Assignment No. 2

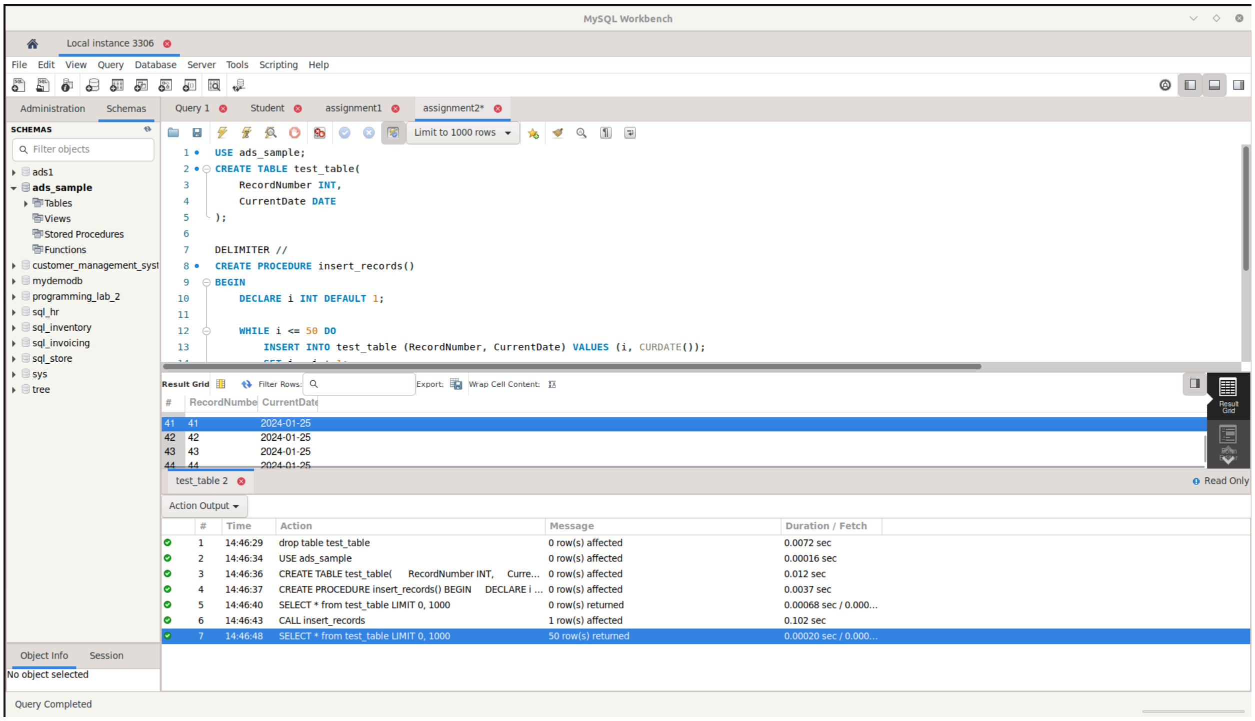
6CS371 : Advanced Database System Lab

Name : Jay Shirgupe
PRN: 21510026
Batch: T-7
TY CSE

I. MySQL / PSM Review:

MySQL PSM is an extension to the SQL standard, providing a way to create stored procedures, functions, and triggers within the MySQL database.

1. Create a table called test_table with 2 columns RecordNumber (type : Number(3)) and CurrentDate (type : Date)). Write a procedure in PSM which will insert 50 records into test_table. Insert the current date value into the table.

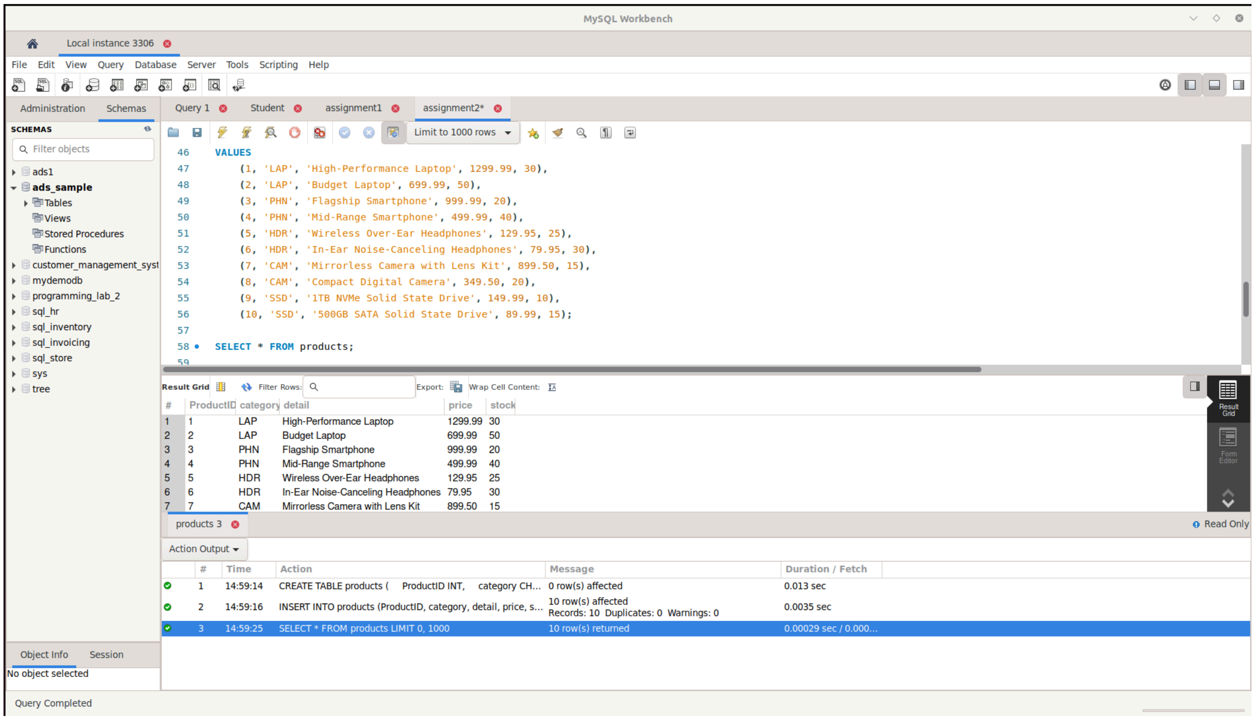


```sql
USE ads_sample;
CREATE TABLE test_table(
    RecordNumber INT,
    CurrentDate DATE
);

DELIMITER //
CREATE PROCEDURE insert_records()
BEGIN
    DECLARE i INT DEFAULT 1;

    WHILE i <= 50 DO
        INSERT INTO test_table (RecordNumber, CurrentDate) VALUES (i, CURDATE());
```

#	RecordNumbe	CurrentDate
41	41	2024-01-25
42	42	2024-01-25
43	43	2024-01-25
44	44	2024-01-25

	#	Time	Action	Message	Duration / Fetch
✓	1	14:46:29	drop table test_table	0 row(s) affected	0.0072 sec
✓	2	14:46:34	USE ads_sample	0 row(s) affected	0.00016 sec
✓	3	14:46:36	CREATE TABLE test_table(RecordNumber INT, Curre...	0 row(s) affected	0.012 sec
✓	4	14:46:37	CREATE PROCEDURE insert_records() BEGIN DECLARE i ...	0 row(s) affected	0.0037 sec
✓	5	14:46:40	SELECT * from test_table LIMIT 0, 1000	0 row(s) returned	0.00068 sec / 0.000...
✓	6	14:46:43	CALL insert_records	1 row(s) affected	0.102 sec
✓	7	14:46:48	SELECT * from test_table LIMIT 0, 1000	50 row(s) returned	0.00020 sec / 0.000...

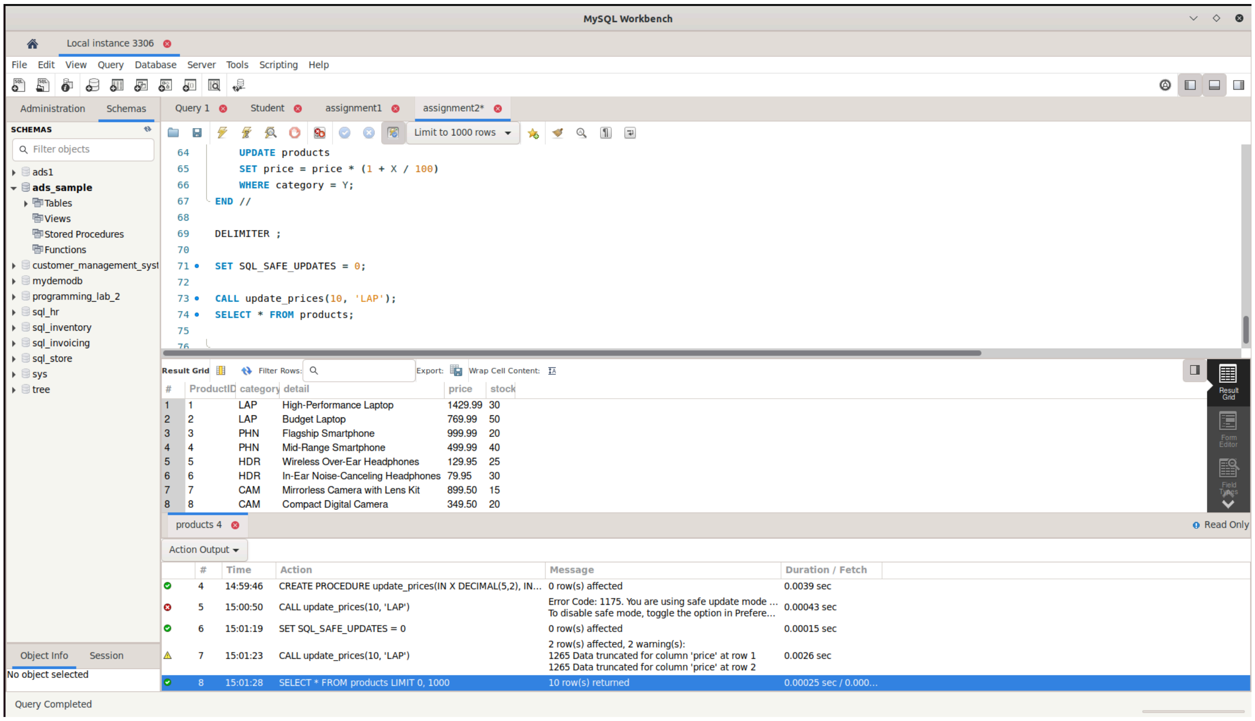
2. Create a products table products(ProductID number(4),category char(3),detail varchar2(30),price number(10,2),stock number(5)). Insert the sample data. Write PSM procedure with two arguments X & Y which will increase price by X% for all products in category Y. X and Y will be given by user.



```sql
VALUES
    (1, 'LAP', 'High-Performance Laptop', 1299.99, 30),
    (2, 'LAP', 'Budget Laptop', 699.99, 50),
    (3, 'PHN', 'Flagship Smartphone', 999.99, 20),
    (4, 'PHN', 'Mid-Range Smartphone', 499.99, 40),
    (5, 'HDR', 'Wireless Over-Ear Headphones', 129.95, 25),
    (6, 'HDR', 'In-Ear Noise-Canceling Headphones', 79.95, 30),
    (7, 'CAM', 'Mirrorless Camera with Lens Kit', 899.50, 15),
    (8, 'CAM', 'Compact Digital Camera', 349.50, 20),
    (9, 'SSD', '1TB NVMe Solid State Drive', 149.99, 10),
    (10, 'SSD', '500GB SATA Solid State Drive', 89.99, 15);

SELECT * FROM products;
```

#	ProductID	category	detail	price	stock
1	1	LAP	High-Performance Laptop	1299.99	30
2	2	LAP	Budget Laptop	699.99	50
3	3	PHN	Flagship Smartphone	999.99	20
4	4	PHN	Mid-Range Smartphone	499.99	40
5	5	HDR	Wireless Over-Ear Headphones	129.95	25
6	6	HDR	In-Ear Noise-Canceling Headphones	79.95	30
7	7	CAM	Mirrorless Camera with Lens Kit	899.50	15

	#	Time	Action	Message	Duration / Fetch
✓	1	14:59:14	CREATE TABLE products (ProductID INT, category CH...	0 row(s) affected	0.013 sec
✓	2	14:59:16	INSERT INTO products (ProductID, category, detail, price, s...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0	0.0035 sec
✓	3	14:59:25	SELECT * FROM products LIMIT 0, 1000	10 row(s) returned	0.00029 sec / 0.000...



```sql
    UPDATE products
    SET price = price * (1 + X / 100)
    WHERE category = Y;
END //

DELIMITER ;

SET SQL_SAFE_UPDATES = 0;

CALL update_prices(10, 'LAP');
SELECT * FROM products;
```

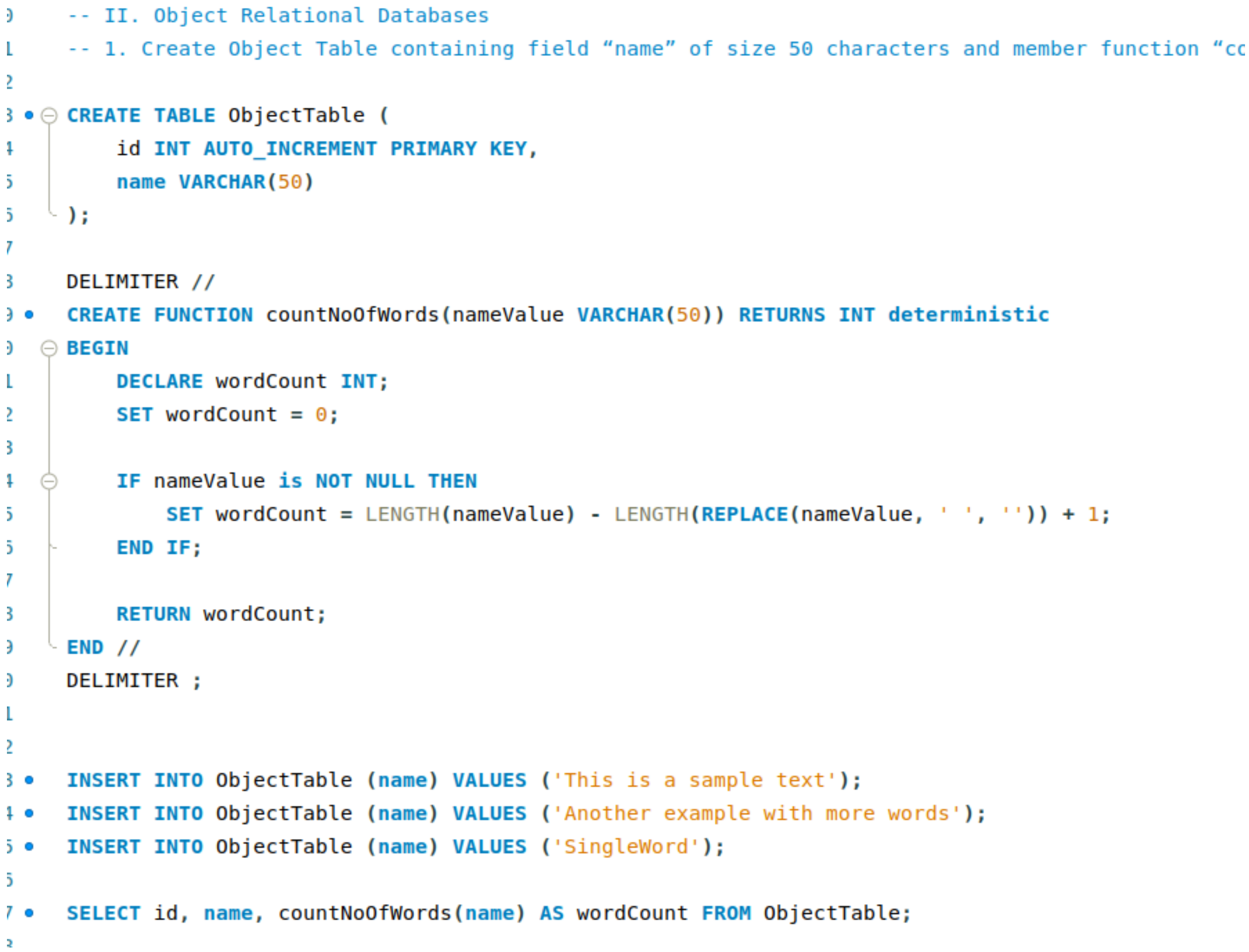
#	ProductID	category	detail	price	stock
1	1	LAP	High-Performance Laptop	1429.99	30
2	2	LAP	Budget Laptop	769.99	50
3	3	PHN	Flagship Smartphone	999.99	20
4	4	PHN	Mid-Range Smartphone	499.99	40
5	5	HDR	Wireless Over-Ear Headphones	129.95	25
6	6	HDR	In-Ear Noise-Canceling Headphones	79.95	30
7	7	CAM	Mirrorless Camera with Lens Kit	899.50	15
8	8	CAM	Compact Digital Camera	349.50	20

	#	Time	Action	Message	Duration / Fetch
✓	4	14:59:46	CREATE PROCEDURE update_prices(IN X DECIMAL(5,2), IN...	0 row(s) affected	0.0039 sec
✗	5	15:00:50	CALL update_prices(10, 'LAP')	Error Code: 1175. You are using safe update mode ... To disable safe mode, toggle the option in Prefere...	0.00043 sec
✓	6	15:01:19	SET SQL_SAFE_UPDATES = 0	0 row(s) affected	0.00015 sec
⚠	7	15:01:23	CALL update_prices(10, 'LAP')	2 row(s) affected, 2 warning(s): 1265 Data truncated for column 'price' at row 1 1265 Data truncated for column 'price' at row 2	0.0026 sec
✓	8	15:01:28	SELECT * FROM products LIMIT 0, 1000	10 row(s) returned	0.00025 sec / 0.000...

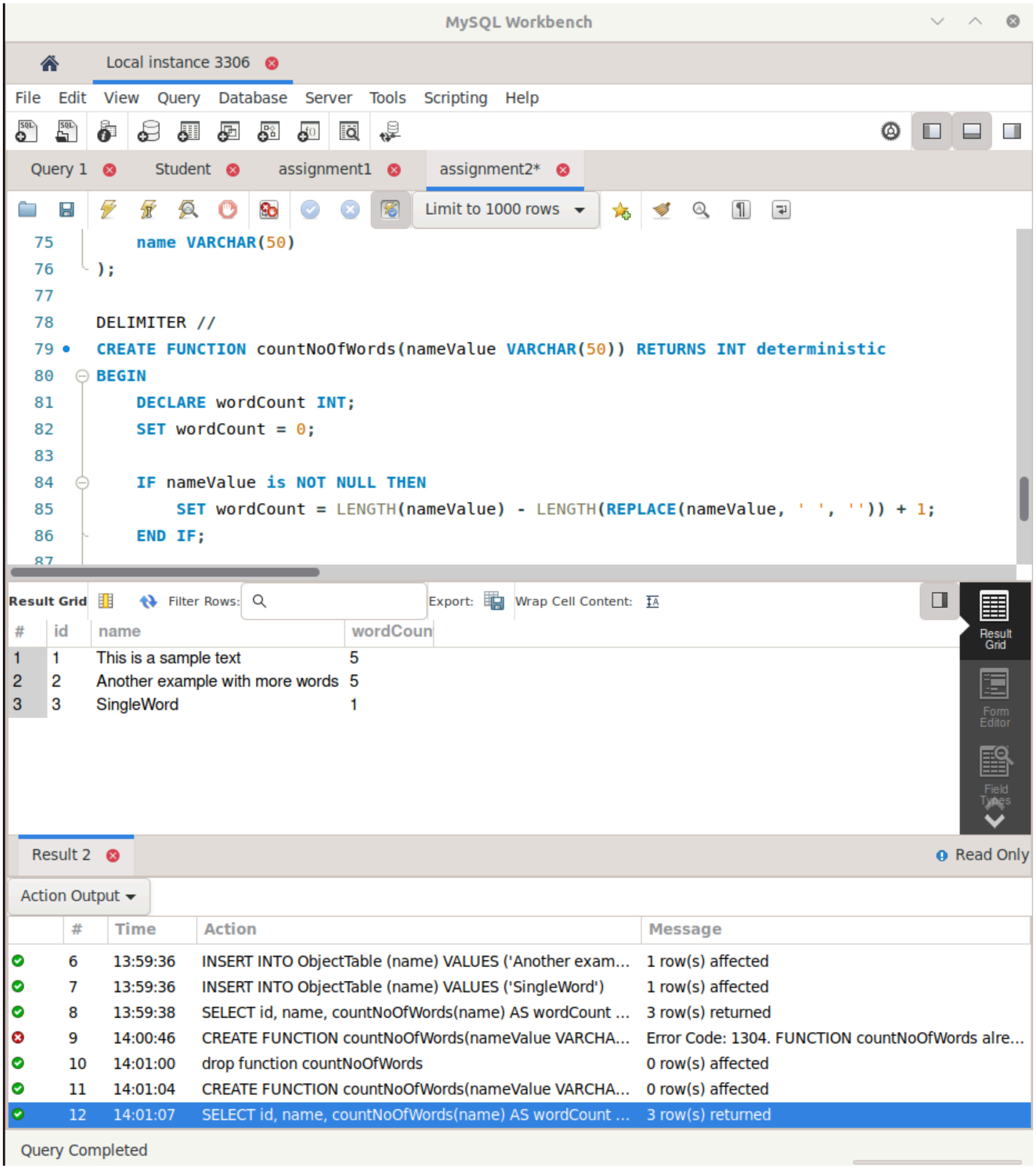
II. Object Relational Databases:

Object-Relational Databases (ORD) blend traditional relational database features with object-oriented principles. They extend the relational model to include complex data types and encapsulated behaviors. ORD systems support the storage of objects, allowing attributes and methods to be associated with data.

1. Create an Object Table containing a field “name” of size 50 characters and member function “countNoOfWords” which returns the no. of words in the “name” field. Demonstrate the working by entering different data.



```sql
-- II. Object Relational Databases
-- 1. Create Object Table containing field “name” of size 50 characters and member function “co

CREATE TABLE ObjectTable (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50)
);

DELIMITER //
CREATE FUNCTION countNoOfWords(nameValue VARCHAR(50)) RETURNS INT deterministic
BEGIN
    DECLARE wordCount INT;
    SET wordCount = 0;

    IF nameValue is NOT NULL THEN
        SET wordCount = LENGTH(nameValue) - LENGTH(REPLACE(nameValue, ' ', '')) + 1;
    END IF;

    RETURN wordCount;
END //
DELIMITER ;


INSERT INTO ObjectTable (name) VALUES ('This is a sample text');
INSERT INTO ObjectTable (name) VALUES ('Another example with more words');
INSERT INTO ObjectTable (name) VALUES ('SingleWord');

SELECT id, name, countNoOfWords(name) AS wordCount FROM ObjectTable;
```

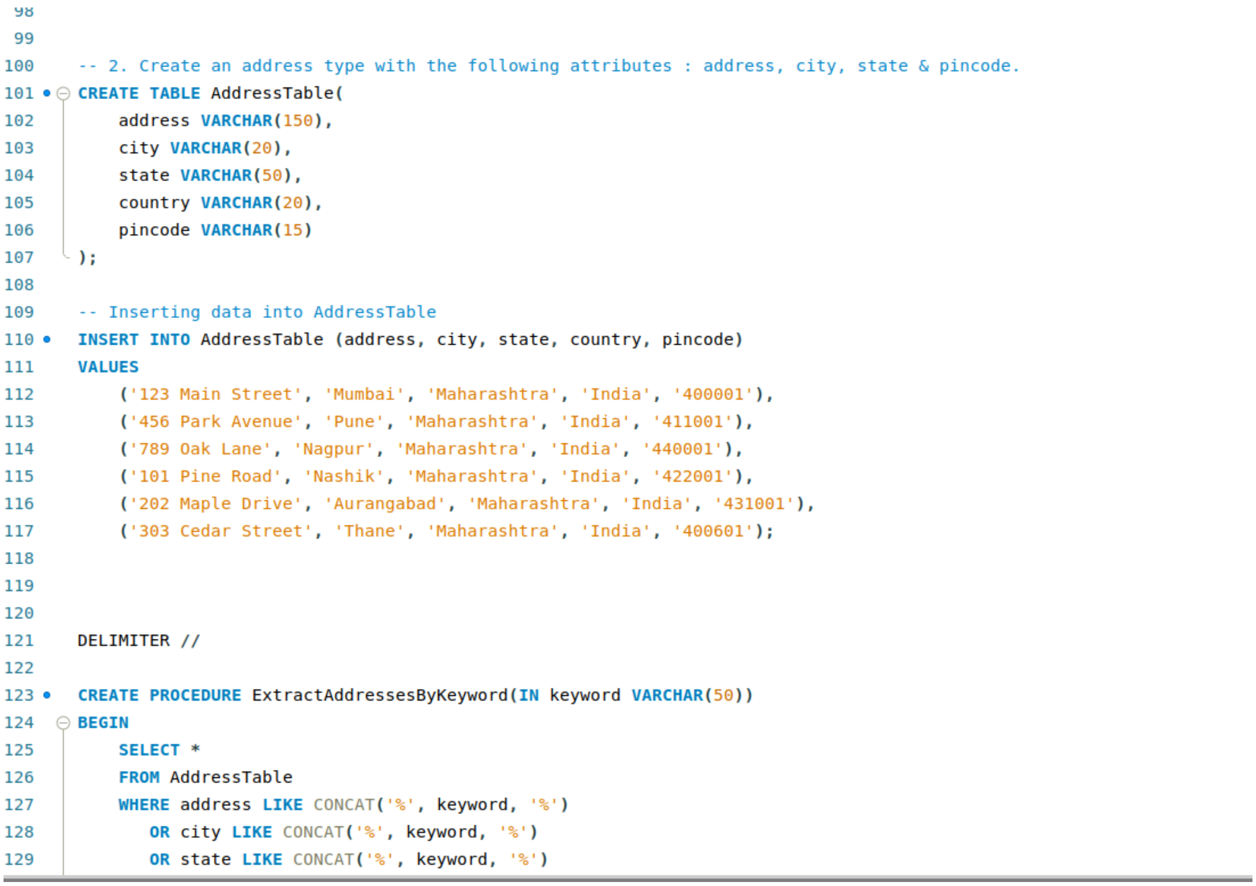


```
    name VARCHAR(50)
);

DELIMITER //
CREATE FUNCTION countNoOfWords(nameValue VARCHAR(50)) RETURNS INT deterministic
BEGIN
    DECLARE wordCount INT;
    SET wordCount = 0;

    IF nameValue is NOT NULL THEN
        SET wordCount = LENGTH(nameValue) - LENGTH(REPLACE(nameValue, ' ', '')) + 1;
    END IF;
```

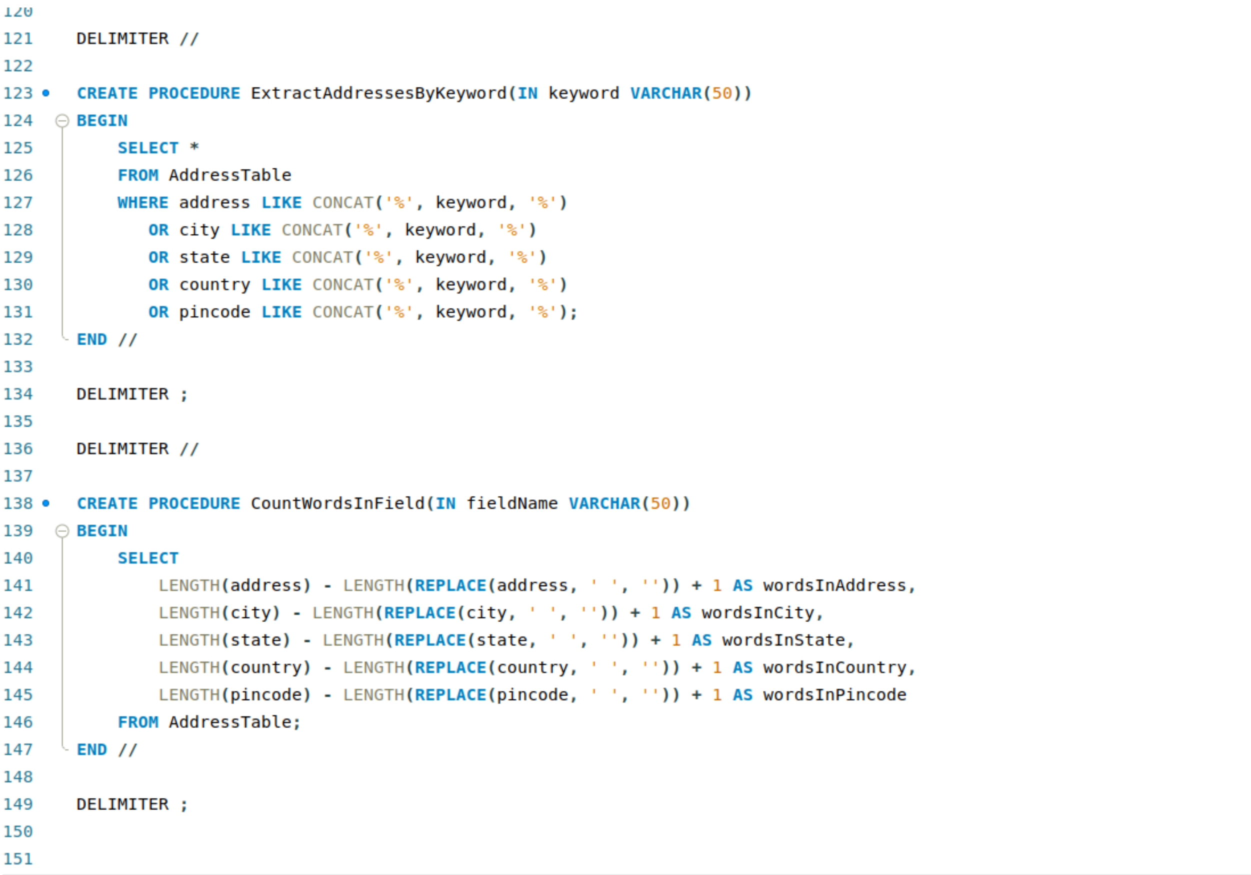
#	id	name	wordCoun
1	1	This is a sample text	5
2	2	Another example with more words	5
3	3	SingleWord	1

	#	Time	Action	Message
✓	6	13:59:36	INSERT INTO ObjectTable (name) VALUES ('Another exam...	1 row(s) affected
✓	7	13:59:36	INSERT INTO ObjectTable (name) VALUES ('SingleWord')	1 row(s) affected
✓	8	13:59:38	SELECT id, name, countNoOfWords(name) AS wordCount ...	3 row(s) returned
✗	9	14:00:46	CREATE FUNCTION countNoOfWords(nameValue VARCHA...	Error Code: 1304. FUNCTION countNoOfWords alre...
✓	10	14:01:00	drop function countNoOfWords	0 row(s) affected
✓	11	14:01:04	CREATE FUNCTION countNoOfWords(nameValue VARCHA...	0 row(s) affected
✓	12	14:01:07	SELECT id, name, countNoOfWords(name) AS wordCount ...	3 row(s) returned

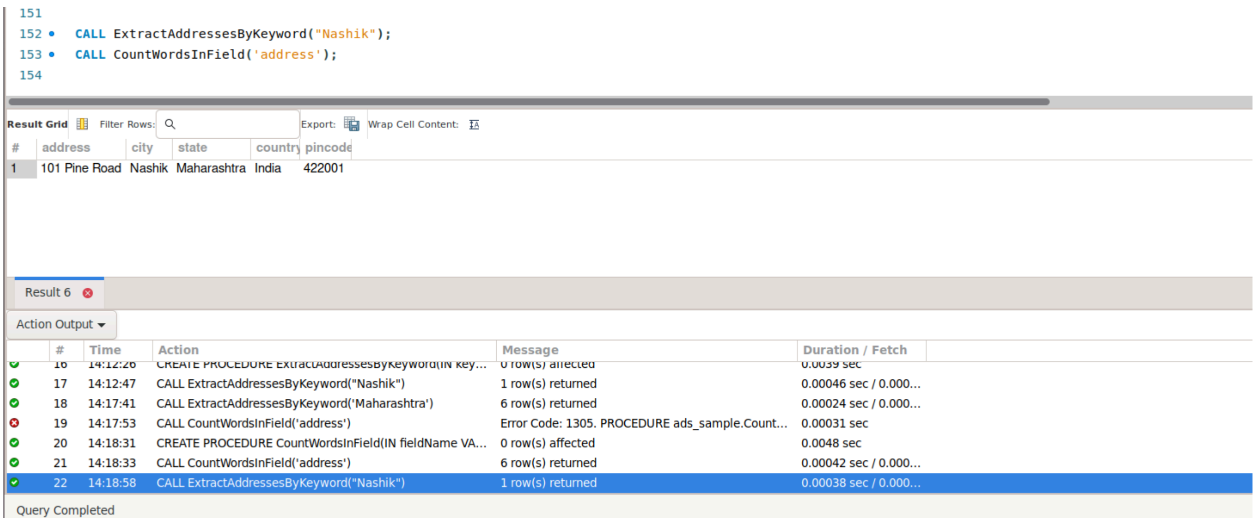
2. Create an address type with the following attributes : address, city, state & pincode. Include the following methods
    a. to extract the addresses based on given keyword.
    b. to return the no. of words in each given field (method should accept the name of attribute/field)



```sql


-- 2. Create an address type with the following attributes : address, city, state & pincode.
CREATE TABLE AddressTable(
    address VARCHAR(150),
    city VARCHAR(20),
    state VARCHAR(50),
    country VARCHAR(20),
    pincode VARCHAR(15)
);

-- Inserting data into AddressTable
INSERT INTO AddressTable (address, city, state, country, pincode)
VALUES
    ('123 Main Street', 'Mumbai', 'Maharashtra', 'India', '400001'),
    ('456 Park Avenue', 'Pune', 'Maharashtra', 'India', '411001'),
    ('789 Oak Lane', 'Nagpur', 'Maharashtra', 'India', '440001'),
    ('101 Pine Road', 'Nashik', 'Maharashtra', 'India', '422001'),
    ('202 Maple Drive', 'Aurangabad', 'Maharashtra', 'India', '431001'),
    ('303 Cedar Street', 'Thane', 'Maharashtra', 'India', '400601');



DELIMITER //

CREATE PROCEDURE ExtractAddressesByKeyword(IN keyword VARCHAR(50))
BEGIN
    SELECT *
    FROM AddressTable
    WHERE address LIKE CONCAT('%', keyword, '%')
       OR city LIKE CONCAT('%', keyword, '%')
       OR state LIKE CONCAT('%', keyword, '%')
```



```sql

DELIMITER //

CREATE PROCEDURE ExtractAddressesByKeyword(IN keyword VARCHAR(50))
BEGIN
    SELECT *
    FROM AddressTable
    WHERE address LIKE CONCAT('%', keyword, '%')
       OR city LIKE CONCAT('%', keyword, '%')
       OR state LIKE CONCAT('%', keyword, '%')
       OR country LIKE CONCAT('%', keyword, '%')
       OR pincode LIKE CONCAT('%', keyword, '%');
END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE CountWordsInField(IN fieldName VARCHAR(50))
BEGIN
    SELECT
        LENGTH(address) - LENGTH(REPLACE(address, ' ', '')) + 1 AS wordsInAddress,
        LENGTH(city) - LENGTH(REPLACE(city, ' ', '')) + 1 AS wordsInCity,
        LENGTH(state) - LENGTH(REPLACE(state, ' ', '')) + 1 AS wordsInState,
        LENGTH(country) - LENGTH(REPLACE(country, ' ', '')) + 1 AS wordsInCountry,
        LENGTH(pincode) - LENGTH(REPLACE(pincode, ' ', '')) + 1 AS wordsInPincode
    FROM AddressTable;
END //

DELIMITER ;

```



```sql

CALL ExtractAddressesByKeyword("Nashik");
CALL CountWordsInField('address');

```

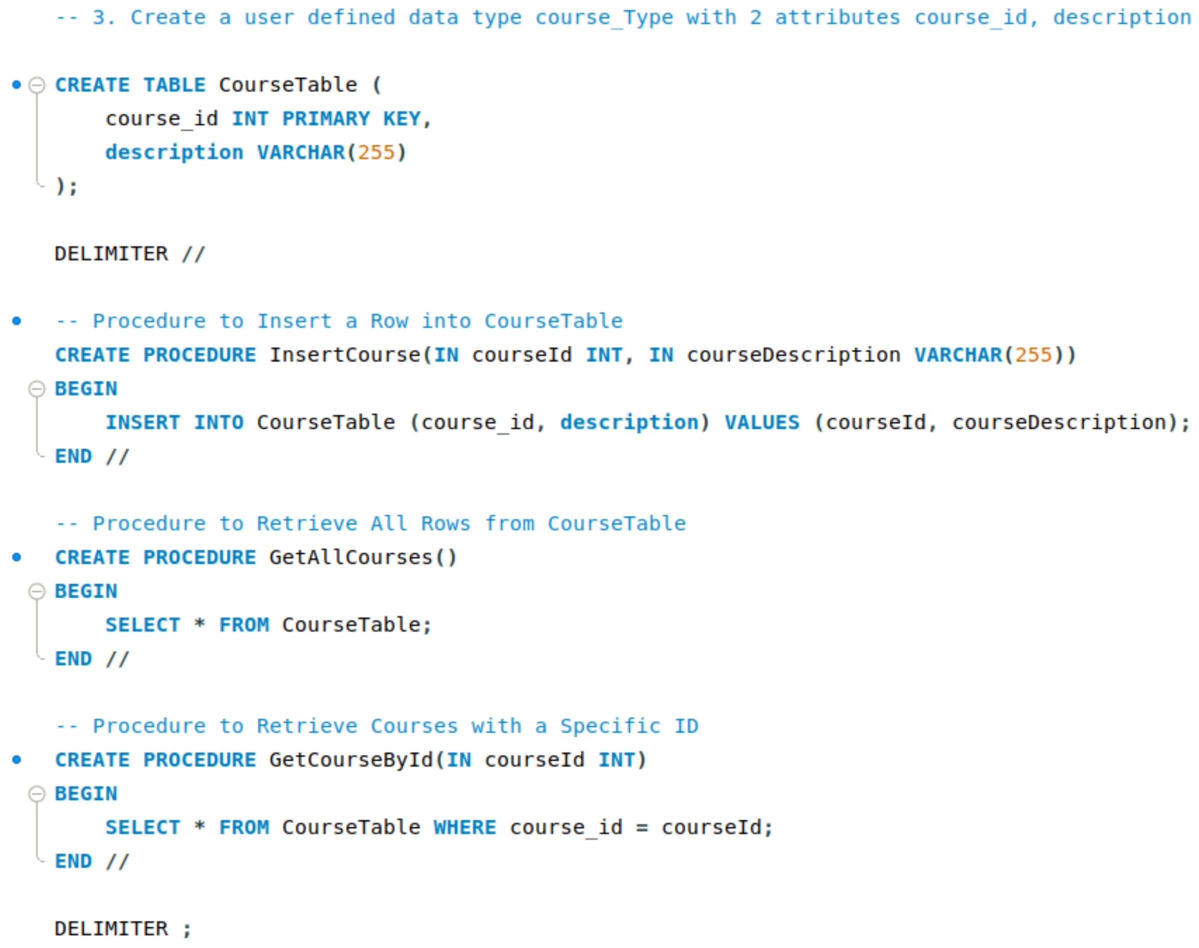
#	address	city	state	country	pincode
1	101 Pine Road	Nashik	Maharashtra	India	422001

	#	Time	Action	Message	Duration / Fetch
✓	16	14:12:26	CREATE PROCEDURE ExtractAddressesByKeyword(IN key...	0 row(s) affected	0.0039 sec
✓	17	14:12:47	CALL ExtractAddressesByKeyword("Nashik")	1 row(s) returned	0.00046 sec / 0.000...
✓	18	14:17:41	CALL ExtractAddressesByKeyword('Maharashtra')	6 row(s) returned	0.00024 sec / 0.000...
✗	19	14:17:53	CALL CountWordsInField('address')	Error Code: 1305. PROCEDURE ads_sample.Count...	0.00031 sec
✓	20	14:18:31	CREATE PROCEDURE CountWordsInField(IN fieldName VA...	0 row(s) affected	0.0048 sec
✓	21	14:18:33	CALL CountWordsInField('address')	6 row(s) returned	0.00042 sec / 0.000...
✓	22	14:18:58	CALL ExtractAddressesByKeyword("Nashik")	1 row(s) returned	0.00038 sec / 0.000...

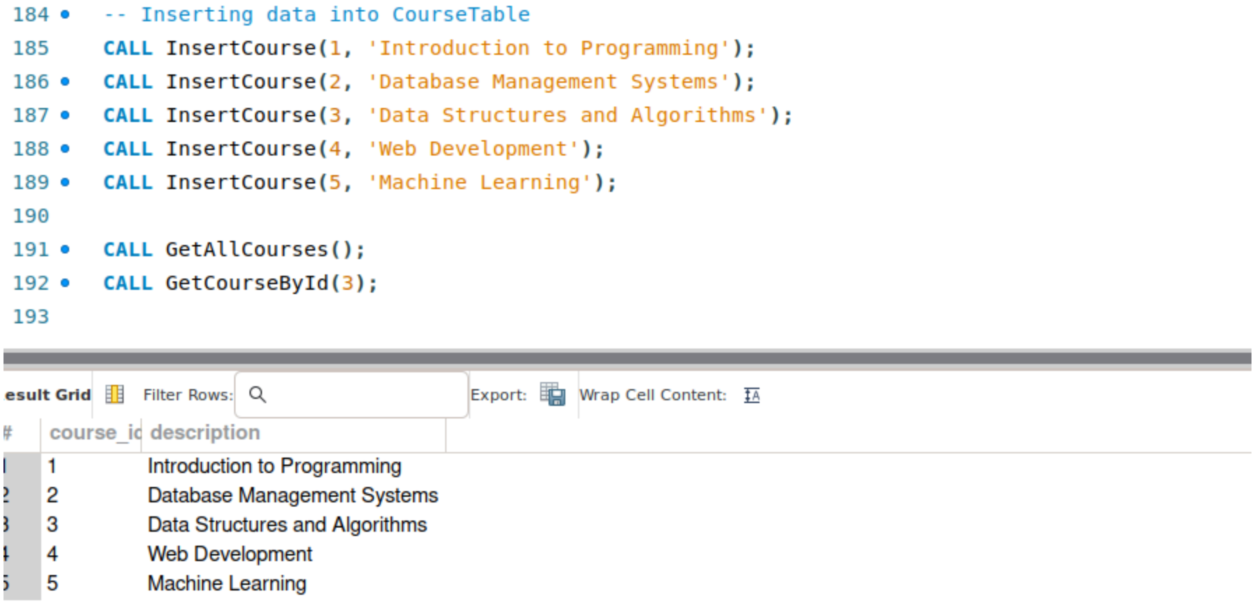
Query Completed

3. Create a user defined data type course_Type with 2 attributes course_id, description :
    a. Create an object table based on the type created.
    b. Insert rows into the table

Demonstrate the working with different data sets

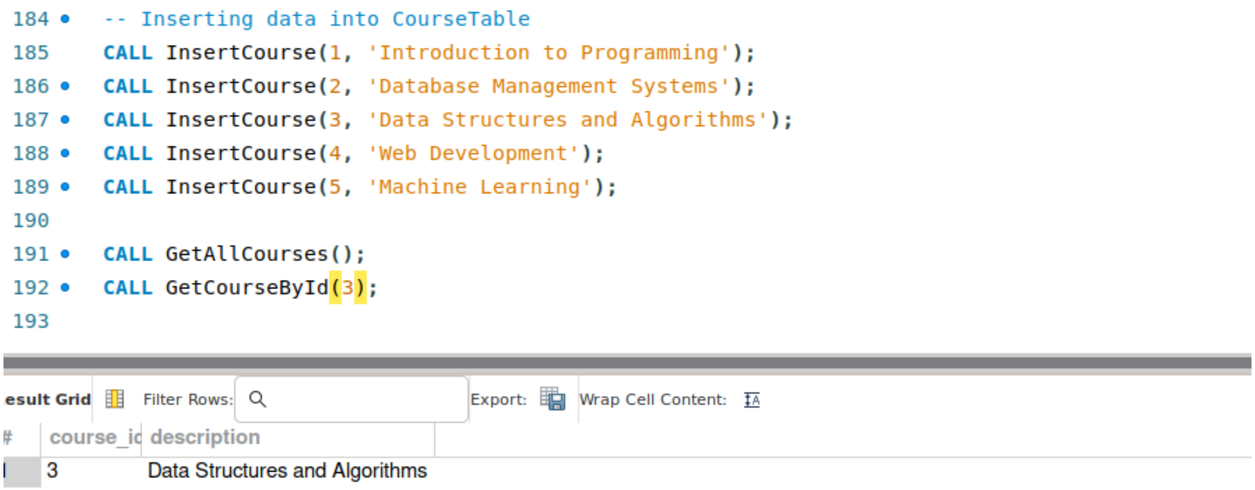


```sql
-- 3. Create a user defined data type course_Type with 2 attributes course_id, description

CREATE TABLE CourseTable (
    course_id INT PRIMARY KEY,
    description VARCHAR(255)
);

DELIMITER //

-- Procedure to Insert a Row into CourseTable
CREATE PROCEDURE InsertCourse(IN courseId INT, IN courseDescription VARCHAR(255))
BEGIN
    INSERT INTO CourseTable (course_id, description) VALUES (courseId, courseDescription);
END //

-- Procedure to Retrieve All Rows from CourseTable
CREATE PROCEDURE GetAllCourses()
BEGIN
    SELECT * FROM CourseTable;
END //

-- Procedure to Retrieve Courses with a Specific ID
CREATE PROCEDURE GetCourseById(IN courseId INT)
BEGIN
    SELECT * FROM CourseTable WHERE course_id = courseId;
END //

DELIMITER ;
```



```sql
-- Inserting data into CourseTable
CALL InsertCourse(1, 'Introduction to Programming');
CALL InsertCourse(2, 'Database Management Systems');
CALL InsertCourse(3, 'Data Structures and Algorithms');
CALL InsertCourse(4, 'Web Development');
CALL InsertCourse(5, 'Machine Learning');

CALL GetAllCourses();
CALL GetCourseById(3);

```

#	course_ic	description
1	1	Introduction to Programming
2	2	Database Management Systems
3	3	Data Structures and Algorithms
4	4	Web Development
5	5	Machine Learning



```sql
-- Inserting data into CourseTable
CALL InsertCourse(1, 'Introduction to Programming');
CALL InsertCourse(2, 'Database Management Systems');
CALL InsertCourse(3, 'Data Structures and Algorithms');
CALL InsertCourse(4, 'Web Development');
CALL InsertCourse(5, 'Machine Learning');

CALL GetAllCourses();
CALL GetCourseById(3);
```

#	course_ic	description
	3	Data Structures and Algorithms

Conclusion

In this assignment we studied about MySQL PSM and how to implement Object Relational Database in MySQL.