TASK – 8

Consider the Worker table with following fields:

- **Worker_Id INT**
- **FirstName CHAR(25),**
- **LastName CHAR(25),**
- **Salary INT(15),**
- **JoiningDate DATETIME,**
- **Department CHAR(25))**

```sql
CREATE DATABASE work;
USE work;

CREATE TABLE worker (
    Worker_Id INT,
    FirstName CHAR(25),
    LastName CHAR(25),
    Salary INT,
    JoiningDate DATETIME,
    Department CHAR(25)
);

INSERT INTO worker VALUES
(1000, 'ADWIN', 'LOE', 80000, '2020-04-05 10:00:00', 'POWER BI ENGINEER'),
(1001, 'AKSHARA', 'MANOJ', 700000, '2023-08-22 11:30:00', 'SOFTWARE ENGINEER'),
(1002, 'ANNIE', 'SABASTEIN', 60000, '2024-02-17 10:00:00', 'DIGITAL MARKETING'),
(1003, 'ZADHA', 'UWAIZ', 70000, '2023-10-28 12:00:00', 'DATA ANALYST'),
(1004, 'ROSHAN', 'AMAN', 90000, '2019-06-09 10:00:00', 'BUSINESS ANALYST'),
(1005, 'AZLAN', 'AZAAD', 50000, '2024-05-28 10:00:00', 'ACCOUNTANT'),
(1006, 'ROHAN', 'ADITH', 70000, '2023-02-10 10:00:00', 'FINANCE'),
(1007, 'DAVID', 'JOSEPH', 90000, '2019-03-17 12:00:00', 'HR MANAGER'),
(1008, 'ARAVINDH', 'NAIR', 80000, '2020-10-11 11:00:00', 'DIGITAL MARKETING'),
(1009, 'IMANUEL', 'FERNATIZ', 90000, '2019-12-02 11:00:00', 'ACCOUNTANT');

SELECT * FROM worker;
```

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

Worker_Id	FirstName	LastName	Salary	JoiningDate	Department
1000	ADWIN	LOE	80000	2020-04-05 10:00:00	POWER BI ENGINEER
1001	AKSHARA	MANOJ	700000	2023-08-22 11:30:00	SOFTWARE ENGINEER
1002	ANNIE	SABASTEIN	60000	2024-02-17 10:00:00	DIGITAL MARKETING
1003	ZADHA	UWAIZ	70000	2023-10-28 12:00:00	DATA ANALYST
1004	ROSHAN	AMAN	90000	2019-06-09 10:00:00	BUSINESS ANALYST
1005	AZLAN	AZAAD	50000	2024-05-28 10:00:00	ACCOUNTANT
1006	ROHAN	ADITH	70000	2023-02-10 10:00:00	FINANCE
1007	DAVID	JOSEPH	90000	2019-03-17 12:00:00	HR MANAGER
1008	ARAVINDH	NAIR	80000	2020-10-11 11:00:00	DIGITAL MARKETING
1009	IMANUEL	FERNATIZ	90000	2019-12-02 11:00:00	ACCOUNTANT

-

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.

```
27    DELIMITER //
28    CREATE PROCEDURE D_Worker(
29        IN p_Worker_Id INT,
30        IN p_FirstName CHAR(25),
31        IN p_LastName CHAR(25),
32        IN p_Salary INT,
33        IN p_JoiningDate DATETIME,
34        IN p_Department CHAR(25))
35    BEGIN
36        INSERT INTO worker VALUES (p_Worker_Id, p_FirstName, p_LastName, p_Salary, p_JoiningDate, p_Department);
37    END //
38    DELIMITER ;
39    CALL D_Worker(1010,'ANADH','SUDEER',60000,'2024-06-11 10:00:00', 'SOFTWARE DEVELOPER');
40    SELECT * FROM worker;
```

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

Worker_Id	FirstName	LastName	Salary	JoiningDate	Department
1000	ADWIN	LOE	80000	2020-04-05 10:00:00	POWER BI ENGINEER
1001	AKSHARA	MANOJ	700000	2023-08-22 11:30:00	SOFTWARE ENGINEER
1002	ANNIE	SABASTEIN	60000	2024-02-17 10:00:00	DIGITAL MARKETING
1003	ZADHA	UWAIZ	70000	2023-10-28 12:00:00	DATA ANALYST
1004	ROSHAN	AMAN	90000	2019-06-09 10:00:00	BUSINESS ANALYST
1005	AZLAN	AZAAD	50000	2024-05-28 10:00:00	ACCOUNTANT
1006	ROHAN	ADITH	70000	2023-02-10 10:00:00	FINANCE
1007	DAVID	JOSEPH	90000	2019-03-17 12:00:00	HR MANAGER
1008	ARAVINDH	NAIR	80000	2020-10-11 11:00:00	DIGITAL MARKETING
1009	IMANUEL	FERNATIZ	90000	2019-12-02 11:00:00	ACCOUNTANT
1010	ANADH	SUDEER	60000	2024-06-11 10:00:00	SOFTWARE DEVELOPER

worker 12 ×

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.

```
43    DELIMITER //
44    CREATE PROCEDURE Sal(
45        IN p_Worker_Id INT,
46        OUT p_Salary INT )
47    BEGIN
48        SELECT Salary INTO p_Salary FROM worker
49        WHERE Worker_Id = p_Worker_Id;
50    END //
51    DELIMITER ;
52
53    CALL Sal (1001, @Sal);
54    SELECT @Sal;
```

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

@Sal
700000

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.

```
56    DELIMITER //
57    CREATE PROCEDURE Dep(
58        p_Worker_Id INT,
59        p_Department CHAR(25) )
60    BEGIN
61        UPDATE worker
62        SET Department = p_Department
63        WHERE Worker_Id = p_Worker_Id ;
64    END //
65    DELIMITER ;
66    CALL Dep (1006 , 'ACCOUNTANT');
67    SELECT * FROM worker;
68
```

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

Worker_Id	FirstName	LastName	Salary	JoiningDate	Department
1000	ADWIN	LOE	80000	2020-04-05 10:00:00	POWER BI ENGINEER
1001	AKSHARA	MANOJ	700000	2023-08-22 11:30:00	SOFTWARE ENGINEER
1002	ANNIE	SABASTEIN	60000	2024-02-17 10:00:00	DIGITAL MARKETING
1003	ZADHA	UWAIZ	70000	2023-10-28 12:00:00	DATA ANALYST
1004	ROSHAN	AMAN	90000	2019-06-09 10:00:00	BUSINESS ANALYST
1005	AZLAN	AZAAD	50000	2024-05-28 10:00:00	ACCOUNTANT
1006	ROHAN	ADITH	70000	2023-02-10 10:00:00	ACCOUNTANT
1007	DAVID	JOSEPH	90000	2019-03-17 12:00:00	HR MANAGER
1008	ARAVINDH	NAIR	80000	2020-10-11 11:00:00	DIGITAL MARKETING
1009	IMANUEL	FERNATIZ	90000	2019-12-02 11:00:00	ACCOUNTANT
1010	ANADH	SUDEER	60000	2024-06-11 10:00:00	SOFTWARE DEVELOPER

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.

```
74    DELIMITER //
75
76    CREATE PROCEDURE W_Count(
77        IN p_Department CHAR(25),
78        OUT p_WorkersCount INT)
79    BEGIN
80        SELECT COUNT(*) INTO p_WorkersCount FROM worker
81        WHERE Department = p_Department;
82    END //
83
84    DELIMITER ;
85
86    CALL W_Count('ACCOUNTANT', @W_count);
87    SELECT @W_count;
```

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

@W_count
3

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.

```
88    DELIMITER //
89    CREATE PROCEDURE Avg_sal(
90        IN p_Department CHAR(25),
91        OUT p_AvgSalary INT )
92    BEGIN
93        SELECT AVG(Salary) INTO p_AvgSalary FROM worker
94        WHERE Department = p_Department;
95    END //
96    DELIMITER ;
97
98    CALL Avg_sal('ACCOUNTANT', @AvgSal);
99    SELECT @AvgSal;
100
```

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

@AvgSal
70000