**Institute of Computer Technology**

**Ganpat University**

**(2CSE301) DATABASE MANAGEMENT SYSTEM**

**Practical 10 MySQL Stored Procedures**

1. Create a stored procedure that prints the following statement:

**hi**, **your name is *your\_name* and your enrolment number is *enrolment\_number***

DELIMITER //

create procedure PrintStudentInfo (in your\_name varchar(255), in enrolment\_number bigint )

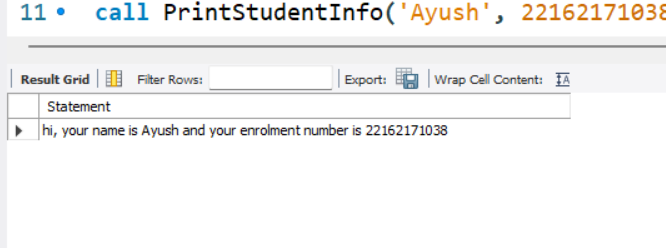
begin

select CONCAT('hi, your name is ', your\_name, ' and your enrolment number is ', enrolment\_number) as Statement;

end//

DELIMITER ;

call PrintStudentInfo('Ayush', 22162171038);



1. Create a stored procedure that displays the full name (e.g. PENELOPE GUINESS) of the first 10 actors.

DELIMITER //

CREATE PROCEDURE GetFirst10ActorNames()

BEGIN

SELECT CONCAT(first\_name, ' ', last\_name) AS full\_name

FROM actor

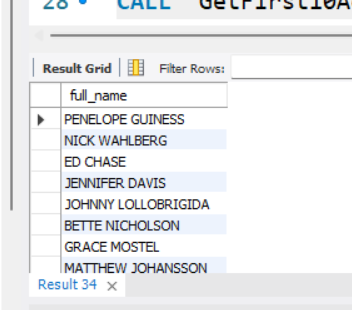
LIMIT 10;

-- WHERE actor\_id BETWEEN 1 AND 10;

END //

DELIMITER ;

CALL GetFirst10ActorNames();



1. Create a stored procedure that displays all the details for the given film title.

DELIMITER //

CREATE PROCEDURE GetFilmDetailsByTitle(IN film\_title VARCHAR(255))

BEGIN

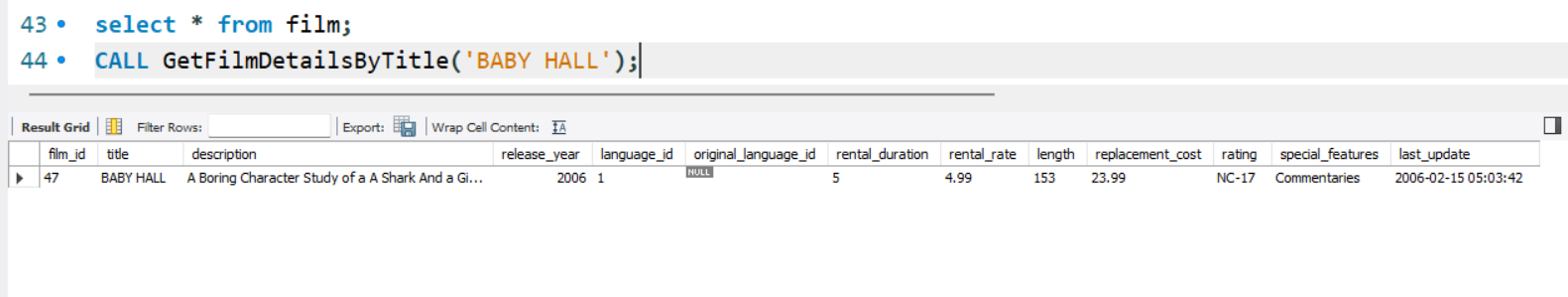
SELECT \*

FROM sakila.film

WHERE title = film\_title;

END//

DELIMITER ;



1. Create a stored procedure that displays all the details of the film whose name contains the given string in its name.

delimiter $$

CREATE procedure Q4(IN A VARCHAR(20))

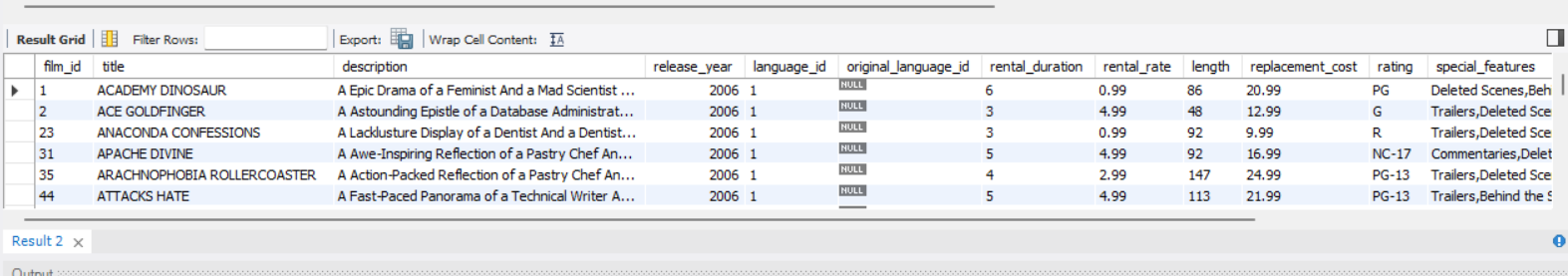
BEGIN

SELECT \* FROM film WHERE title LIKE (CONCAT('%',A,'%'));

END;

$$

call Q4("AC");



1. Create a stored procedure that displays all the films for the given category\_id.

delimiter $$

create procedure Q5(IN A numeric)

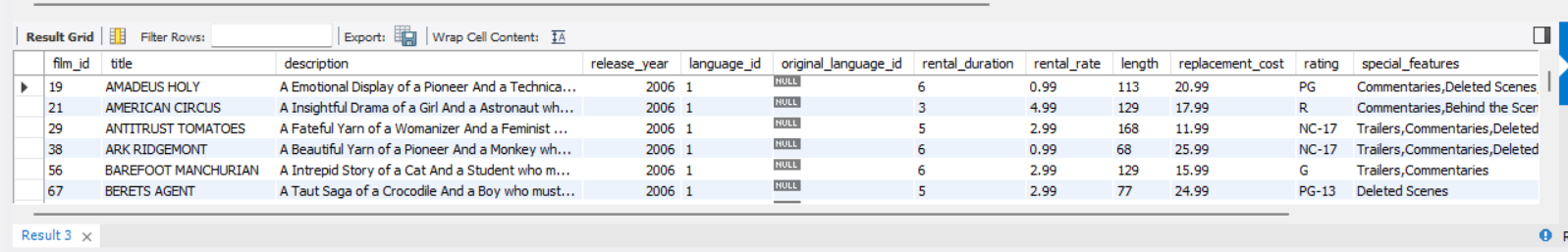
BEGIN

SELECT \* FROM film where film\_id IN (select film\_id from film\_category where category\_id = A);

END;

$$

call Q5('1');



1. Create a stored procedure that displays all the films for the given category name.

delimiter $$

CREATE procedure Q6(IN A VARCHAR(15))

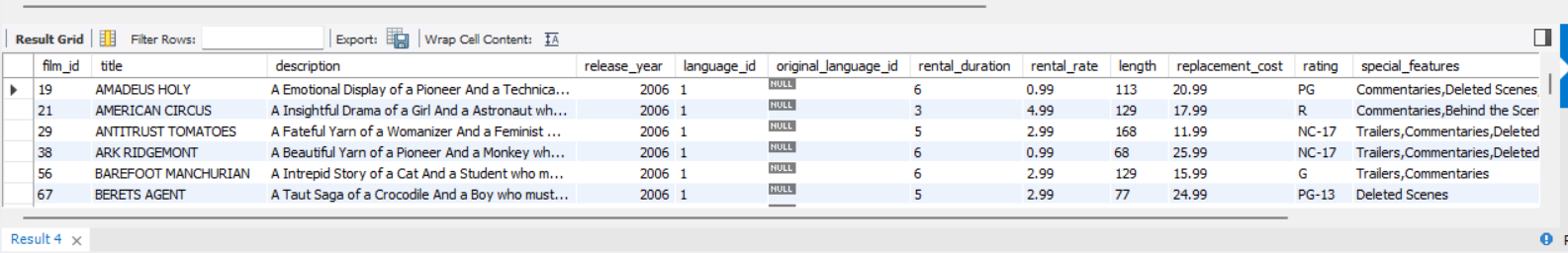
BEGIN

SELECT \* FROM film where film\_id IN (select film\_id from film\_category where category\_id IN (SELECT category\_id FROM category WHERE name = A));

END;

$$

call Q6("Action");



1. Create a procedure to display details of all the films whose rental\_rate falls between specified range.

delimiter $$

create procedure Q7(IN A numeric(10,2),IN B numeric(10,2))

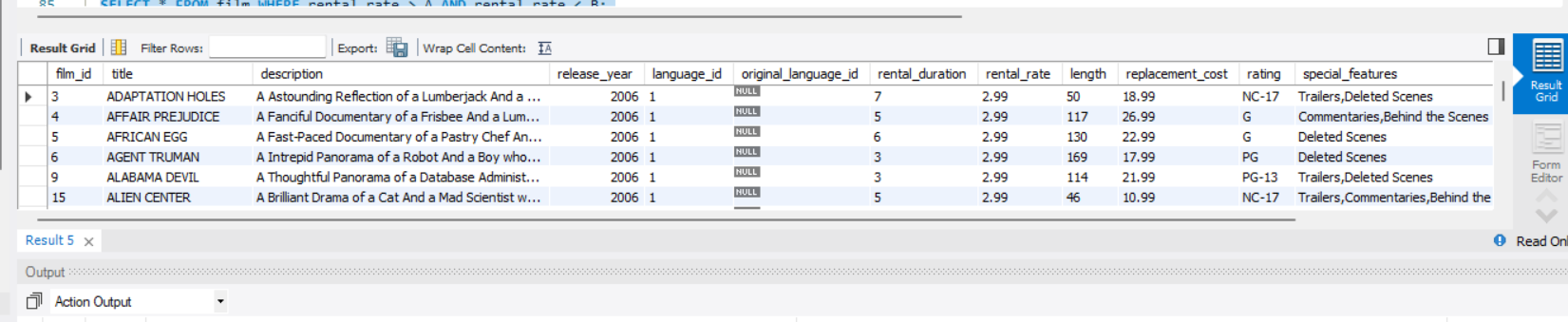
BEGIN

SELECT \* FROM film WHERE rental\_rate > A AND rental\_rate < B;

END;

$$

call Q7(1.99,3.99);



1. Create a stored procedure that displays the full name (e.g. PENELOPE GUINESS) of the actor for given actor id.

delimiter $$

create procedure Q8 (IN A numeric)

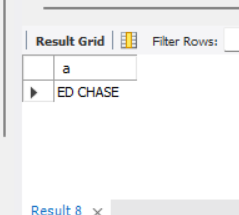
BEGIN

SELECT CONCAT(first\_name, " " , last\_name) AS a FROM actor WHERE actor\_id = A;

END;

$$

call Q8(3);



1. Create a procedure to display total number of films for the given category

delimiter $$

CREATE procedure Q9(IN A VARCHAR(15))

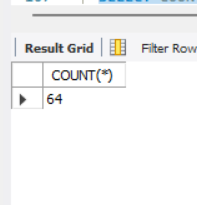
BEGIN

SELECT COUNT(\*) FROM film where film\_id IN (select film\_id from film\_category where category\_id IN (SELECT category\_id FROM category WHERE name = A));

END;

$$

call Q9("Action");



1. Create a procedure to display the total number of cities for the countries in which number of cities are more than the given number.

delimiter $$

CREATE procedure Q10(IN A numeric)

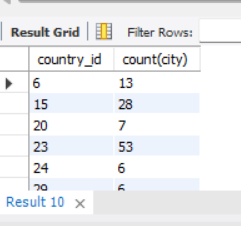
BEGIN

select country\_id,count(city) from city group by country\_id HAVING count(city)>A;

END;

$$

CALL Q10(5);



1. Create a stored procedure to Display total number of customers as per the active status given as an input

delimiter $$

CREATE procedure Q11(IN A numeric)

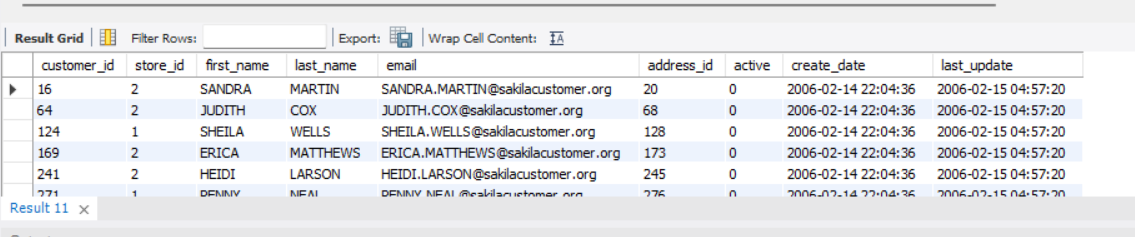
BEGIN

SELECT \* FROM customer WHERE active=A;

END;

$$

call Q11(0);



1. Create a stored procedure to update an actor’s first name with given name and actor id. Also display the updated name in the same procedure.

delimiter $$

CREATE procedure Q12(IN A NUMERIC,IN B VARCHAR(15))

BEGIN

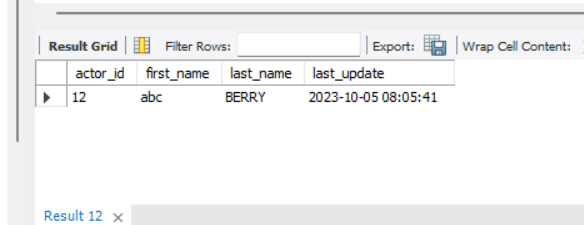
update actor set first\_name = B where actor\_id = A;

SELECT \* FROM actor WHERE actor\_id = A;

END;

$$

call Q12(12,"abc");



1. Display rental duration for the given film id (both film id and rental duration should be accessible from a single variable)

delimiter $$

create procedure Q13(INOUT A NUMERIC)

BEGIN

SELECT rental\_duration into A FROM film where film\_id = A;

END;

$$

set @film = 11;

call Q13(@film);

select @film;

