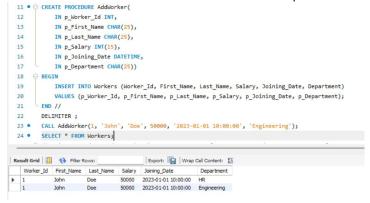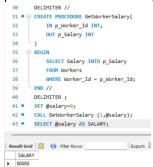1. Consider the Worker table with following fields: Worker_Id INT FirstName CHAR(25), LastName CHAR(25), Salary INT(15), JoiningDate DATETIME, Department CHAR(25)).

```
1 •   CREATE TABLE Workers
2   ⊖ (Worker_Id INT ,
3       First_Name CHAR(25),
4       Last_Name CHAR(25),
5       Salary INT(15),
6       Joining_Date DATETIME,
7     ─ Department CHAR(25));
```
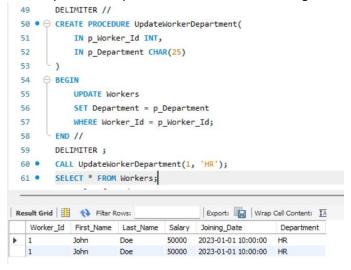
2. Create a stored procedure that takes in IN parameters for all the columns in the Worker table and adds a new record to the table and then invokes the procedure call.

```
11 • ⊖ CREATE PROCEDURE AddWorker(
12        IN p_Worker_Id INT,
13        IN p_First_Name CHAR(25),
14        IN p_Last_Name CHAR(25),
15        IN p_Salary INT(15),
16        IN p_Joining_Date DATETIME,
17        IN p_Department CHAR(25))
18   ⊖ BEGIN
19        INSERT INTO Workers (Worker_Id, First_Name, Last_Name, Salary, Joining_Date, Department)
20        VALUES (p_Worker_Id, p_First_Name, p_Last_Name, p_Salary, p_Joining_Date, p_Department);
21    ─ END //
22      DELIMITER ;
23 •   CALL AddWorker(1, 'John', 'Doe', 50000, '2023-01-01 10:00:00', 'Engineering');
24 •   SELECT * FROM Workers;
```

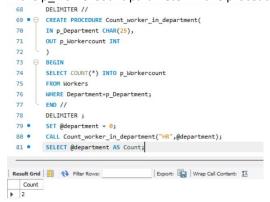| Result Grid		Filter Rows:		Export:	Wrap Cell Content: IA	
	Worker_Id	First_Name	Last_Name	Salary	Joining_Date	Department
▶	1	John	Doe	50000	2023-01-01 10:00:00	HR
	1	John	Doe	50000	2023-01-01 10:00:00	Engineering

3. Write stored procedure takes in an IN parameter for WORKER_ID and an OUT parameter for SALARY. It should retrieve the salary of the worker with the given ID and returns it in the p_salary parameter. Then make the procedure call.

```
30      DELIMITER //
31 • ⊖ CREATE PROCEDURE GetWorkerSalary(
32        IN p_Worker_Id INT,
33        OUT p_Salary INT
34    ─ )
35   ⊖ BEGIN
36        SELECT Salary INTO p_Salary
37        FROM Workers
38        WHERE Worker_Id = p_Worker_Id;
39    ─ END //
40      DELIMITER ;
41 •   SET @salary=0;
42 •   CALL GetWorkerSalary (1,@salary);
43 •   SELECT @salary AS SALARY;
```

Result Grid		Filter Rows:		Export:
	SALARY			
▶	50000			

4. Create a stored procedure that takes in IN parameters for WORKER_ID and DEPARTMENT. It should update the department of the worker with the given ID. Then make a procedure call.

```
49      DELIMITER //
50 • ⊖ CREATE PROCEDURE UpdateWorkerDepartment(
51          IN p_Worker_Id INT,
52          IN p_Department CHAR(25)
53    ─ )
54   ⊖ BEGIN
55          UPDATE Workers
56          SET Department = p_Department
57          WHERE Worker_Id = p_Worker_Id;
58    ─ END //
59      DELIMITER ;
60 •   CALL UpdateWorkerDepartment(1, 'HR');
61 •   SELECT * FROM Workers;
```

Result Grid		Filter Rows:		Export:	Wrap Cell Content: IA	
	Worker_Id	First_Name	Last_Name	Salary	Joining_Date	Department
▶	1	John	Doe	50000	2023-01-01 10:00:00	HR
	1	John	Doe	50000	2023-01-01 10:00:00	HR

5. Write a stored procedure that takes in an IN parameter for DEPARTMENT and an OUT parameter for p_workerCount. It should retrieve the number of workers in the given department and returns it in the p_workerCount parameter. Make procedure call.

```
68        DELIMITER //
69  •  ⊖  CREATE PROCEDURE Count_worker_in_department(
70          IN p_Department CHAR(25),
71          OUT p_Workercount INT
72        )
73     ⊖  BEGIN
74          SELECT COUNT(*) INTO p_Workercount
75          FROM Workers
76          WHERE Department=p_Department;
77        END //
78        DELIMITER ;
79  •     SET @department = 0;
80  •     CALL Count_worker_in_department("HR",@department);
81  •     SELECT @department AS Count;
```

| Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

Count
2

6. Write a stored procedure that takes in an IN parameter for DEPARTMENT and an OUT parameter for p_avgSalary. It should retrieve the average salary of all workers in the given department and returns it in the p_avgSalary parameter and call the procedure.

```
88        DELIMITER //
89  •  ⊖  CREATE PROCEDURE AverageSalaryInDepartment(
90          IN p_Department CHAR(25),
91          OUT p_AvgSalary DECIMAL(10, 2)
92        )
93     ⊖  BEGIN
94          SELECT AVG(Salary) INTO p_AvgSalary
95          FROM Workers
96          WHERE Department = p_Department;
97        END //
98        DELIMITER ;
99  •     SET @avgSalary = 0;
100 •     CALL AverageSalaryInDepartment('HR', @avgSalary);
101 •     SELECT @avgSalary AS AvgSalary;
```

| Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

AvgSalary
50000.00