교재 3장

-기본 SQL문

\*\*\*SQLDeveloper\*\*\*

: 새로만들기

Name : system

사용자 이름 : system(슈퍼유저)

비밀번호 : oracle

호스트이름 : localhost

포트 : 1521

SID : xe(데이터베이스이름)

: 계정생성

system계정으로 접속한 상태에서

ez과 scott 계정을 생성한다.

create user ez identified by oracle;

create user scott identified by tiger;

계정생성 후 권한부여를 해야한다.

# 권한 종류와 역할

1. 시스템 권한 : 시스템 작업시 사용하는 권한

DB 수준의 작업을 수행하기 위한 권한 (유저생성, 접속권한, 테이블생성, 모든 테이블 조회)

CREATE USER 데이터베이스 유저생성 권한

SELECT ANY TABLE 모든 유저의 테이블 조회

CREATE ANT TABLE 모든 유저의 테이블 생성

CREATE SESSION 데이터베이스 접속 권한

CREATE TABLE 테이블 생성 권한

2. 오브젝트 권한(DB 객체) : 테이블, 인덱스, 함수 및 프로시저 등 오브젝트에 대한 작

테이블에 대한 데이터 삽입 갱신 및 삭제 권한

ALTER, INSERT, SELECT, UPDATE, DELETE테이블에 대한 데이터 삽입 갱신 및 삭제 권한

3. 기타옵션

WITH GRANT OPTION 권한을 할당받은 유저가 다른 유저에게 해당 권한을 재할당

PUBLIC 해당 권한을 DB에 존재하는 모든 유저에게 할당

# 권한 부여

GRANT [권한1],[권한2..] USER TO [유저명];

GRANT [오브젝트권한] ON [유저명].[테이블명] TO [권한을부여할유저명];

-> grant resource, connect,dba to ez;

grant resource, connect,dba to scott;

-REVOKE 사용자로부터 부여한 권한을 삭제한다

REVOKE CREATE 권한 FROM

: 새로만들기

Name : ez

사용자 이름 : ez

비밀번호 : oracle

호스트이름 : localhost

포트 : 1521

SID : xe(데이터베이스이름)

: 새로만들기

Name : scott

사용자 이름 : scott

비밀번호 : tiger

호스트이름 : localhost

포트 : 1521

SID : xe(데이터베이스이름)

=======================================================

#테이블 만들기

create table MEMBER (

MEMBERID VARCHAR2(10) NOT NULL PRIMARY KEY,

PASSWORD VARCHAR2(10) NOT NULL,

NAME VARCHAR2(20) NOT NULL,

EMAIL VARCHAR2(80)

);

무결성 제약조건의 종류

NOT NULL : NULL을 허용하지 않는다.

UNIQUE : 중복된 값을 허용하지 않는다. 항상 유일한 값이어야한다.

PRIMARY KEY : (NOT NULL+UNIQUE)

FOREIGN KEY : 참조되는 테이블의 칼럼의 값이 존재하면 허용한다.

CHECK : 저장 가능한 데이터 값의 범위나 조건을 정해서 설정한 값만을 허용한다.

PRIMARY KEY(기본키)

기본키를 선택하기 전에 고려해야할 부분은

1 중복되지않아야하고 변경할수 없는 것이어야한다.

2 NULL값을 허용하지 않아야한다.(값이 입력되지않았다는 뜻)

FOREIGN KEY(외래키)

외래키는 다른 기본키를 참조하는 속성이다.

기본키를 참조하여 관계를 표현한다.

참조되는 기본키 값이 변경되면 이 기본키를 참조하는 외래키도 같이 변경된다.

외래키는 NULL 값을 포함할 수 있다.

Drop table Member;테이블삭제

ALTER TABLE MEMBER MODIFY(NAME VARCHAR2(30);기존의 MEMBER테이블의 NAME칼럼의 크기를 20으로 변경

RENAME MEMBER TO MEMBER2 기존의 테이블 MEMBER을 MEMEBER2로 변경한다

TRUNCATE TABLE MEMBER MEMBER테이블의 내용을 전부 제거한다

DROP TABLE MEMBER MEMBER테이블을 제거한다

#데이터 입력하기

insert into member values('SSOL','1234','홍길동','hh@Daum.net');

insert into member values('SSL','1234','홍길동형','hhh@Daum.net');

insert into member values('SSS','1234','홍길동누나','hh11@Daum.net');

insert into member values('SSSS','1234','홍길동아들','hh222@Daum.net');

insert into member values('SSLL','1234','홍길동딸','hhhh@Daum.net');

#데이터 꺼내기

- 전체 꺼내기

Select [DISTINCT](중복되면 하나만가져오기) \* from member(테이블);

- 조건검색

select \* from member(,붙여 다른 테이블이름도 넣을수있음)

where memberid = 'SSOL'; or where sal>=3000 (조건을 추가하여 검색할수있음)

-or연산자 사용하기위해

Where 테이블명 in(조건,조건,조건…);

-manager제외하고 출력

Where not 테이블명 = ‘manager’;

-특정값’K%’시 첫번째글자가 K이고 두자리 세자리 …모두가능한 출력, ’\_A’시 두번째글짜가A인값, ‘A\_\_’시 A가첫번째이고 글자가 세자리인수

Whrere 컬럼명 like ‘K%’;

-사이의값 출력

Where 컬럼명 between 데이터 and 데이터

-NULL값인지 확인

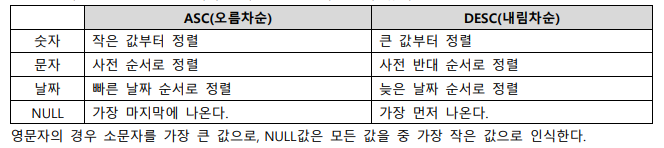
WHERE 컬럼명 IS NULL;

-A~B까지의 글자 가져오기

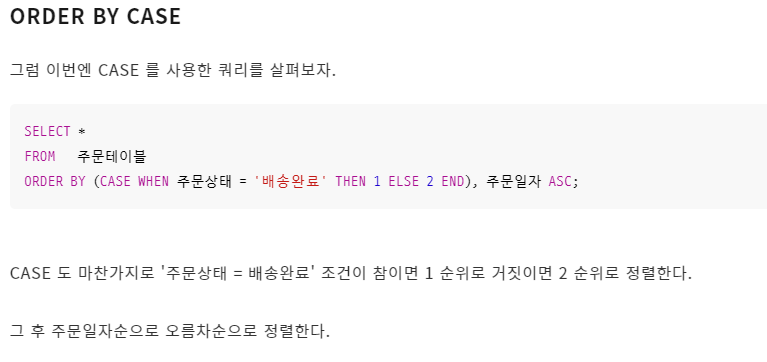
WHERE SUBSTR(칼럼명,A,B);

#데이터 정렬

ORDER BY 컬럼명 DESC;



ORDER BY SALARY ASC, NAME DESC->SALARY ASC순으로 정렬후 그안의 각각 테이블의 NAME은 DESC순으로 정렬한다



#데이터 수정

update member

set name = '홍길동엄마'

where memberid = 'SSS';

#데이터 테이블 삭제

delete from member

where memberid = 'SSS';

rollback;(삭제이전으로 롤백시켜줌)

#작성한 테이블 반영

commit