**Quiz**

When using the SELECT statement in PL/SQL, the INTO clause is required and queries can return one or more row.

1. True

2. False

**Answer: False, queries must return only one row**

**Practice 4: Interacting with the Oracle Server**

**In this practice, you use PL/SQL code to interact with the Oracle Server.**

1) Create a PL/SQL block that selects the maximum department ID in the departments table and stores it in the v\_max\_deptno variable. Display the maximum department ID.

a) Declare a variable v\_max\_deptno of type NUMBER in the declarative section.

**DECLARE**

**v\_max\_deptno NUMBER;**

**Изображение выглядит как текст

Автоматически созданное описание**

b) Start the executable section with the BEGIN keyword and include a SELECT statement to retrieve the maximum department\_id from the departments table.

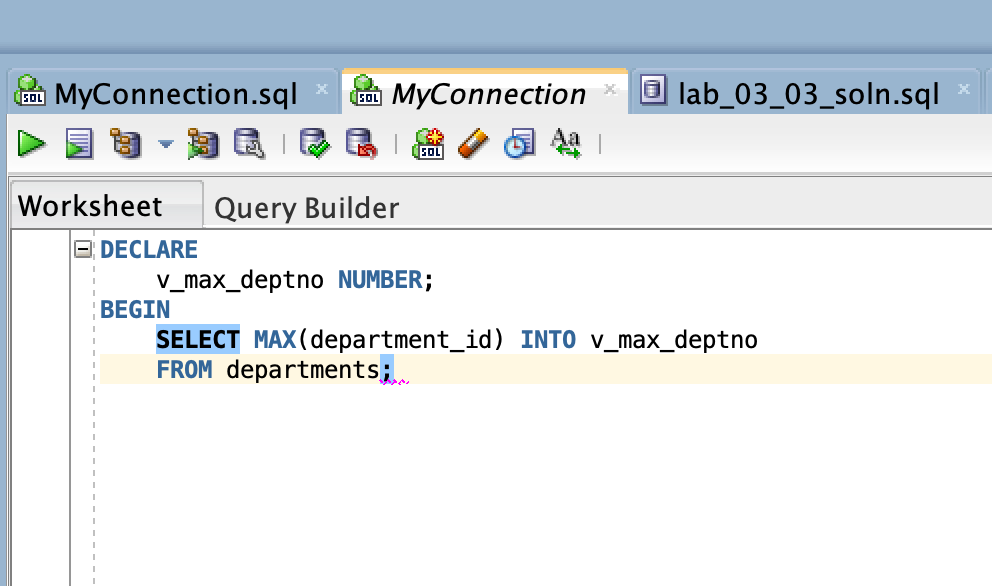
**DECLARE**

**v\_max\_deptno NUMBER;**

**BEGIN**

**SELECT MAX(department\_id) INTO v\_max\_deptno**

**FROM departments;**



c) Display v\_max\_deptno and end the executable block.

**SET SERVEROUTPUT ON**

**DECLARE**

**v\_max\_deptno NUMBER;**

**BEGIN**

**SELECT MAX(department\_id) INTO v\_max\_deptno**

**FROM departments;**

**DBMS\_OUTPUT.PUT\_LINE('The maximum department\_id: ' || v\_max\_deptno);**

**END;**

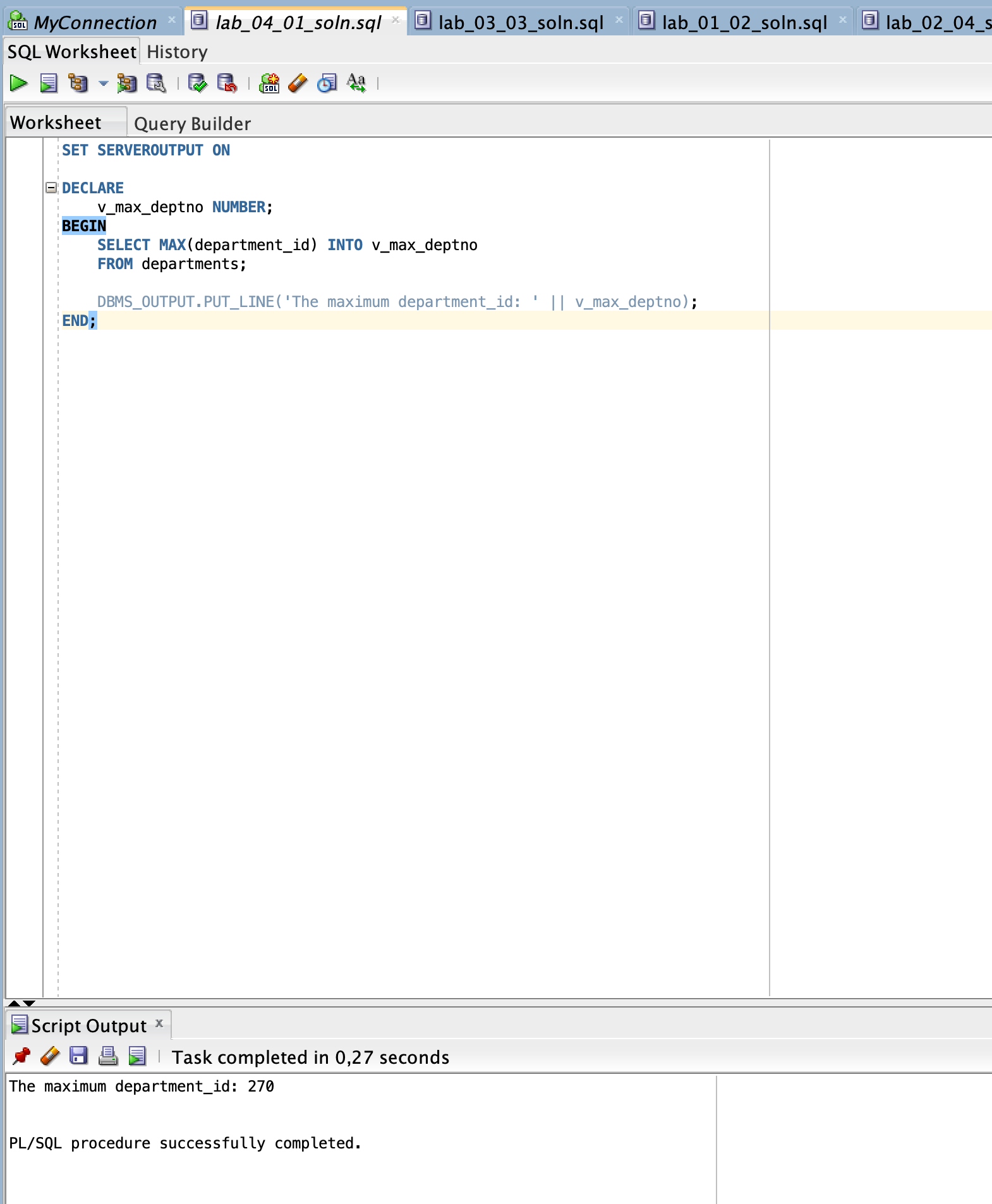
Изображение выглядит как текст

Автоматически созданное описание

d) Execute and save your script as lab\_04\_01\_soln.sql. The sample output is as follows:

Изображение выглядит как текст

Автоматически созданное описание



2) Modify the PL/SQL block that you created in step 1 to insert a new department into the departments table.

a) Load the lab\_04\_01\_soln.sql script. Declare two variables: v\_dept\_name of type departments.department\_name and v\_dept\_id of type NUMBER

Assign 'Education' to v\_dept\_name in the declarative section.

**DECLARE**

**v\_max\_deptno NUMBER;**

**v\_dept\_name departments.department\_name%TYPE := 'Education';**

**v\_dept\_id NUMBER;**

Изображение выглядит как текст

Автоматически созданное описание

b) You have already retrieved the current maximum department number from the departments table. Add 10 to it and assign the result v\_dept\_id.

**v\_dept\_id := v\_max\_deptno + 10;**

**Изображение выглядит как текст

Автоматически созданное описание**

c) Include an INSERT statement to insert data into the department\_name, department\_id, and location\_id columns of the departments table. Use values in dept\_name and dept\_id for department\_name and department\_id, respectively, and use NULL for location\_id.

**INSERT INTO departments (department\_name, department\_id, location\_id)**

**VALUES (v\_dept\_name, v\_dept\_id, NULL);**

Изображение выглядит как текст

Автоматически созданное описание

d) Use the SQL attribute SQL%ROWCOUNT to display the number of rows that are affected.

**DBMS\_OUTPUT.PUT\_LINE('Number of rows that are affected: ' || SQL%ROWCOUNT);**

**Изображение выглядит как текст

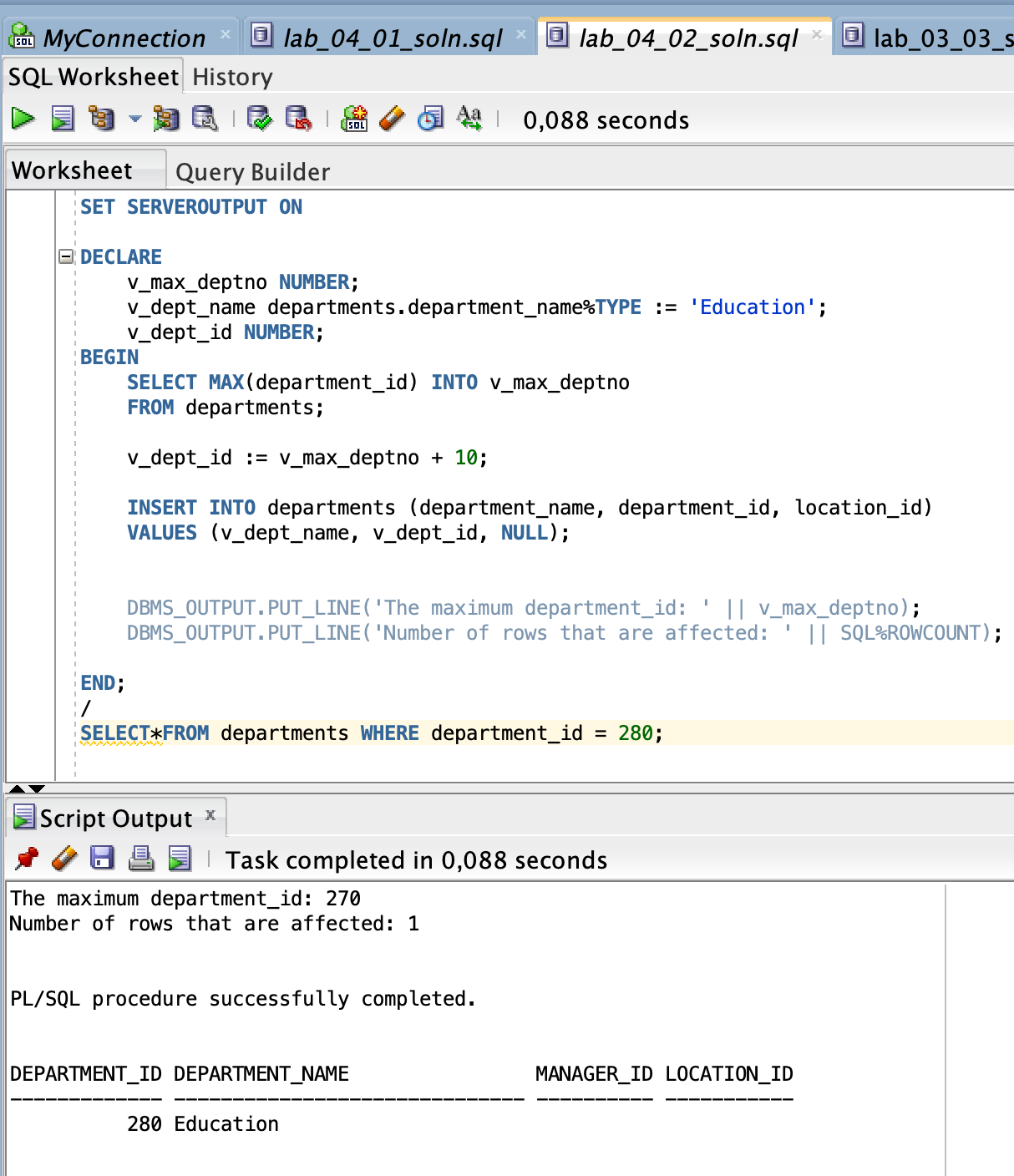
Автоматически созданное описание**

e) Execute a SELECT statement to check whether the new department is inserted. You can terminate the PL/SQL block with "/" and include the SELECT statement in your script.

**/**

**SELECT\*FROM departments WHERE department\_id = 280;**

f) Execute and save your script as lab\_04\_02\_soln.sql. The sample output is as follows:Изображение выглядит как стол

Автоматически созданное описание

3) In step 2, you set location\_id to NULL. Create a PL/SQL block that updates the location\_id to 3000 for the new department.

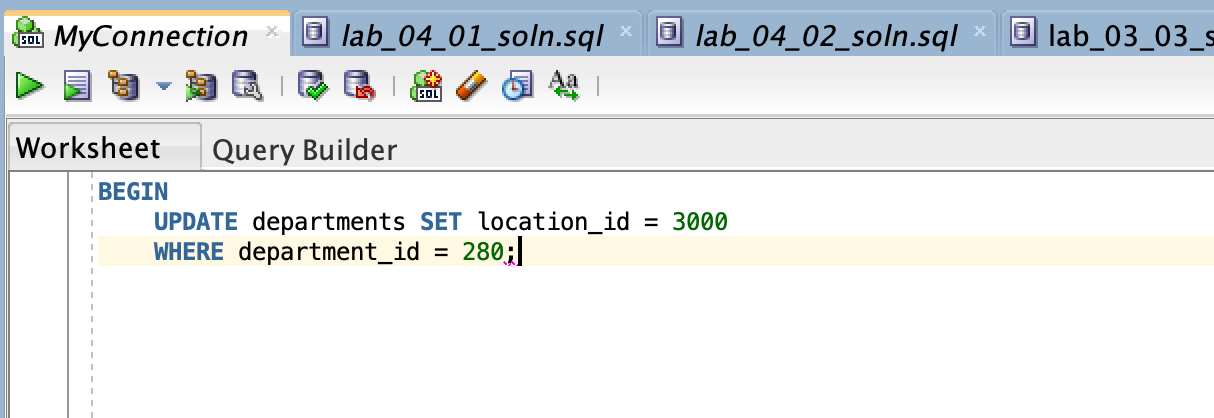
Note: If you successfully completed step 2, continue with step 3a. If not, first execute the solution script /soln/sol\_04\_02.sql.

a) Start the executable block with the BEGIN keyword. Include the UPDATE statement to set the location\_id to 3000 for the new department (dept\_id=280).

**BEGIN**

**UPDATE departments SET location\_id = 3000**

**WHERE department\_id = 280;**

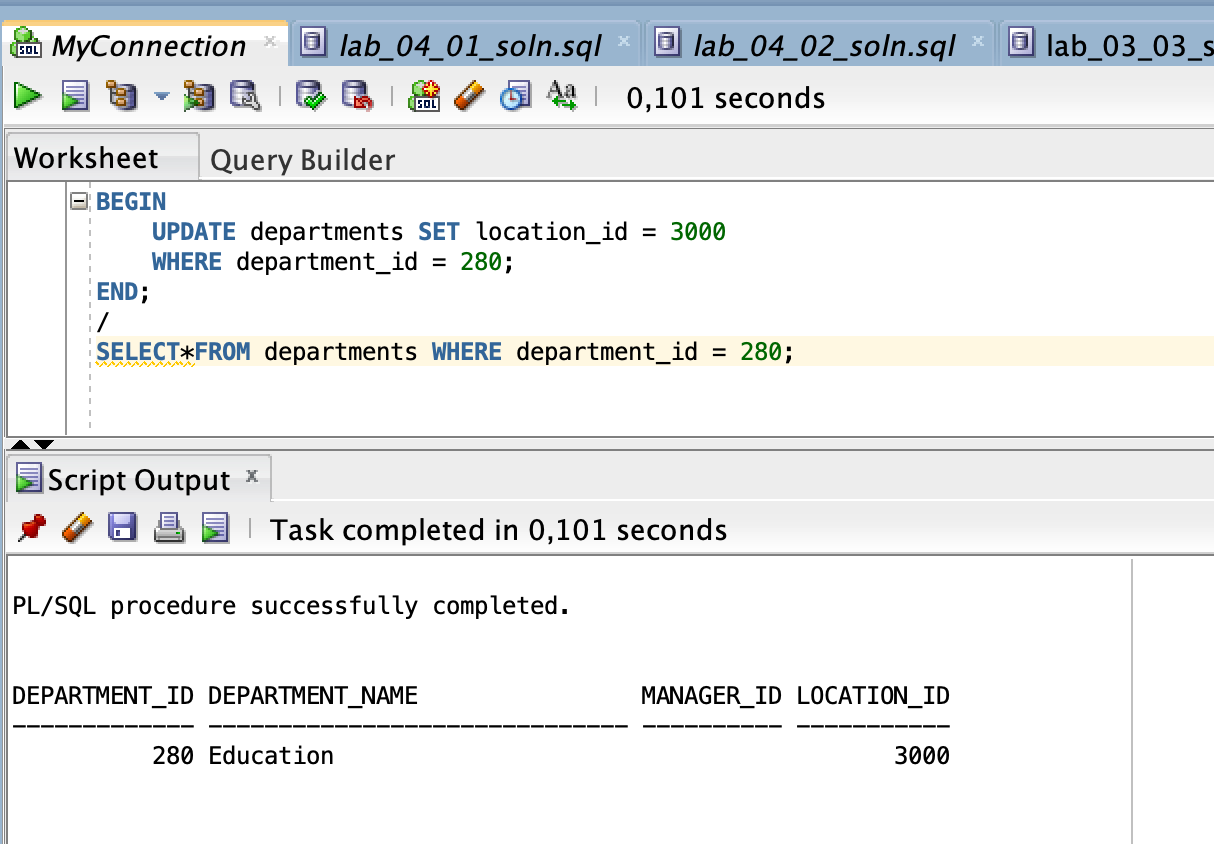
****

b) End the executable block with the END keyword. Terminate the PL/SQL block with "/" and include a SELECT statement to display the department that you updated.

**END;**

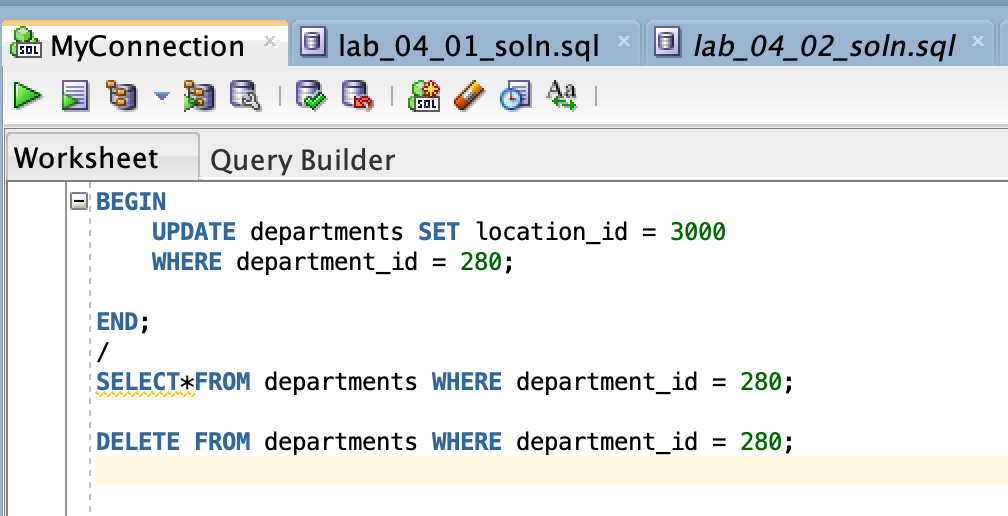
**/**

**SELECT\*FROM departments WHERE department\_id = 280;**

****

c) Include a DELETE statement to delete the department that you added.

**DELETE FROM departments WHERE department\_id = 280;**

****

d) Execute and save your script as lab\_04\_03\_soln.sql. The sample output is as follows: Изображение выглядит как текст, стол

Автоматически созданное описание

