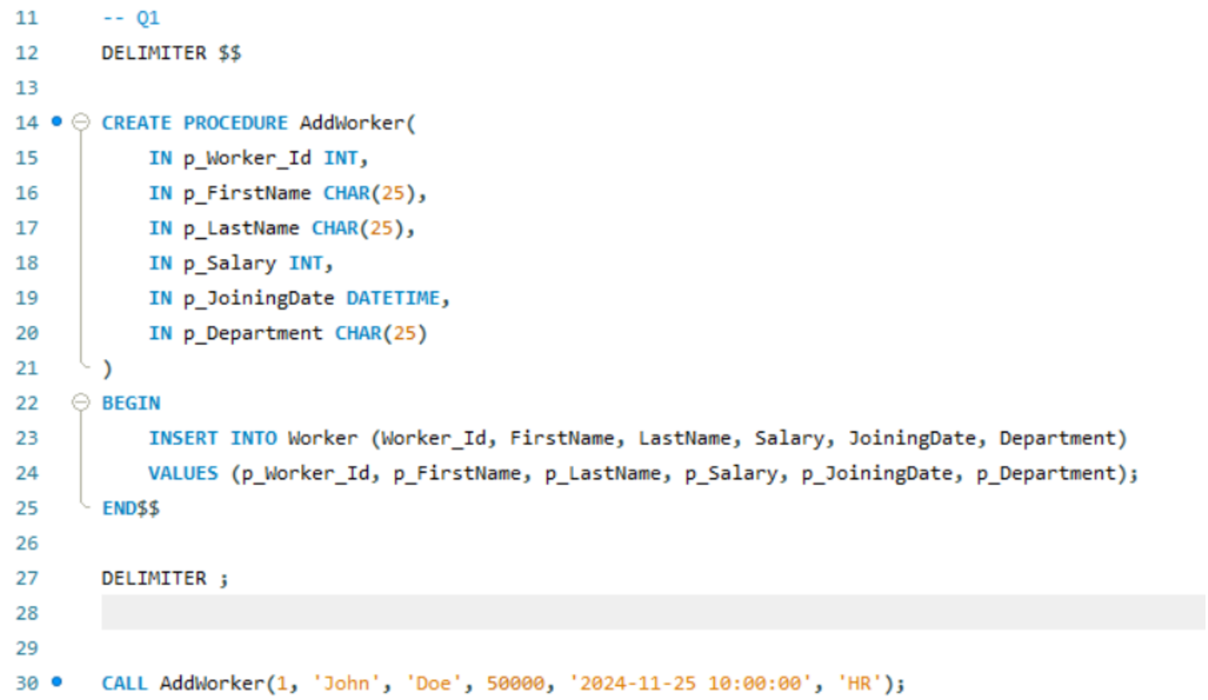
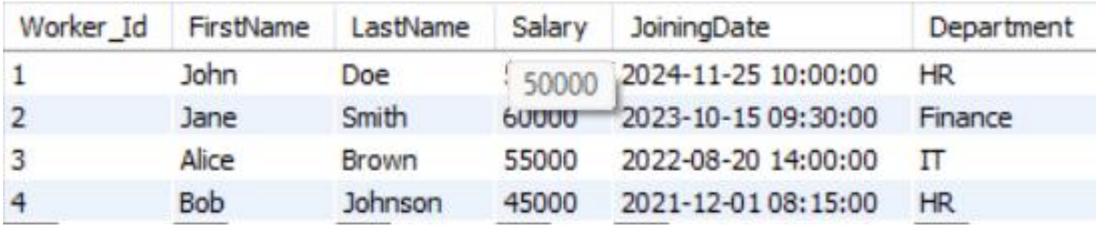
1. Create a stored procedure that takes in IN parameters for all the columns in the Worker table and adds a new record to the table and then invokes the procedure call.

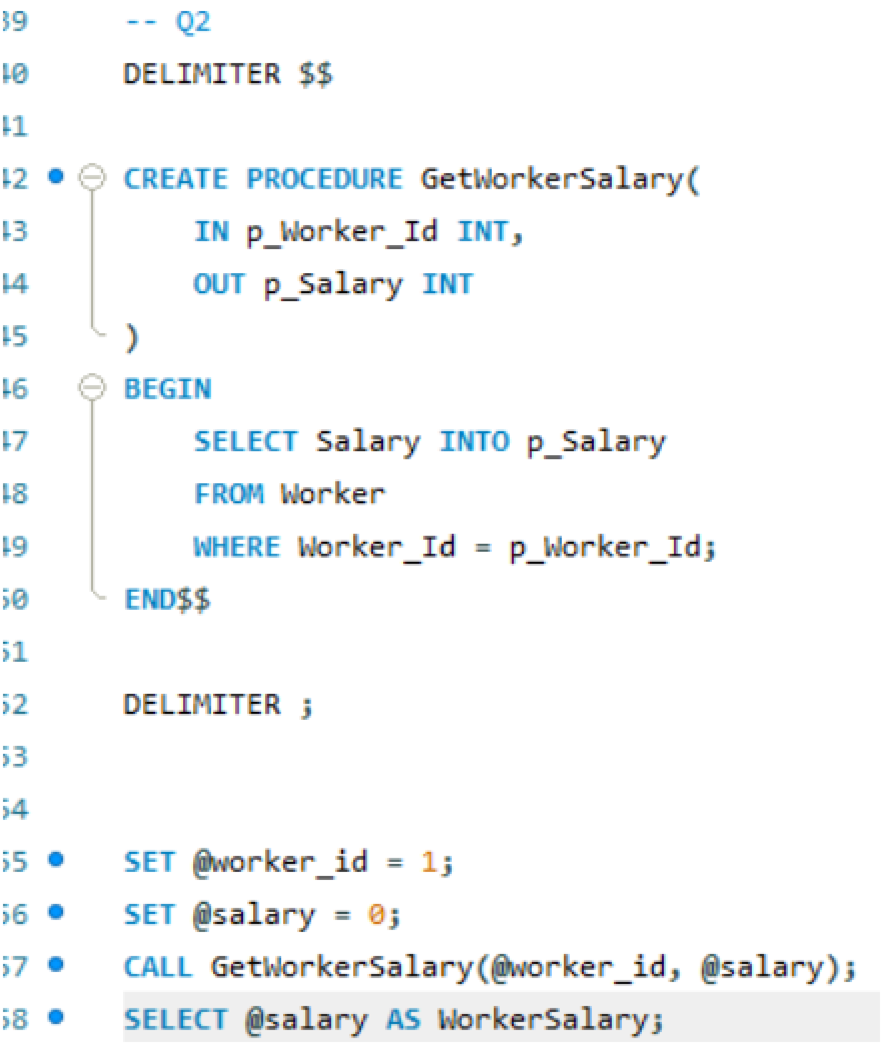


```
-- Q1
DELIMITER $$

CREATE PROCEDURE AddWorker(
    IN p_Worker_Id INT,
    IN p_FirstName CHAR(25),
    IN p_LastName CHAR(25),
    IN p_Salary INT,
    IN p_JoiningDate DATETIME,
    IN p_Department CHAR(25)
)
BEGIN
    INSERT INTO Worker (Worker_Id, FirstName, LastName, Salary, JoiningDate, Department)
    VALUES (p_Worker_Id, p_FirstName, p_LastName, p_Salary, p_JoiningDate, p_Department);
END$$

DELIMITER ;


CALL AddWorker(1, 'John', 'Doe', 50000, '2024-11-25 10:00:00', 'HR');
```

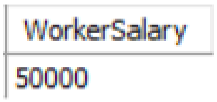


Worker_Id	FirstName	LastName	Salary	JoiningDate	Department
1	John	Doe	50000	2024-11-25 10:00:00	HR
2	Jane	Smith	60000	2023-10-15 09:30:00	Finance
3	Alice	Brown	55000	2022-08-20 14:00:00	IT
4	Bob	Johnson	45000	2021-12-01 08:15:00	HR

2. 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.

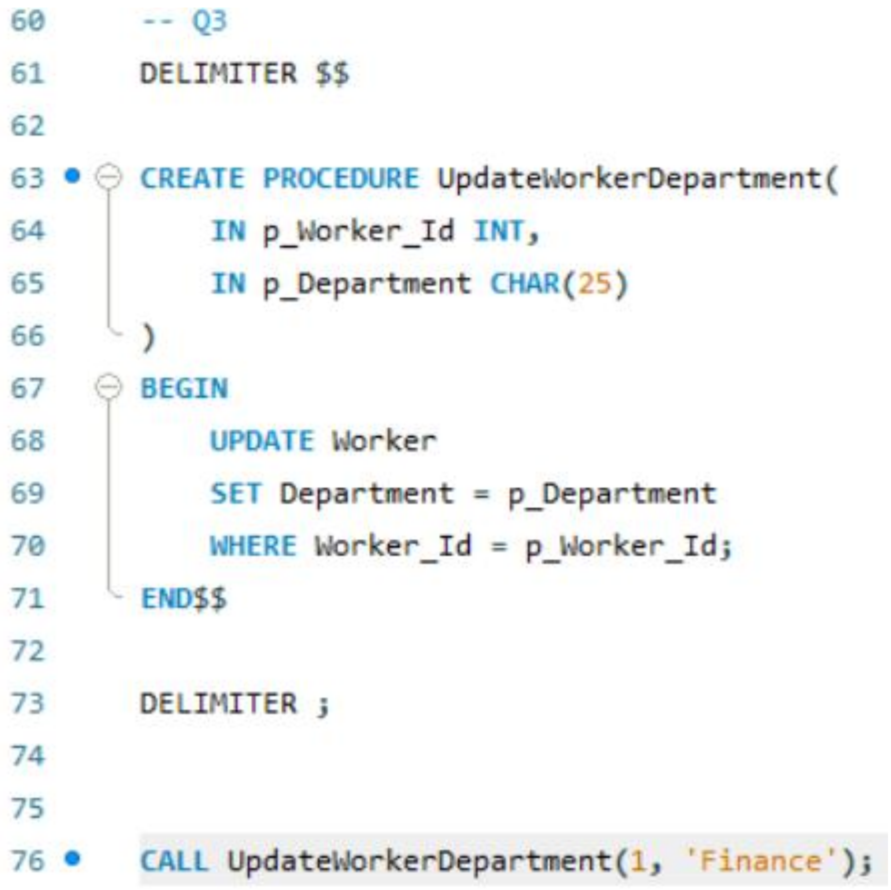


```
-- Q2
DELIMITER $$

CREATE PROCEDURE GetWorkerSalary(
    IN p_Worker_Id INT,
    OUT p_Salary INT
)
BEGIN
    SELECT Salary INTO p_Salary
    FROM Worker
    WHERE Worker_Id = p_Worker_Id;
END$$

DELIMITER ;


SET @worker_id = 1;
SET @salary = 0;
CALL GetWorkerSalary(@worker_id, @salary);
SELECT @salary AS WorkerSalary;
```



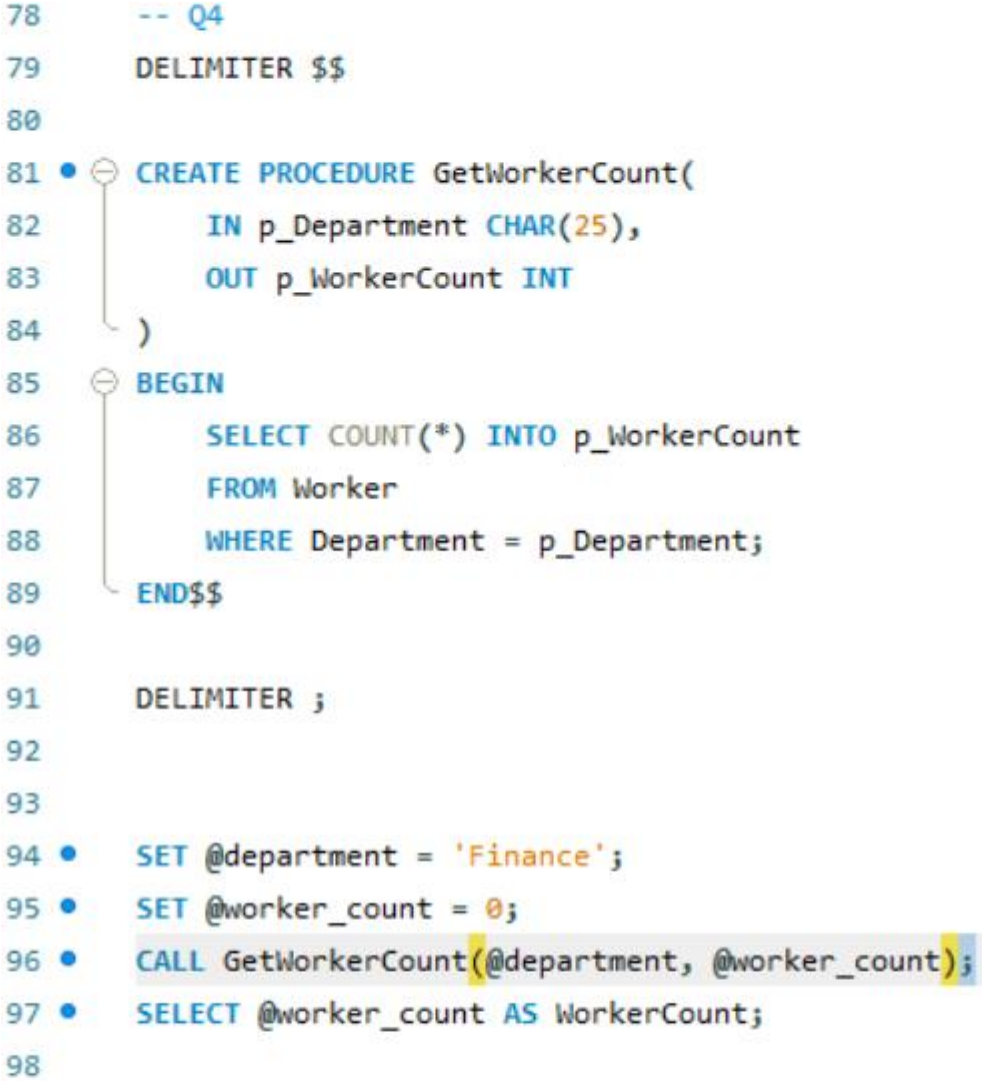
WorkerSalary
50000

3. 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.

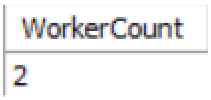


```
-- Q3
DELIMITER $$

CREATE PROCEDURE UpdateWorkerDepartment(
    IN p_Worker_Id INT,
    IN p_Department CHAR(25)
)
BEGIN
    UPDATE Worker
    SET Department = p_Department
    WHERE Worker_Id = p_Worker_Id;
END$$

DELIMITER ;


CALL UpdateWorkerDepartment(1, 'Finance');
```

4. 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.

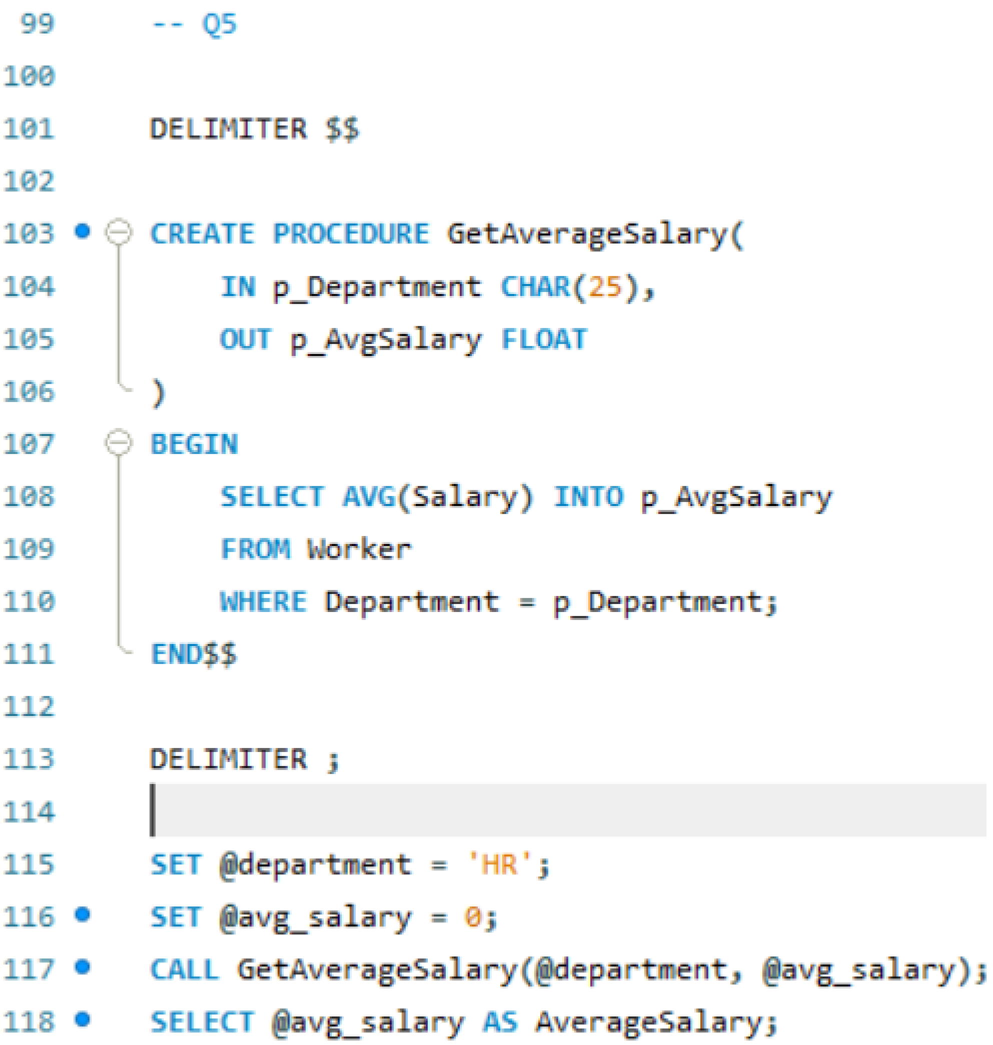


```
-- Q4
DELIMITER $$

CREATE PROCEDURE GetWorkerCount(
    IN p_Department CHAR(25),
    OUT p_WorkerCount INT
)
BEGIN
    SELECT COUNT(*) INTO p_WorkerCount
    FROM Worker
    WHERE Department = p_Department;
END$$

DELIMITER ;


SET @department = 'Finance';
SET @worker_count = 0;
CALL GetWorkerCount(@department, @worker_count);
SELECT @worker_count AS WorkerCount;

```

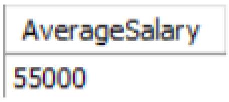


```
WorkerCount
2
```

5. 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.



```
-- Q5

DELIMITER $$

CREATE PROCEDURE GetAverageSalary(
    IN p_Department CHAR(25),
    OUT p_AvgSalary FLOAT
)
BEGIN
    SELECT AVG(Salary) INTO p_AvgSalary
    FROM Worker
    WHERE Department = p_Department;
END$$

DELIMITER ;

SET @department = 'HR';
SET @avg_salary = 0;
CALL GetAverageSalary(@department, @avg_salary);
SELECT @avg_salary AS AverageSalary;
```



AverageSalary
55000