

SQL 활용

저장 프로시저와 사용자 정의 함수

학습내용

- 프로시저
- 사용자 정의 함수

학습목표

- 프로시저의 개념을 이해하고 프로시저의
 매개변수를 활용하여 프로시저를 작성할 수 있다.
- 사용자 정의 함수와 프로시저의 차이점을 알고, 테이블을 반환하는 함수를 작성할 수 있다.

1. 프로시저의 개념

◆ 프로시저

- 자주 자용되는 질의문들을 하나로 묶어서 저장해두고 필요할 때마다 명령문처럼 실행할 수 있도록 해주는 것
 - 선택적으로 매개변수를 받아 일련의 질의문을 실행시켜 결과를 돌려주는 것
 - 범용 언어의 프로시저 ⇒ 함수와 유사한 개념

◆ 일반 질의문과의 차이점

- 일반 질의문
- 사용자 또는 응용 프로그램이 실행하고자 하는 SQL문을 DBMS에 전송하고 그 결과를 받음
- 대량의 복잡한 질의문들이 반복적으로 입력되면 그만큼 시스템에 부담이 됨
- DBMS에도 처리해야 하는 일이 늘어남
- 프로시저
- 프로시저 내용은 DBMS에 포함되어 있고 실행 방안도 미리 작성되어 있음

◆ 프로시저 생성 구문

CREATE [PROCEDURE|PROC] 프로시저이름 AS

BEGIN SQL문 END //BEGIN END는 SQL문이 하나만 있다면 생략 가능

1. 프로시저의 개념

◆ 프로시저 실행 문법

EXEC 프로시저이름

◆ 프로시저 수정 문법

ALTER PROCEDURE 프로시저이름 AS SQL문

DROP PROCEDURE 프로시저이름

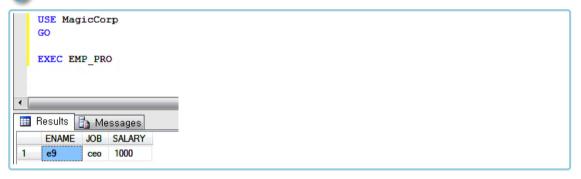
- 1. 프로시저의 개념
 - ◆ 단순 프로시저 생성 및 호출
 - ◎ MagicCorp DB에 employ 테이블에서 사번이 109인 사원의 이름, 직급, 급여를 검색하는 emp_pro 프로시저 생성하기

```
USE MagicCorp
GO

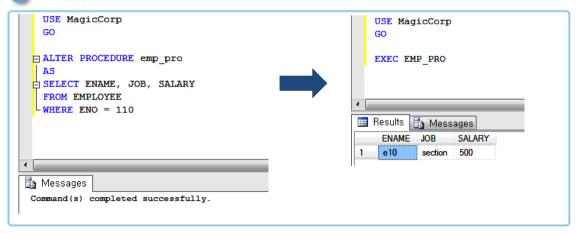
CREATE PROCEDURE emp_pro
AS
SELECT ENAME, JOB, SALARY
FROM EMPLOYEE
WHERE ENO = 109

Messages
Command(s) completed successfully.
```

Q emp_pro 호출하기



② emp_pro 프로시저를 수정하여 사원번호 110번의 정보 검색하기



2. 프로시저의 매개변수

◆ 매개 변수

- 프로시저 실행 시 조건값 등을 변경 할 수 없을까?
- 사원번호 109번에 대한 정보를 추출하는 저장 프로시저를 생성하고 이를 110번에 대한 정보를 추출하도록 저장 프로시저 변경
 - ⇒ 저장 프로시저 수행 시 사원번호를 입력으로 주어 해당 사원정보를 추출하도록 할 수 없을까?



매개 변수를 사용함

저장 프로시저 수행 시 수행 질의문에 특정 값을 매개변수로 전달할 수
 있도록 하여 다양한 조건을 하나의 질의문으로 수행할 수 있도록 지원해 줌

◆ 입력 매개변수의 선언

● 생성

CREATE PROCEDURE 프로시저이름 @매개변수명 타입, ··· AS SQL문

● 실행

EXEC 프로시저이름 매개변수값

2. 프로시저의 매개변수

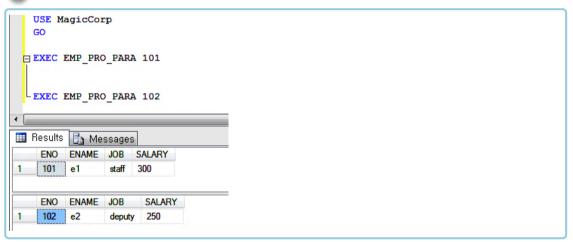
- ◆ 입력 매개변수 프로시저 예제
 - MagicCorp DB에 employ 테이블에서 임의의 사번을 입력 받아 해당 사원의 사번, 이름, 직급, 급여를 검색하는 emp_pro_para 프로시저 생성하기

```
USE MagicCorp
GO

CREATE PROCEDURE emp_pro_para
Genumber int
AS
SELECT ENO, ENAME, JOB, SALARY
FROM EMPLOYEE
WHERE ENO = Genumber

Messages
Command(s) completed successfully.
```

🔾 사원번호 101, 102등 임의 사원번호를 입력하여 emp_pro_para 수행하기



2. 프로시저의 매개변수

- ◆ 출력 매개변수란?
 - 입력 매개변수와 반대로 프로시저의 처리 결과값을 반환하는 매개변수

◆ 출력 매개변수 선언

CREATE PROCEDURE 프로시저이름 @매개변수명 타입 OUTPUT, … AS SELECT @매개변수명= 속성명 FROM … WHERE…

◆ 출력 매개변수

- 출력 매개변수 값 받기
- 프로시저 실행 전에 매개변수를 선언함(DECLARE 문 이용)
- 선언된 매개변수를 출력함(SELECT 문 이용)
- 프로시저 실행 전에 매개변수를 선언함(DECLARE 문 이용)

DECLARE @매개변수명

■ 프로시저 실행

EXEC 프로시저명 @매개변수명 OUTPUT

■ 선언된 매개변수를 출력함(SELECT 문 이용)

SELECT @매개변수명

2. 프로시저의 매개변수

- ◆ 출력 매개변수 프로시저 예제
 - 사원 테이블에서 부서번호인 사원들의 평균급여를 알아내는 프로시저 emp_out_put_para 작성하기

```
USE MagicCorp
GO

CREATE PROC emp_out_put_para
Odid int,
Oavg_sal int OUTPUT
AS
SELECT Oavg_sal= AVG(salary)
FROM EMPLOYEE
WHERE EMPLOYEE.DNO = Odid
```

- omp out put para를 호출하여 그 결과 출력하기
 - 호출 전 출력 매개변수 값을 받을 변수 선언 필요



1. 프로시저와의 차이점

- ♦ 사용자 정의 함수
 - 프로시저와 달리 RETURN문을 이용하여 하나의 값을 반드시 반환해야 함

◆ 프로시저

- RETURN문을 이용하여 값을 반환함
 - ⇒ 프로시저의 반환 값은 int 형만 가능하고 프로시저의 결과가 아닌 상태값 만을 반환함

◆ 사용자 정의 함수의 특징

- ① 사용자 정의 함수는 프로시저와 달리 다양한 타입의 값을 반환할 수 있음
- ② 함수이므로 질의문 내에서 사용이 가능함

◆ 사용자 정의 함수의 선언

● 기본적으로 프로시저와 유사함

```
CREATE FUNCTION 함수명
(@매개변수명 타입, …)
RETURNS 반환타입
AS
[BEGIN] SQL 문 [END]
```

- 주의점
- SQL문 내에는 반드시 RETURN문이 있어야 함

- 1. 프로시저와의 차이점
 - ◆ 사용자 정의 함수의 예
 - 임의의 부서 번호를 입력하면 해당 부서 사원들의 최대 급여를 반환하는 함수 MAX SAL 생성하기

```
USE MagicCorp GO

CREATE FUNCTION MAX_SAL
(@param int)
RETURNS int
AS
BEGIN
DECLARE @MAX_VAL int

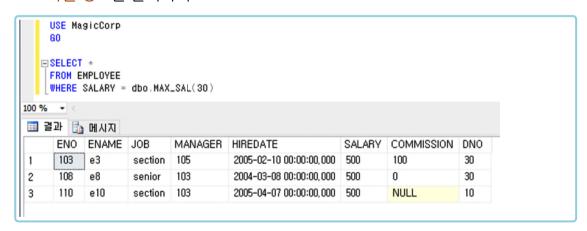
SELECT @MAX_VAL = MAX(salary)
FROM EMPLOYEE
WHERE DNO = @param

RETURN @MAX_VAL
END

100 % 

GÖO 완료되었습니다.
```

MAX_SAL 함수를 이용하여 부서번호 30의 최대 급여와 동일한 급여를 받는 사원 정보를 출력하기



■ 사용자 정의 함수 호출 시에는 반드시 접두사로 "dbo."을 넣어야 함

2. 테이블 반환 함수

- ◆ 테이블 반환 함수
 - 함수 결과로 테이블을 반환하는 함수

◆ 프로시저 + 뷰

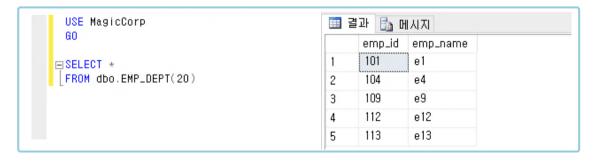
- ◆ 테이블 반환 함수의 선언

CREATE FUNCTION 함수명 (@매개변수명 타입, ….) RETURNS @반환변수 TABLE (테이블정의) AS [BEGIN] SQL 문 [END]

- 2. 테이블 반화 함수
 - ◆ 테이블 반환 함수의 예
 - 부서번호를 입력 받아 해당 부서 사원들의 사원번호와 이름을 반환하는 사용자 정의 함수 EMP_DEPT 생성하기

```
USE MagicCorp
    GO
   □ CREATE FUNCTION EMP_DEPT
    (Oparam int)
    RETURNS @emp_dept_table TABLE (
        emp_id int,
        emp_name varchar(20)
    AS
    BEGIN
    INSERT INTO @emp_dept_table
    SELECT EMPLOYEE.ENO, EMPLOYEE.ENAME
    FROM EMPLOYEE
    WHERE DNO = @param
    RETURN
    END
100 % ▼ <
다시면 🚮
   명령이 완료되었습니다.
```

사용자 정의 함수 EMP_DEPT를 이용하여 20번 부서의 사원번호, 사원명 출력하기



1. 프로시저

- 프로시저의 개념
 - 자주 자용되는 질의문들을 하나로 묶어서 저장해두고 필요할 때 마다 명령문처럼 실행할 수 있도록 해주는 개념
 - 선택적으로 매개변수를 받아 일련의 질의문을 수행해서 결과를 돌려주는 것
 - 범용 언어의 프로시저 / 함수와 유사한 개념임
 - 일반 질의문과의 차이점
 - ① 일반 질의문
 - 사용자 또는 응용 프로그램이 실행하고자 하는 SQL문을 DBMS에 전송하고 그 결과를 받음
 - 대량의 복잡한 질의문들이 반복적으로 입력되면 그만큼 시스템에 부담이 됨
 - DBMS에도 처리해야 하는 일이 늘어나 시스템에 더 큰 부담이 됨
 - ② 프로시저
 - 프로시저 내용은 DBMS에 포함되어 있고 실행 방안도 미리 작성되어 있음
 - 사용자나 응용 프로그램은 프로시저 이름과 매개변수 값(필요 시)만을 전송하면 ⇒ 복잡한 SQL문의 단순화

1. 프로시저

- 프로시저의 개념
 - 프로시저 생성 구문

CREATE [PROCEDURE|PROC] 프로시저이름 AS BEGIN SQL문 END //BEGIN END는 SQL문이 하나만 있다면 생략 가능

■ 프로시저 실행 문법 EXEC 프로시저이름

■ 프로시저 수정 문법

ALTER PROCEDURE 프로시저이름 AS SQL문

DROP PROCEDURE 프로시저이름

1. 프로시저

- 프로시저의 매개변수
 - 프로시저 실행 시 조건값 등을 변경 할 수 없을까?
 - 사원번호 109번에 대한 정보를 추출하는 저장 프로시저를 생성하고 이를 110번에 대한 정보를 추출하도록 저장 프로시저 변경
 - 저장 프로시저 수행 시 사원번호를 입력으로 주어 해당 사원정보를 추출하도록 할 수 없을까?
 - ⇒ 매개 변수를 사용함
 - 저장 프로시저 수행 시 수행 질의문에 특정 값을 매개변수로 전달할 수 있도록 하여 다양한 조건을 하나의 질의문으로 수행할 수 있도록 지원해 줌
 - 입력 매개변수의 선언
 - 생성

CREATE PROCEDURE 프로시저이름 @매개변수명 타입, … AS SOL문

- 실행

EXEC 프로시저이름 매개변수값

1. 프로시저

- 프로시저의 매개변수
 - 출력 매개변수
 - 입력 매개변수와 반대로 프로시저의 처리 결과값을 반환하는 매개변수
 - 출력 매개변수 선언

CREATE PROCEDURE 프로시저이름 @매개변수명 타입 OUTPUT, … AS SELECT @매개변수명= 속성명 FROM … WHERE…

- 출력 매개변수 값 받기
 - ▶ 프로시저 실행 전에 매개변수를 선언함(DECLARE문 이용)
 - ▶ 선언된 매개변수를 출력함(SELECT문 이용)
 - ▶ 프로시저 실행 전에 매개변수를 선언함(DECLARE문 이용)

DECLARE @매개변수명

- 프로시저 실행

EXEC 프로시저명 @매개변수명 OUTPUT

- 선언된 매개변수를 출력함(SELECT 문 이용)

SELECT @매개변수명