Chap18. 커서와 예외 처리

- ▶ 커서란?
  - ▶ SQL문을 실행했을 때 해당 SQL문 처리 정보를 저장
  - ▶ SELECT문의 결과 행 별로 특정 작업 수행

- ▶ SELECT INTO 방식
  - ▶ 조회되는 행이 하나 일 때
  - ▶ 커서는 조회되는 행의 수와 상관없이 사용

### ▶ 명시적 커서

▶ 직접 커서를 선언하고 사용

단계	명칭	설명
1단계	커서 선언 (declaration)	사용자가 직접 이름을 지정하여 사용할 커서를 SQL문과 함께 선언합니다.
2단계	커서 열기 (open)	커서를 선언할 때 작성한 SQL문을 실행합니다. 이때 실행한 SQL문에 영향을 받는 행을 active set라 합니다.
3단계	커서에서 읽어온 데이터 사용 (fetch)	실행된 SQL문의 결과 행 정보를 하나씩 읽어 와서 변수에 저장한 후 필요한 작업을 수행합니다. 각 행별로 공통 작업을 반복해서 실행하기 위해 여러 종류의 LOOP문을 함께 사용할 수 있습니다.
4단계	커서 닫기 (close)	모든 행의 사용이 끝나고 커서를 종료합니다.

#### ▶ 명시적 커서

```
DECLARE

CURSOR 커서 이름 IS SQL문; -- 커서 선언(Declaration)

BEGIN

OPEN 커서 이름; -- 커서 열기(Open)

FETCH 커서이름 INTO 변수 -- 커서로부터 읽어온 데이터 사용(Fetch)

CLOSE 커서이름; -- 커서 닫기(Close)

END;
```

- ▶ 명시적 커서
  - ▶ 하나의 행만 조회되는 경우
  - ▶ 여러 행이 조회되는 경우
    - **LOOP**
    - ▶ FOR LOOP

속성	설명
커서 이름%NOTFOUND	수행된 FETCH문을 통해 추출된 행이 있으면 false, 없으면 true를 반환합니다.
커서 이름%FOUND	수행된 FETCH문을 통해 추출된 행이 있으면 true, 없으면 false를 반환합니다.
커서 이름%ROWCOUNT	현재까지 추출된 행 수를 반환합니다.
커서 이름%ISOPEN	커서가 열려(open) 있으면 true, 닫혀(close) 있으면 false를 반환합니다.

커서에 파라미터 사용하기

- ▶ 묵시적 커서
  - ▶ SQL문을 사용했을 때 자동으로 선언되는 커서
  - ▶ OPEN, FETCH, CLOSE 지정없음

속성	설명
SQL%NOTFOUND	묵시적 커서 안에 추출한 행이 있으면 false, 없으면 true를 반환합니다. DML 명령어로 영향을 받는 행이 없을 경우에도 true를 반환합니다.
SQL%FOUND	묵시적 커서 안에 추출한 행이 있으면 true, 없으면 false를 반환합니다. DML 명령어로 영향을 받는 행이 있다면 true를 반환합니다.
SQL%ROWCOUNT	묵시적 커서에 현재까지 추출한 행 수 또는 DML 명령어로 영향받는 행 수를 반환합니다.
SQL%ISOPEN	묵시적 커서는 자동으로 SQL문을 실행한 후 CLOSE되므로 이 속성은 항상 false를 반환합니다.

#### ▶ 오류

- ▶ 컴파일 오류 혹은 문법 오류
- ▶ 런타임 오류 혹은 실행 오류 : 예외

#### ▶ 예외 종류

예외 종류		설명
내부 예외 (internal excpetions)	사전 정의된 예외 (predefined name exceptions)	내부 예외 중 예외 번호에 해당하는 이름이 존재하는 예외
	이름이 없는 예외 (unnamed exceptions)	내부 예외 중 이름이 존재하지 않는 예외(사용자가 필요에 따라 이름을 지정할 수 있음)
사용자 정의 예외(user-defined exceptions)		사용자가 필요에 따라 직접 정의한 예외

#### ▶ 예외 처리부 작성

```
EXCEPTION

WHEN 예외 이름1 [OR 예외 이름2 - ] THEN 예외 처리에 사용할 명령어;
WHEN 예외 이름3 [OR 예외 이름4 - ] THEN 예외 처리에 사용할 명령어;

···
WHEN OTHERS THEN 예외 처리에 사용할 명령어;
```

▶ 이름 없는 예외 사용

```
DECLARE
예외 이름1 EXCEPTION;
PRAGMA EXCEPTION_INIT(예외 이름1, 예외 번호);
:

EXCEPTION
WHEN 예외 이름1 THEN
예외 처리에 사용할 명령어;
...
END;
```

▶ 사용자 정의 예외 사용

```
DECLARE
사용자 예외 이름 EXCEPTION;
...
BEGIN
IF 사용자 예외를 발생시킬 조건 THEN
RAISE 사용자 예외 이름
...
END IF;
EXCEPTION
WHEN 사용자 예외 이름 THEN
예외 처리에 사용할 명령어;
...
END;
```

#### ▶ 요류 코드와 오류 메시지 사용

함수	설명
SQLCODE	오류 번호를 반환하는 함수
SQLERRM	오류 메시지를 반환하는 함수