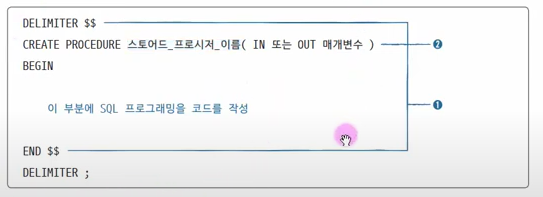
# 스토어드 프로시저

- $$는 1개만 사용해도 되지만 명확하게 표시하기 위해 2개를 사용함

- ##, %%, &&, // 등으로 바꿔도 됨



- CREATE PROCEDURE는 스토어드 프로시저를 만든 것뿐, 실행(호출)을 해야 사용할 수 있음

----> CALL 스토어드\_프로시저\_이름();

\* 프로시저 삭제

DROP PROCEDURE user\_proc;

1. 입력 매개변수의 사용



Ex)

USE market\_db;

DROP PROCEDURE IF EXISTS user\_proc1;

DELIMITER $$

CREATE PROCEDURE user\_proc1(IN userName VARCHAR(10))

BEGIN

SELECT \* FROM member WHERE mem\_name = userName;

END $$

DELIMITER ;

CALL user\_proc1('에이핑크');



Ex2)

DROP PROCEDURE IF EXISTS user\_proc2;

DELIMITER $$

CREATE PROCEDURE user\_proc2(

IN userNumber INT,

IN userHeight INT )

BEGIN

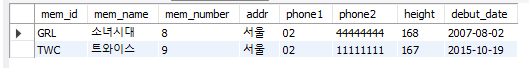
SELECT \* FROM member

WHERE mem\_number > userNumber AND height > userHeight;

END $$

DELIMITER ;

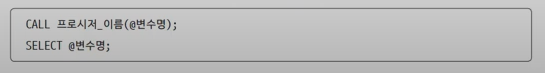
CALL user\_proc2(6, 165);



2. 출력 매개변수의 사용



3. 출력 매개변수가 있는 스토어드 프로시저 실행



Ex)

DROP PROCEDURE IF EXISTS user\_proc3;

DELIMITER $$

CREATE PROCEDURE user\_proc3(

IN txtValue CHAR(10),

OUT outValue INT )

BEGIN

INSERT INTO noTable VALUES(NULL,txtValue);

SELECT MAX(id) INTO outValue FROM noTable;

END $$

DELIMITER ;

DESC noTable;

CREATE TABLE IF NOT EXISTS noTable( // 테이블 조회 후 없어서 생성

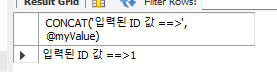
id INT AUTO\_INCREMENT PRIMARY KEY,

txt CHAR(10)

);

CALL user\_proc3 ('테스트1', @outValue); // 프로시저 안에서 변수를 선언할때는 DECLARE를 사용해서 선언하지만 그 외에는 @를 붙여서 변수를 선언함

SELECT CONCAT('입력된 ID 값 ==>', @outValue);

 -> 프로시저를 호출해서 값을 넣을수록 출력되는 숫자가 커짐

Ex2)

DROP PROCEDURE IF EXISTS ifelse\_proc;

DELIMITER $$

CREATE PROCEDURE ifelse\_proc(

IN memName VARCHAR(10)

)

BEGIN

DECLARE debutYear INT; -- 변수 선언

SELECT YEAR(debut\_date) into debutYear FROM member

WHERE mem\_name = memName;

IF (debutYear >= 2015) THEN

SELECT '신인 가수네요. 화이팅 하세요.' AS '메시지';

ELSE

SELECT '고참 가수네요. 그동안 수고하셨어요.'AS '메시지';

END IF;

END $$

DELIMITER ;

CALL ifelse\_proc ('오마이걸');



Ex3)

DELIMITER $$

CREATE PROCEDURE while\_proc()

BEGIN

DECLARE hap INT; -- 합계

DECLARE num INT; -- 1부터 100까지 증가

SET hap = 0; -- 합계 초기화

SET num = 1;

WHILE (num <= 100) DO -- 100까지 반복.

SET hap = hap + num;

SET num = num + 1; -- 숫자 증가

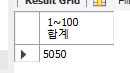
END WHILE;

SELECT hap AS '1~100 합계';

END $$

DELIMITER ;

CALL while\_proc();



Ex4)

DROP PROCEDURE IF EXISTS dynamic\_proc;

DELIMITER $$

CREATE PROCEDURE dynamic\_proc(

IN tableName VARCHAR(20)

)

BEGIN

SET @sqlQuery = CONCAT('SELECT \* FROM ', tableName);

PREPARE myQuery FROM @sqlQuery;

EXECUTE myQuery;

DEALLOCATE PREPARE myQuery; // 할당 해제를 하기 위해 사용

END $$

DELIMITER ;

CALL dynamic\_proc ('buy');

