

-- Task :Stored Procedures and Functions

-- database

USE library_management_system;

-- Count books by category

DELIMITER //

CREATE PROCEDURE CountBooksInCategory(

 IN categoryName VARCHAR(50),

 OUT totalBooks INT

)

BEGIN

 -- Count books in the given category

 SELECT COUNT(*) INTO totalBooks

 FROM Books

 WHERE category = categoryName;

 -- Safety: if no books found, return 0

 IF totalBooks IS NULL THEN

 SET totalBooks = 0;

 END IF;

END //

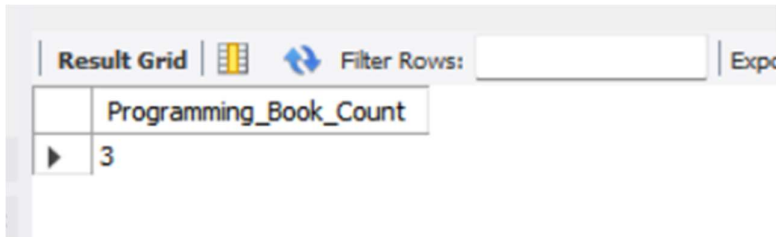
DELIMITER ;

-- Usage

```
CALL CountBooksInCategory('Programming', @count);
```

```
SELECT @count AS Programming_Book_Count;
```

Output:-



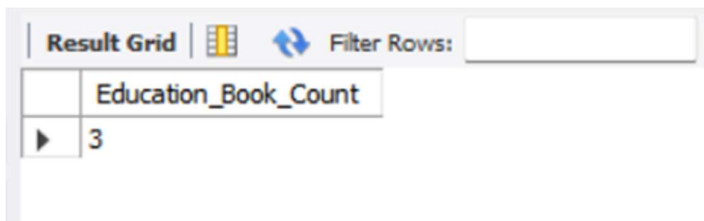
The screenshot shows a SQL Server Result Grid window. At the top, there is a tab labeled 'Result Grid' and a 'Filter Rows' section with a search icon and an empty text box. Below the header, the column name 'Programming_Book_Count' is displayed. A single row is visible, containing the value '3'.

	Programming_Book_Count
▶	3

```
CALL CountBooksInCategory('Education', @count);
```

```
SELECT @count AS Education_Book_Count;
```

Output:-



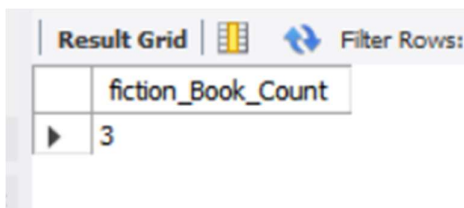
The screenshot shows a SQL Server Result Grid window. At the top, there is a tab labeled 'Result Grid' and a 'Filter Rows' section with a search icon and an empty text box. Below the header, the column name 'Education_Book_Count' is displayed. A single row is visible, containing the value '3'.

	Education_Book_Count
▶	3

```
CALL CountBooksInCategory('fiction', @count);
```

```
SELECT @count AS fiction_Book_Count;
```

Output:-



The screenshot shows a SQL Server Result Grid window. At the top, there is a tab labeled 'Result Grid' and a 'Filter Rows' section with a search icon and an empty text box. Below the header, the column name 'fiction_Book_Count' is displayed. A single row is visible, containing the value '3'.

	fiction_Book_Count
▶	3

```
-- Calculate Late Fee
```

```
DELIMITER //
```

```
CREATE FUNCTION CalculateLateFee(daysLate INT)
```

```
RETURNS DECIMAL(10,2)
```

```
DETERMINISTIC
```

```
BEGIN
```

```
    DECLARE fee DECIMAL(10,2);
```

```
    -- Rs. 2.50 per day late
```

```
    SET fee = daysLate * 2.50;
```

```
    RETURN fee;
```

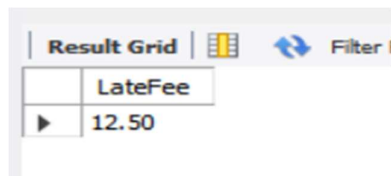
```
END //
```

```
DELIMITER ;
```

```
-- Usage
```

```
SELECT CalculateLateFee(5) AS LateFee; -- 5 days late
```

Output:-



	LateFee
▶	12.50

```
SELECT CalculateLateFee(0) AS LateFee; -- 0 days late
```

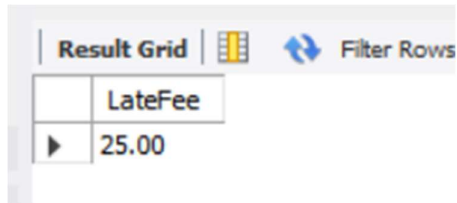
Output:-



	LateFee
▶	0.00

```
SELECT CalculateLateFee(10) AS LateFee; -- 10 days late
```

Output:-



	LateFee
▶	25.00

```
-- Get full name of a reader by readerId safely
```

```
DELIMITER //
```

```
CREATE FUNCTION GetReaderFullName(readerId INT)
```

```
RETURNS VARCHAR(100)
```

```
DETERMINISTIC
```

```
BEGIN
```

```
    DECLARE fullName VARCHAR(100);
```

```
    -- Combine first and last name
```

```
    SELECT CONCAT(first_name, ' ', last_name)
```

```
    INTO fullName
```

```
    FROM Readers
```

```
    WHERE reader_id = readerId;
```

```
    -- if reader is not existed so return messege 'Reader Not Found'
```

```
    IF fullName IS NULL THEN
```

```
        RETURN 'Reader Not Found';
```

```
    ELSE
```

```
        RETURN fullName;
```

```
    END IF;
```

```
END //
```

```
DELIMITER ;
```

-- Usage

```
SELECT GetReaderFullName(3) AS ReaderName;
```

Output:-

Result Grid		Filter Rows:
	ReaderName	
▶	Priyanka Nikam	