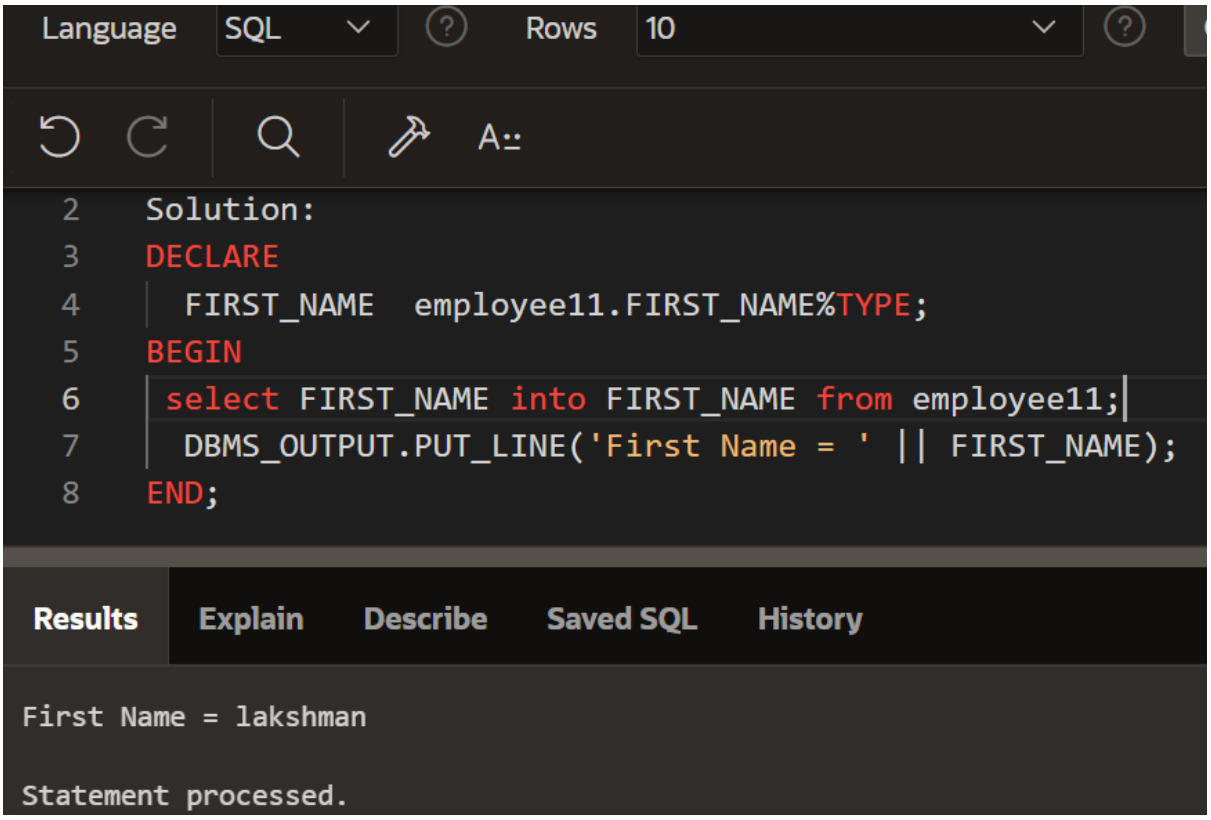
# DBMS (PL/SQL) Test -8

1. Write PL/SQL blocks to retrieve the first _name from the employees table.

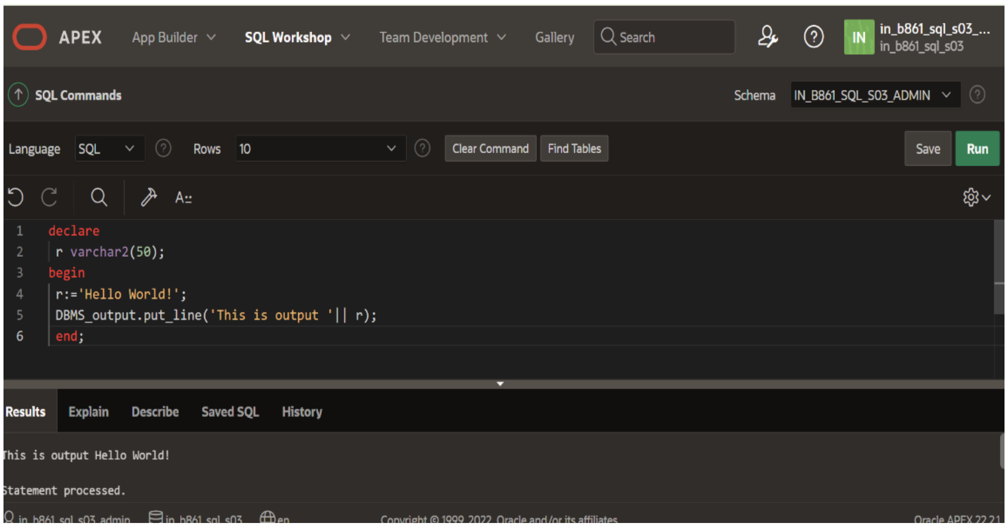


```
Solution:
DECLARE
  FIRST_NAME   employee11.FIRST_NAME%TYPE;
BEGIN
 select FIRST_NAME into FIRST_NAME from employee11;
  DBMS_OUTPUT.PUT_LINE('First Name = ' || FIRST_NAME);
END;
```

```
First Name = lakshman

Statement processed.
```

2. Write a PL/SQL block to display Hello World.

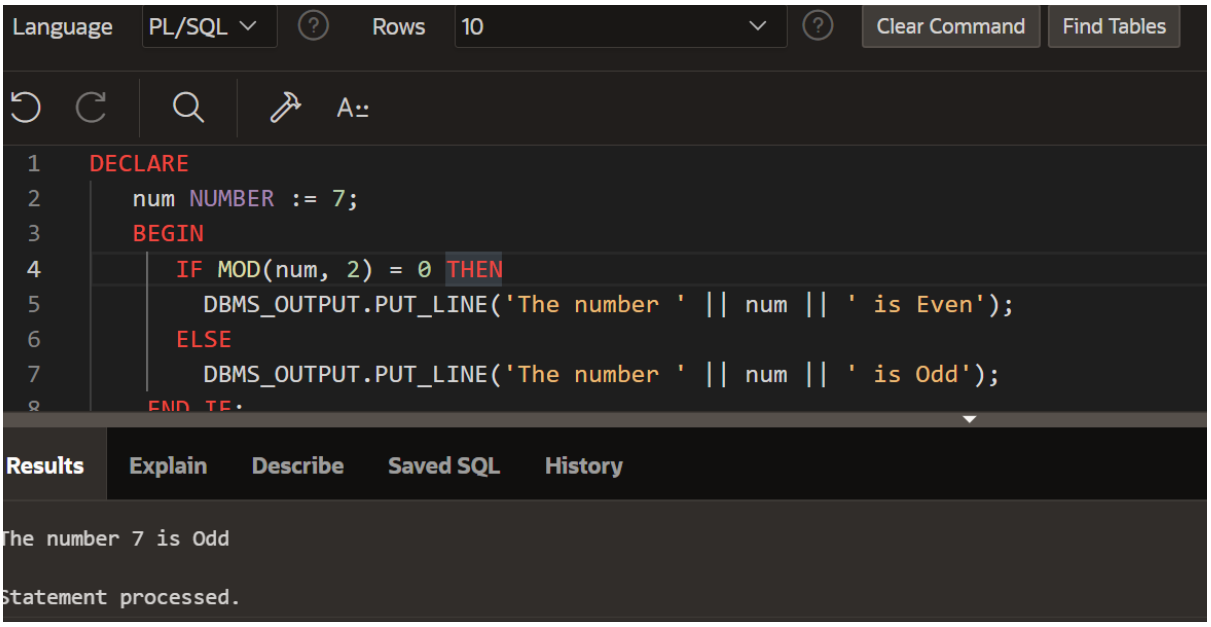


```
declare
 r varchar2(50);
begin
 r:='Hello World!';
 DBMS_output.put_line('This is output '|| r);
 end;
```

```
This is output Hello World!

Statement processed.
```

3. Write a PL/SQL program to check whether a number is even or odd.



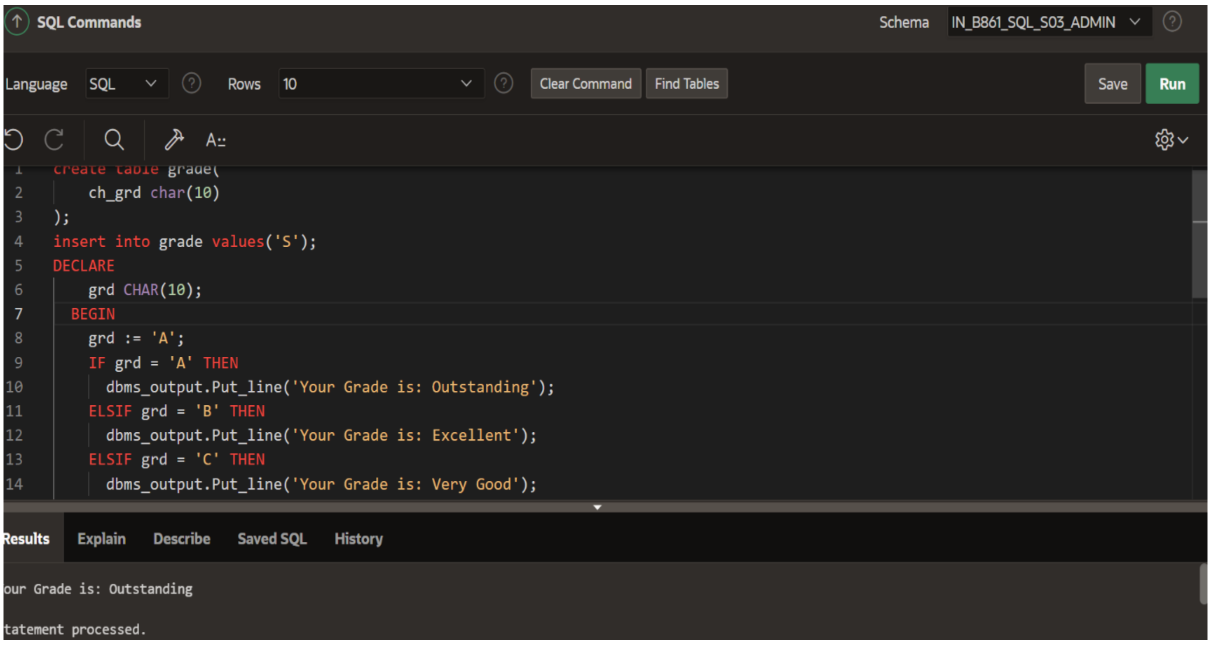
Language PL/SQL Rows 10

```
DECLARE
   num NUMBER := 7;
   BEGIN
      IF MOD(num, 2) = 0 THEN
        DBMS_OUTPUT.PUT_LINE('The number ' || num || ' is Even');
      ELSE
        DBMS_OUTPUT.PUT_LINE('The number ' || num || ' is Odd');
    END IF;
```

```
The number 7 is Odd

Statement processed.
```

4. Write a PL/SQL program to display the description against a grade.



Schema IN_B861_SQL_S03_ADMIN

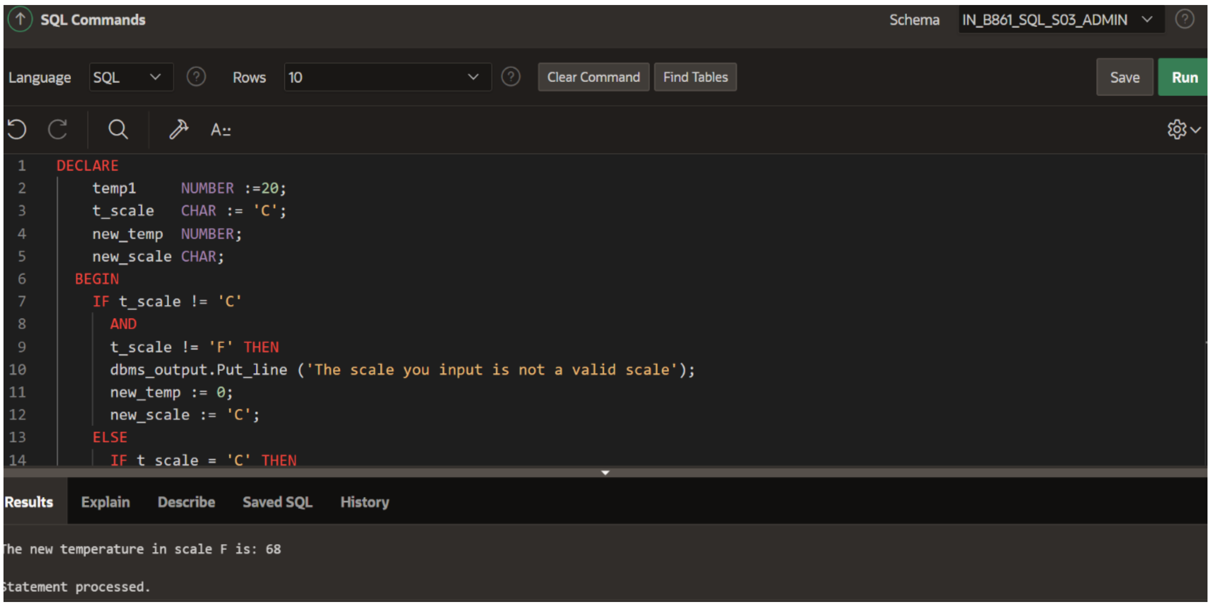
Language SQL Rows 10

```
create table grade(
    ch_grd char(10)
);
insert into grade values('S');
DECLARE
    grd CHAR(10);
  BEGIN
    grd := 'A';
    IF grd = 'A' THEN
      dbms_output.Put_line('Your Grade is: Outstanding');
    ELSIF grd = 'B' THEN
      dbms_output.Put_line('Your Grade is: Excellent');
    ELSIF grd = 'C' THEN
      dbms_output.Put_line('Your Grade is: Very Good');
```

```
our Grade is: Outstanding

tatement processed.
```

5. Write a PL/SQL program to convert a temperature in scale Fahrenheit to Celsius and vice versa.

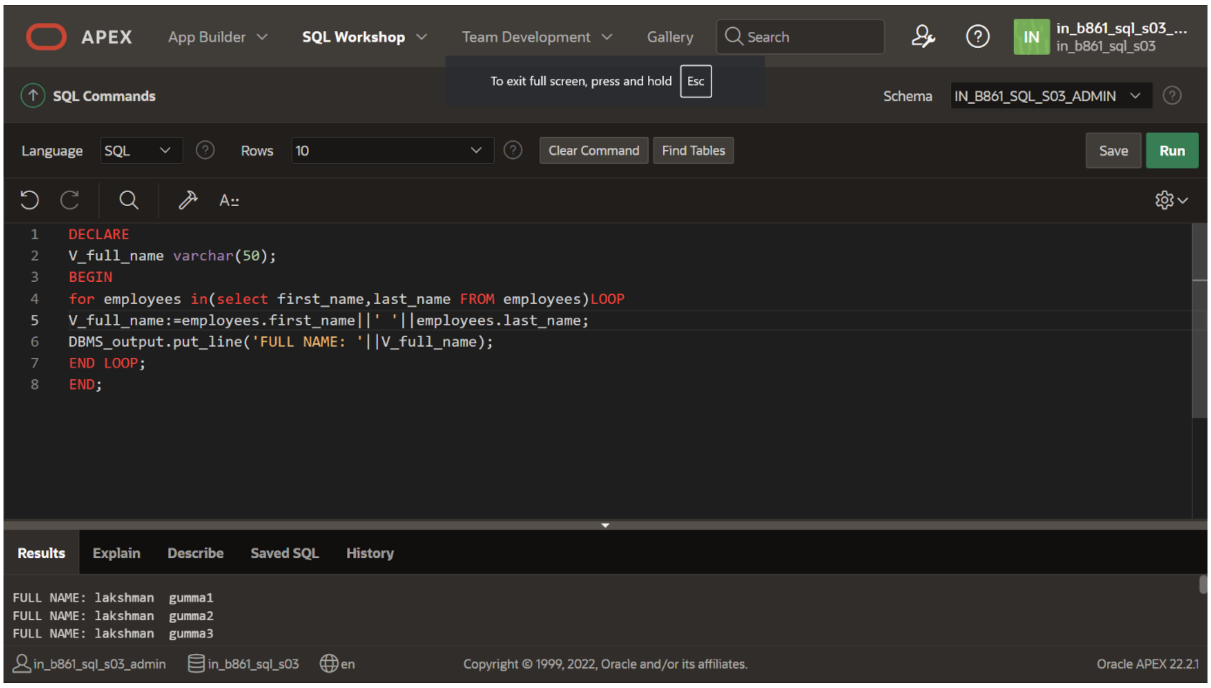


```
DECLARE
    temp1     NUMBER :=20;
    t_scale   CHAR := 'C';
    new_temp  NUMBER;
    new_scale CHAR;
  BEGIN
    IF t_scale != 'C'
      AND
      t_scale != 'F' THEN
      dbms_output.Put_line ('The scale you input is not a valid scale');
      new_temp := 0;
      new_scale := 'C';
    ELSE
      IF t_scale = 'C' THEN
```

```
he new temperature in scale F is: 68

tatement processed.
```

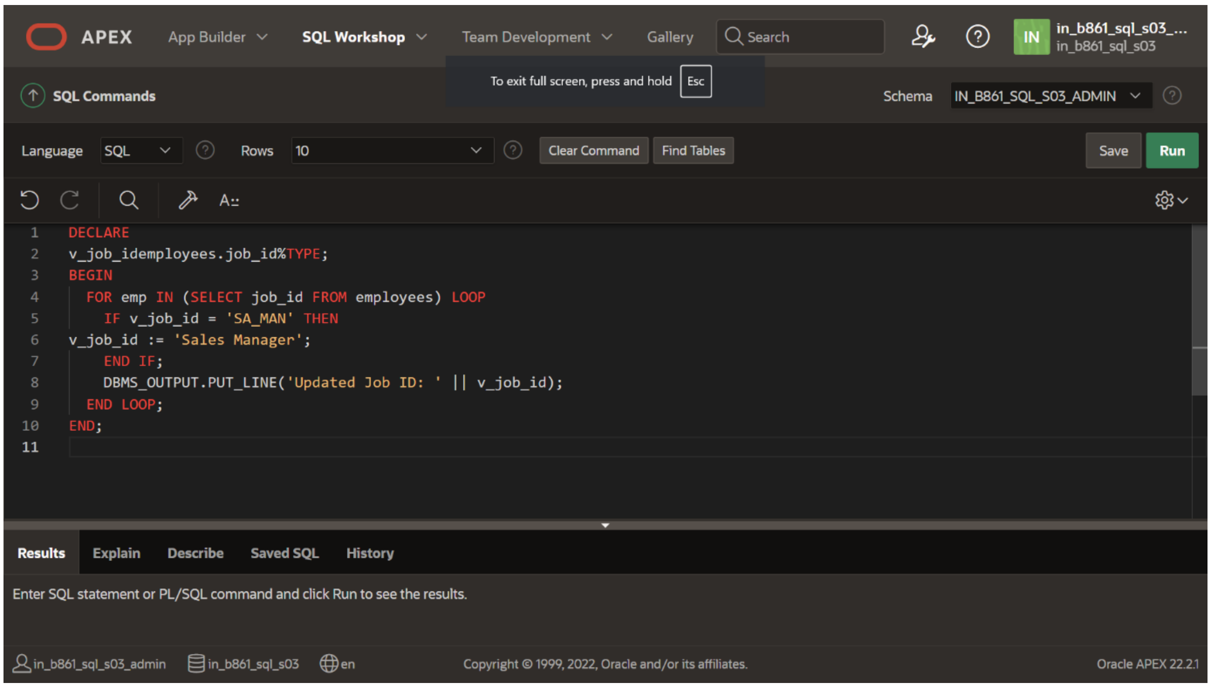
6. Write a PL/SQL block to concatenate the first name and last name of each employee in the employees table and display the full name.



```
DECLARE
V_full_name varchar(50);
BEGIN
for employees in(select first_name,last_name FROM employees)LOOP
V_full_name:=employees.first_name||' '||employees.last_name;
DBMS_output.put_line('FULL NAME: '||V_full_name);
END LOOP;
END;
```

```
FULL NAME: lakshman  gumma1
FULL NAME: lakshman  gumma2
FULL NAME: lakshman  gumma3
```

7. Write a PL/SQL block that replaces all occurrences of the substring 'SA_MAN' with Sales Manager' in the job titles of employees in the employees table. Display the updated job titles.

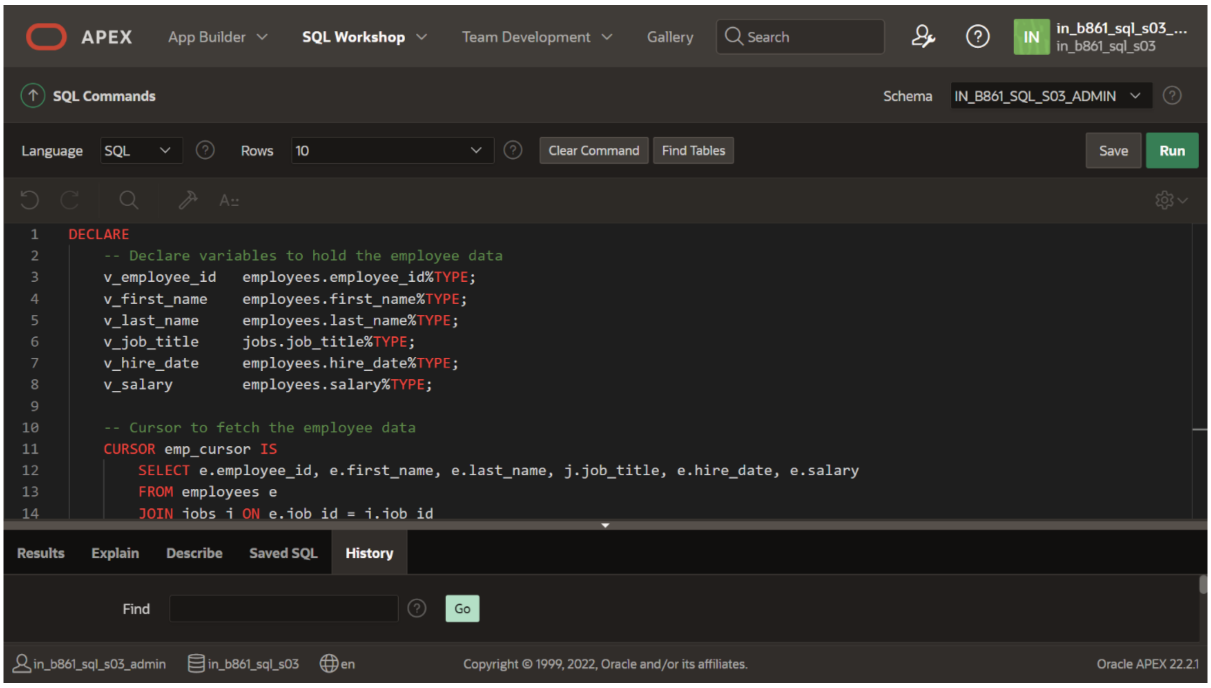


```
DECLARE
v_job_idemployees.job_id%TYPE;
BEGIN
  FOR emp IN (SELECT job_id FROM employees) LOOP
    IF v_job_id = 'SA_MAN' THEN
v_job_id := 'Sales Manager';
    END IF;
    DBMS_OUTPUT.PUT_LINE('Updated Job ID: ' || v_job_id);
  END LOOP;
END;
```

Enter SQL statement or PL/SQL command and click Run to see the results.

in_b861_sql_s03_admin   in_b861_sql_s03   en   Copyright © 1999, 2022, Oracle and/or its affiliates.   Oracle APEX 22.2.1

8. Write a PL/SQL program to display the employee IDs, names, job titles, hire dates , and salaries of all employees.

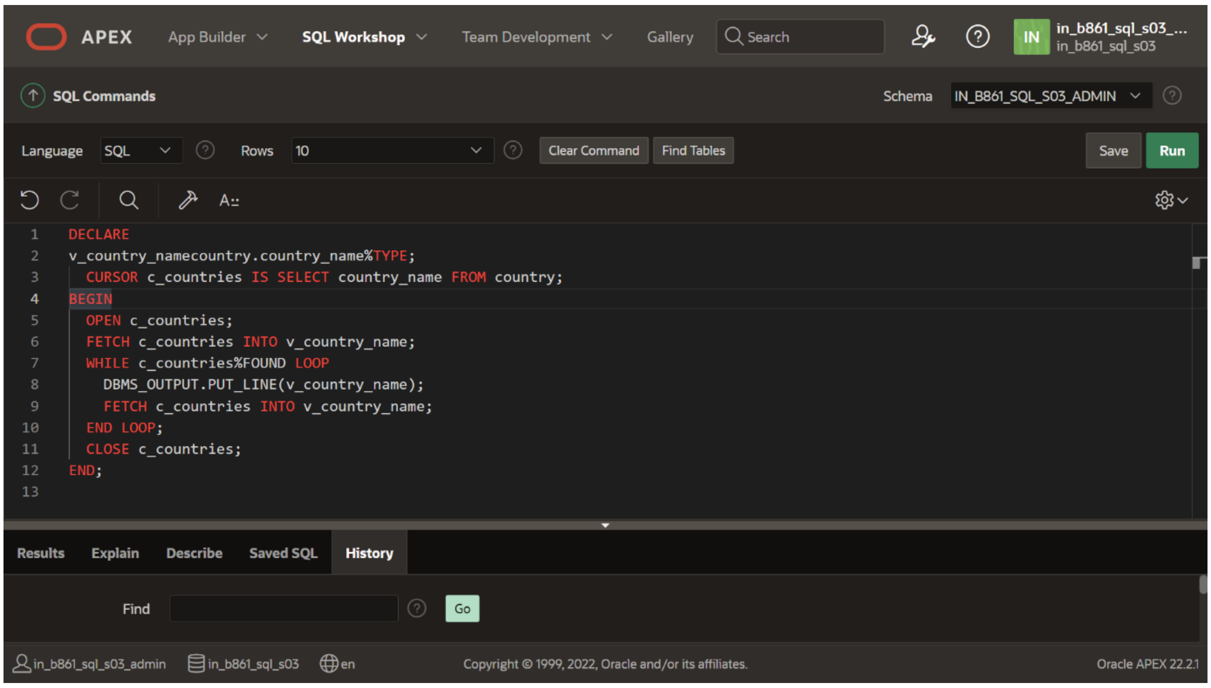


```
DECLARE
    -- Declare variables to hold the employee data
    v_employee_id   employees.employee_id%TYPE;
    v_first_name    employees.first_name%TYPE;
    v_last_name     employees.last_name%TYPE;
    v_job_title     jobs.job_title%TYPE;
    v_hire_date     employees.hire_date%TYPE;
    v_salary        employees.salary%TYPE;

    -- Cursor to fetch the employee data
    CURSOR emp_cursor IS
        SELECT e.employee_id, e.first_name, e.last_name, j.job_title, e.hire_date, e.salary
        FROM employees e
        JOIN jobs j ON e.job_id = j.job_id
```

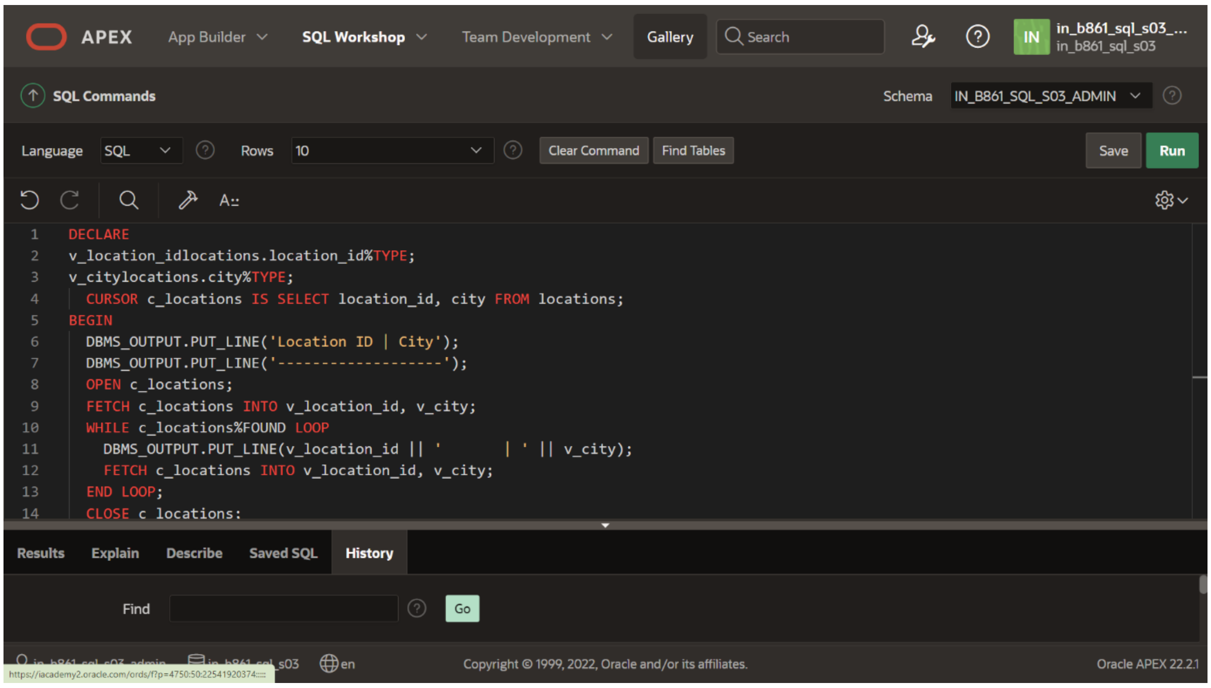
in_b861_sql_s03_admin   in_b861_sql_s03   en   Copyright © 1999, 2022, Oracle and/or its affiliates.   Oracle APEX 22.2.1

9. Write a PL/SQL program to display the names of all countries.



```
DECLARE
v_country_namecountry.country_name%TYPE;
  CURSOR c_countries IS SELECT country_name FROM country;
BEGIN
  OPEN c_countries;
  FETCH c_countries INTO v_country_name;
  WHILE c_countries%FOUND LOOP
    DBMS_OUTPUT.PUT_LINE(v_country_name);
    FETCH c_countries INTO v_country_name;
  END LOOP;
  CLOSE c_countries;
END;
```

10. Write a PL/SQL program to display the job titles of all employees. Return a heading of job title.



```
DECLARE
v_location_idlocations.location_id%TYPE;
v_citylocations.city%TYPE;
  CURSOR c_locations IS SELECT location_id, city FROM locations;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Location ID | City');
  DBMS_OUTPUT.PUT_LINE('-------------------');
  OPEN c_locations;
  FETCH c_locations INTO v_location_id, v_city;
  WHILE c_locations%FOUND LOOP
    DBMS_OUTPUT.PUT_LINE(v_location_id || '       | ' || v_city);
    FETCH c_locations INTO v_location_id, v_city;
  END LOOP;
  CLOSE c_locations;
```