

저장 프로시저 (Stored Procedure)

저장 프로시저는 데이터베이스 내에 미리 작성된 SQL 코드 블록으로, 일련의 작업을 수행하기 위한 프로그램입니다. 저장 프로시저는 반복적으로 수행되는 작업이나 복잡한 작업을 수행할 때 유용하며, 실행 속도와 보안성을 향상시킬 수 있습니다.

- 간단한 예제

```
create table student (  
    id int auto_increment primary key ,  
    name varchar(20),  
    major varchar(20)  
);  
  
create table course (  
    id int auto_increment primary key ,  
    course_name varchar(20),  
    instructor varchar(20)  
);  
  
insert into student (name, major) values ('Alice', 'English');  
insert into student (name, major) values ('Bob', 'Math');  
insert into student (name, major) values ('Claire', 'Science');  
insert into student (name, major) values ('David', 'History');  
  
insert into course (course_name, instructor) values ('English','Smith');  
insert into course (course_name, instructor) values ('Math','Johnson');  
insert into course (course_name, instructor) values ('Science','Lee');  
insert into course (course_name, instructor) values ('History','Kim');  
  
create procedure getStudent(  
    param_student_id varchar(20)  
)  
begin  
    select s.name,  
        s.major,  
        c.instructor  
    from student s  
    join course c on s.major = c.course_name  
    where s.name = param_student_id;  
end;  
  
call getStudent('Bob');
```

- 결과

name	major	instructor
Bob	Math	Johnson

★ 프로시저의 장점

1. 성능
 - 저장 프로시저는 컴파일된 형태로 데이터베이스 내에 저장 되어 있기 때문에, 일반 SQL 쿼리보다 더 빠르게 실행됩니다. 특히, 반복적으로 수행되는 작업을 수행할 때 성능이 향상 됩니다.
2. 보안성
 - 저장 프로시저는 데이터베이스 내부에서 실행되므로, 외부에서 직접 SQL 쿼리를 실행하는 것보다 보안성이 높습니다. 또한 저장 프로시저를 사용하여 접근 권한을 관리할 수 있습니다.
3. 코드의 재사용성
 - 저장 프로시저는 데이터베이스 내부에 저장되어 있으므로, 다른 어플리케이션에서도 쉽게 재사용할 수 있습니다.
4. 유지 보수성
 - 저장 프로시저를 사용하면 코드를 한 곳에서 관리할 수 있으므로, 유지 보수성이 향상됩니다.

👤 프로시저의 단점

1. 학습 비용
 - 저장 프로시저는 별도의 언어로 작성되기 때문에, SQL 쿼리를 작성하는 것보다 학습 비용이 높을 수 있습니다.
2. 유연성
 - 저장 프로시저는 데이터베이스 내에서만 실행될 수 있으므로, 다른 환경에서 사용하기 어려울 수 있습니다.
3. 디버깅
 - 저장 프로시저를 디버깅하기 어렵습니다. 일반적으로 저장 프로시저를 디버깅하기 위해서는 별도의 도구를 사용해야 합니다.
4. 유지보수 및 개발자 의존성
 - 저장 프로시저를 작성한 개발자가 조직을 떠나거나, 업무를 변경하면 저장 프로시저의 유지보수나 변경이 어렵습니다.