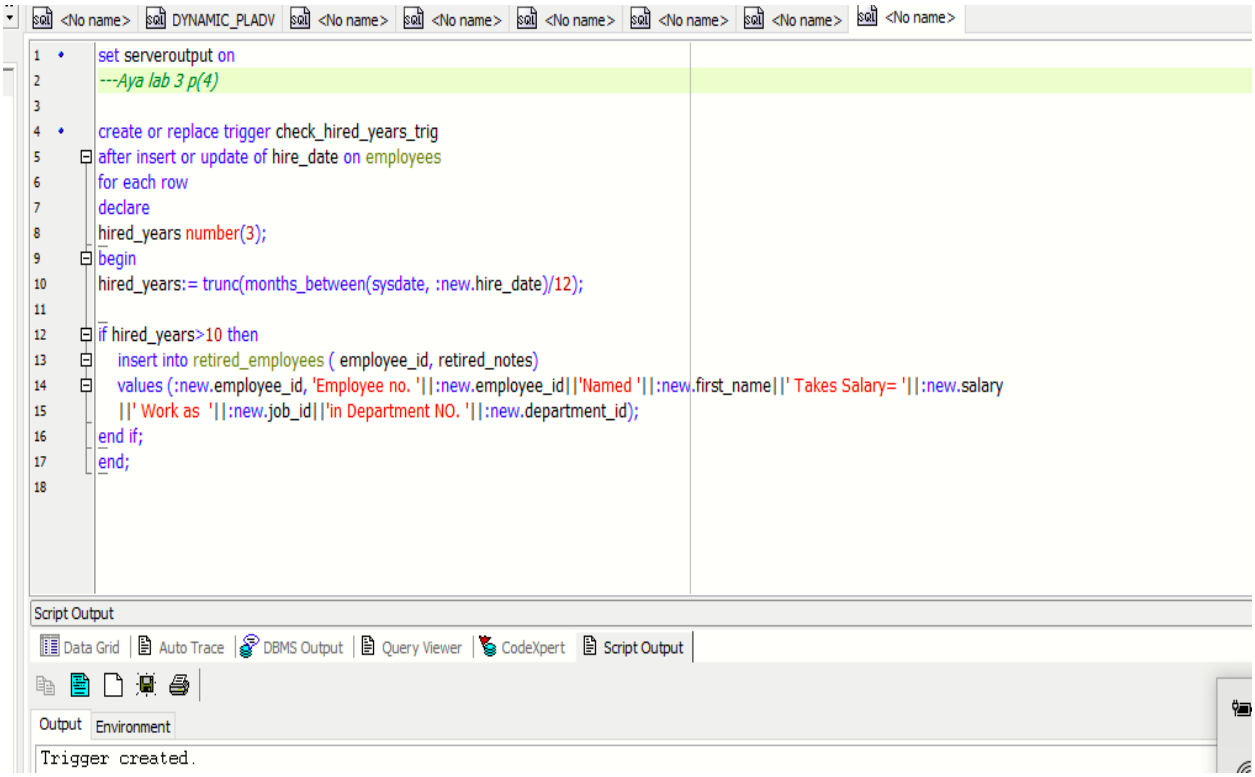
Script Output

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

Output | Environment

```
Trigger created.
```

4. **Create trigger on employees table that run upon insert, or update of hire_date and check if the hired years extends 10 years; insert employee data to retired_employees table (emp_id, retired_notes) columns only as the previous lab**

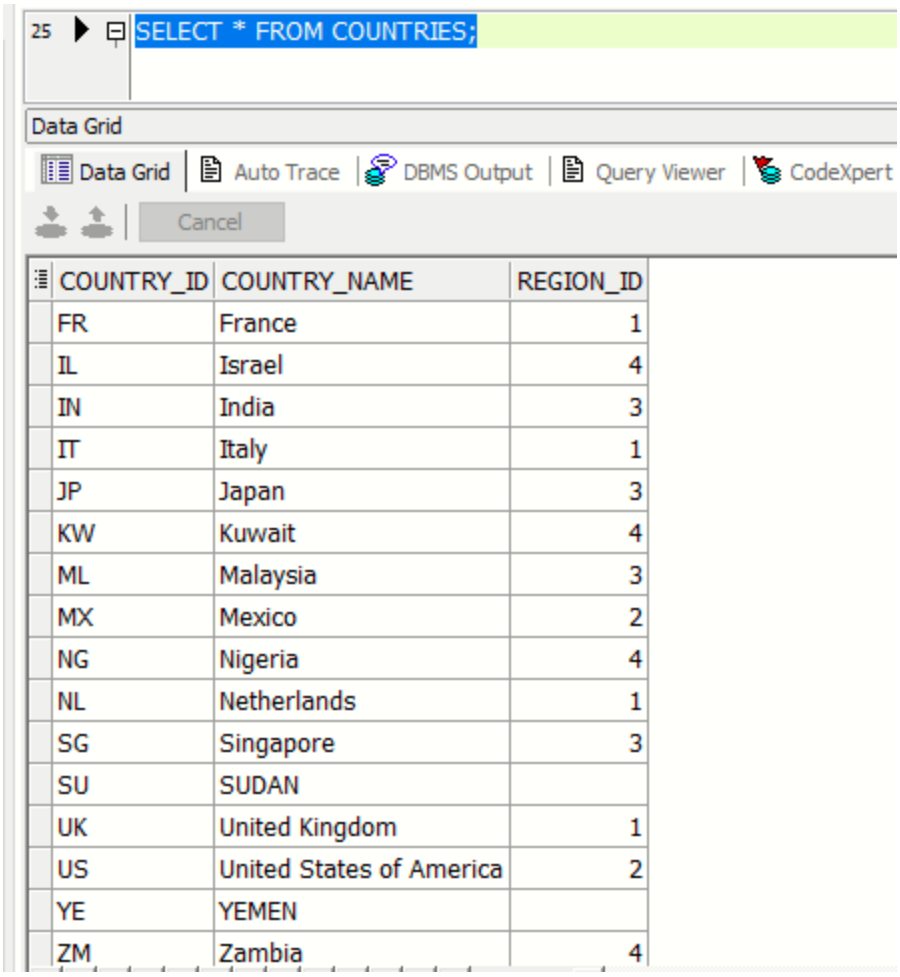


5. **Insert new country in countries table ; insert 2 locations in the previous country; insert 2 departments in the previous locations; insert 2 employees in the previous departments**
   Using insert statements
   Then create trigger to cascade delete the created country;

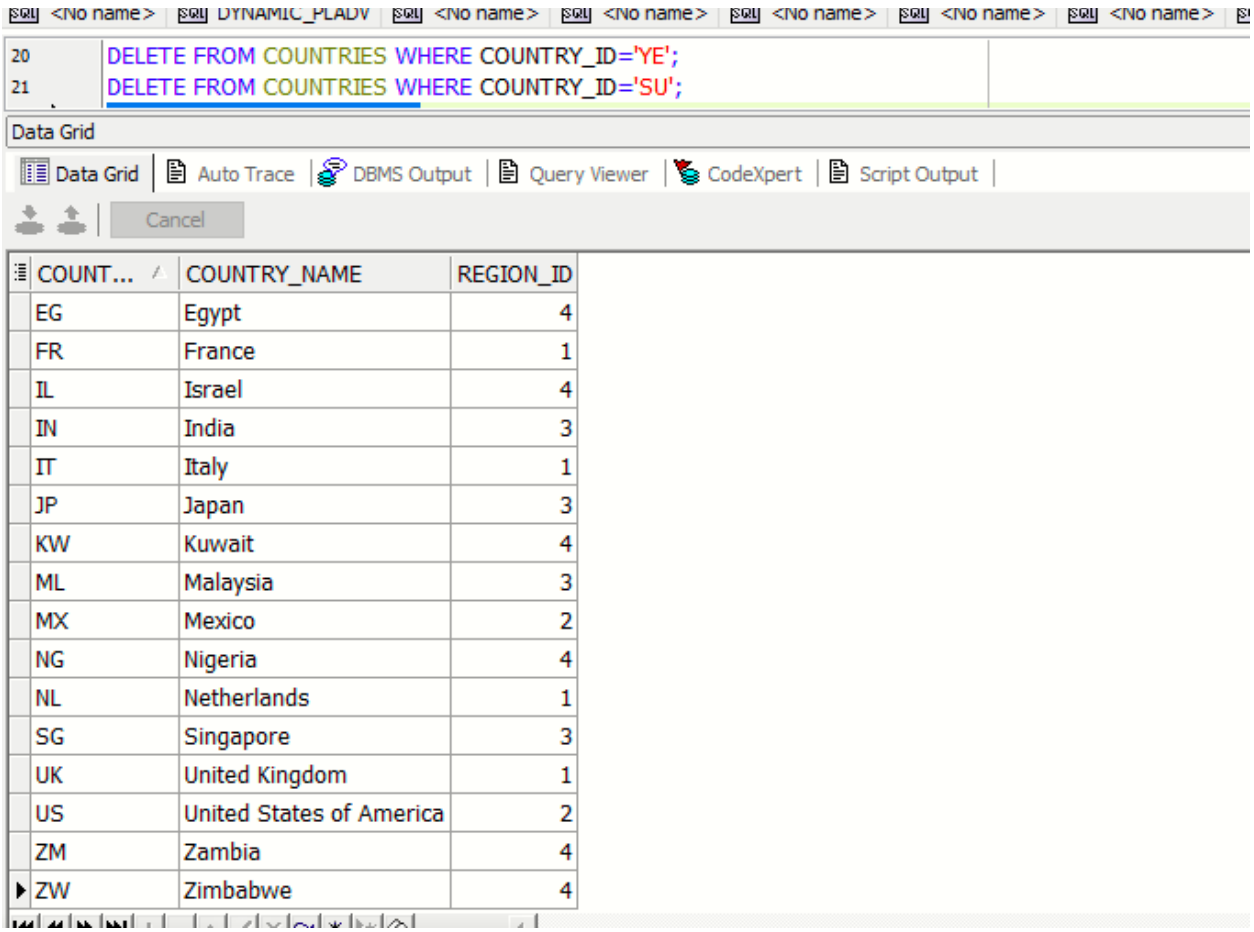
```
set serveroutput on
---Aya lab 3 p(5)

INSERT INTO COUNTRIES(COUNTRY_ID, COUNTRY_NAME)
VALUES('YE','YEMEN');
INSERT INTO COUNTRIES(COUNTRY_ID, COUNTRY_NAME)
VALUES('SU','SUDAN');
INSERT INTO  LOCATIONS (LOCATION_ID, CITY, COUNTRY_ID )
VALUES (3500, 'SANAA', 'YE' );
INSERT INTO  LOCATIONS (LOCATION_ID, CITY, COUNTRY_ID )
VALUES (3600, 'KHARTOM', 'SU' );
INSERT INTO  DEPARTMENTS (DEPARTMENT_ID, DEPARTMENT_NAME, LOCATION_ID)
VALUES (800, 'HEALTHCARE', 3500);
INSERT INTO  DEPARTMENTS (DEPARTMENT_ID, DEPARTMENT_NAME, LOCATION_ID)
VALUES (810, 'MINING', 3600 );
INSERT INTO EMPLOYEES(EMPLOYEE_ID, LAST_NAME, EMAIL, HIRE_DATE,    JOB_ID,DEPARTMENT_ID)
VALUES (208,'ADAM','AD@GMAIL.COM',to_date('05/01/2012','dd/mm/yyyy'),'IT_PROG',800);
INSERT INTO EMPLOYEES (EMPLOYEE_ID, LAST_NAME, EMAIL, HIRE_DATE,    JOB_ID,DEPARTMENT_ID)
VALUES (209,'HOSSAM','HOS@GMAIL.COM',to_date('22/10/2009','dd/mm/yyyy'),'FI_MGR',810);
```

```
1  SET SERVEROUTPUT ON
2  ---Aya lab 3 p(5)
3  CREATE OR REPLACE TRIGGER DELETE_COUNTRY_CASCADE
4  BEFORE DELETE ON COUNTRIES
5  FOR EACH ROW
6  BEGIN
7  DELETE FROM EMPLOYEES
8  WHERE DEPARTMENT_ID IN
9  (SELECT DEPARTMENT_ID FROM DEPARTMENTS where LOCATION_ID IN
10
11 (SELECT LOCATION_ID FROM LOCATIONS WHERE COUNTRY_ID =: OLD.COUNTRY_ID);
12 DELETE FROM DEPARTMENTS
13 where LOCATION_ID in (SELECT LOCATION_ID FROM LOCATIONS WHERE COUNTRY_ID =: OLD.COUNTRY_ID);
14 DELETE FROM LOCATIONS
15 WHERE COUNTRY_ID= :OLD.COUNTRY_ID;
16 END;
17 SELECT * FROM COUNTRIES;
18 DELETE FROM COUNTRIES WHERE COUNTRY_ID='YE';
19 DELETE FROM COUNTRIES WHERE COUNTRY_ID='SU';
20 SELECT * FROM COUNTRIES;
```



COUNTRY_ID	COUNTRY_NAME	REGION_ID
FR	France	1
IL	Israel	4
IN	India	3
IT	Italy	1
JP	Japan	3
KW	Kuwait	4
ML	Malaysia	3
MX	Mexico	2
NG	Nigeria	4
NL	Netherlands	1
SG	Singapore	3
SU	SUDAN	
UK	United Kingdom	1
US	United States of America	2
YE	YEMEN	
ZM	Zambia	4



COUNT...	COUNTRY_NAME	REGION_ID
EG	Egypt	4
FR	France	1
IL	Israel	4
IN	India	3
IT	Italy	1
JP	Japan	3
KW	Kuwait	4
ML	Malaysia	3
MX	Mexico	2
NG	Nigeria	4
NL	Netherlands	1
SG	Singapore	3
UK	United Kingdom	1
US	United States of America	2
ZM	Zambia	4
ZW	Zimbabwe	4