```sql
--ex7) group by rollup : a, b별 집계(Subtotal) 부서별,직무ID별 급여
평균구하기
select DEPARTMENT_NAME,JOB_TITLE,round(avg(SALARY),2)as
"Avg_sal"
from EMPLOYEES
join DEPARTMENTS using(department_id)
join JOBS using(job_id)
group by rollup (DEPARTMENT_NAME, JOB_TITLE );

--ex8) group by cube : a별 집계 또는 b별 집계
--부서별, 직무 ID별 급여평균구하기(부서를 기준으로 나타내는 평균급여)
select DEPARTMENT_NAME, JOB_TITLE,round(avg(SALARY),2)as
"Avg_sal"
from EMPLOYEES
join DEPARTMENTS using(department_id)
join JOBS using(job_id)
group by cube (DEPARTMENT_NAME, JOB_TITLE);

--ex9) group by grouping Sets
--직무별 평균급여와 전체사원의 평균급여를 함께 구하시오
select JOB_TITLE,round(avg(SALARY),2) as "Avg_sal"
from EMPLOYEES
join JOBS using(job_id)
group by grouping sets (( JOB_TITLE ),());


--ex1)
create table test(
id number(5),
name char(10),
address varchar2(50));

create table user1(
    idx number primary key ,
    id varchar2(10) unique ,
    name varchar2(10) not null ,
    phone varchar2(15),
    address varchar2(50),
    score number(6,2) check ( score >= 0 and score <= 100),
    subject_code number(5),
    hire_date date default sysdate,
    marriage char(1) default 'N' check ( marriage
in('Y','N')));

--제약조건 확인
select constraint_name, constraint_type
from USER_CONSTRAINTS
where table_name = 'USER1';

select *
```

```
from USER_CONSTRAINTS
where table_name='USER1';

create table user2(
    idx number constraint PKIDX primary key,
    id varchar2(10) constraint UNID unique ,
    name varchar2(10) constraint NOTNAME not null ,
    phone varchar2(15),
    address varchar2(50),
    score number(6,2) constraint CKSCORE check ( score>=0 and
score <=100 ),
    subject_code number(5),
    hire_date date default sysdate,
    marriage char(1) default 'N'constraint CKMARR check
( marriage in('Y','N' )));
--transaction처리
--:일의 시작과 끝이 완벽하게 마무리(commit)
--처리도중 인터럽터(interrupt:장애)가 발생하면 되돌아옴(rollback)

-- ex6)
insert into
user1(idx,id,name,phone,address,score,subject_code,hire_date,m
arriage) values(1,'aaa','kim','010-000-0000','서
울',75,100,'2010-08-01','Y');
insert into
user1(idx,id,name,phone,address,score,subject_code,hire_date,m
arriage) values(1,'aaa','kim','010-000-0000','서
울',75,100,'2010-08-01','Y');
--→ 무결성제약조건에 위배(이유: idx 1 중복)
insert into
user1(idx,id,name,phone,address,score,subject_code,hire_date,m
arriage) values(2,'aaa','kim','010-000-0000','서
울',75,100,'2010-08-01','Y');
--→ 무결성제약조건에 위배(이유: id aaa 중복)
insert into
user1(idx,id,name,phone,address,score,subject_code,hire_date,m
arriage) values(2,'bbb','','010-000-0000','서
울',75,100,'2010-08-01','Y');
--→ NULL을 ("HR"."USER1"."NAME") 안에 삽입할 수 없습니다
insert into
user1(idx,id,name,phone,address,score,subject_code,hire_date,m
arriage) values(2,'bbb','lee','010-000-0000','서
울',120,100,'2010-08-01','Y');
--→ 체크 제약조건에 위배되었습니다(이유: score가 0~100사이의 수 이어야함)
insert into
user1(idx,id,name,phone,address,score,subject_code,hire_date,m
arriage) values(2,'bbb','lee','010-000-0000','서
울',75,100,'2010-08-01','K');
--→ 체크 제약조건에 위배되었습니다(이유 : marriage가 Y 또는 N 이어야함)
```

```
insert into
user1(idx,id,name,phone,address,score,subject_code,hire_date,m
arriage) values(2,'bbb','lee','010-000-0000','서
울',75,100,'2010-08-01','N');

commit ;

--ex7)테이블 목록 확인
select * from tab;
--ex8)테이블의 레코드 확인
select * from user1;
select * from user2;

--ex10)시퀀스 목록 확인
select * from USER_SEQUENCES;

--ex11)테이블명 변경 : test -> user3
alter table test rename to user3;

--ex12)컬럼 추가 : user3 -> phone varchar2(15)
alter table user3 add phone varchar2(15);

--ex13) 제약조건 추가 : user3 -> id에 unique,제약조건명 UID2
alter table user3 add constraint UID2 unique(id);

select constraint_name, constraint_type
from USER_CONSTRAINTS
where table_name ='USER3';

--ex14) 제약조건 삭제 -alter table 테이블명 drop constraint 제약조건
명 ;
alter table user3 drop constraint UID2;

select *
from USER_CONSTRAINTS
where table_name = 'USER3';

--ex15) 컬럼 추가 : user3 -> no number (pk 설정)
alter table user3 add no number primary key ;

--ex16) 구조 변경 : user3 -> name char(10) 를 varchar2(10)로 바꿈
alter table user3 modify name varchar2(10);

--ex17) 컬럼 삭제 :user3 ->address
alter table user3 drop column address;

--ex18) 테이블 삭제/ 휴지통비우기 :user3
drop table user3;
select* from tab;
```

```
purge recyclebin;--휴지통비우기

drop table user1 purge;-- 휴지통에 넣지 않고 바로 삭제

drop table user2;

show recyclebin;-- 휴지통 보기

flashback table user2 to before drop ; --휴지통에서 되살리기

--ex19)시퀀스 생성 / 삭제
create sequence idx_sql increment by 2 start with 1 maxvalue 9
cycle nocache ;

select IDX_SQL.nextval from dual;--다음 시퀀스값 표시
select IDX_SQL.currval from dual;--현재 시퀀스값 표시

 select * from USER_SEQUENCES;

drop sequence idx_sql;

--ex20) 테이블 생서과 시퀀스 적용
create table book(
    no number primary key ,
    subject varchar2(50),
    price number,
    year date);

create sequence no_seq increment by 1 start with 1 nocycle
nocache ;

insert into book(no, subject, price, year)
values (no_seq.nextval,'오라클 무작정 따라하기', 100000, sysdate);

insert into book(no, subject, price, year)
values (no_seq.nextval, '자바 3일 완성', 15000, '2007-03-01');

insert into book values (no_seq.nextval, 'JSP달인 되
기',18000,'2010-01-01');

select*from book;

--ex21) 테이블 구조만 복사
create table user3 as select * from user2 where 1=0;
-->not null을 제외하고는 제약조건이 복사 안됨
-->not null 제약조건도 sys_*****로 복사됨(제약조건명 그대로 복사가 안된다)

--ex22) 테이블(idx -> bunho, name -> irum, address->juso)를 복사하
고 id가 bbb인 레코드를 복사하시오
```

```sql
create table user4(bunho,irum,juso)
as select idx,name, address from user1 where id = 'bbb';

select * from user1;
SELECT * from user4;

--ex23) 테이블 생성 후 행 추가
create table dept(
    deptno number constraint DNO primary key,
    dname varchar2(30) constraint DNAME not null);
create table emp(
    empno number constraint ENO primary key,
    ename varchar2(30) constraint ENAME not null,
    deptno number,
    constraint FKNO foreign key(deptno) references dept on
delete set null);

insert into dept(deptno, dname) values(10, '개발부');
insert into dept(deptno, dname) values(20, '영업부');
insert into dept(deptno, dname) values(30, '관리부');
insert into dept(dname) values(40, '경리부');
--→ ORA-00913: 값의 수가 너무 많습니다.
insert into dept(deptno, dname) values(40, '경리부');
insert into emp(empno, ename, deptno) values(100, '홍길동', 10);
insert into emp(empno, ename, deptno) values(101, '라이언', 20);
insert into emp(empno, ename, deptno) values(102, '튜브', 50);
--→ 50번부서 없음(무결성제약조건위배) - 부모키가 없습니다
insert into emp(empno, ename, deptno) values(103, '어피치', 40);
insert into emp(empno, ename) values(105, '프로도');
insert into emp(ename, deptno) values('콘', 10);
--→ primary key는 NULL허용 안함
commit;

--ex24) 삭제
delete from dept;
select * from dept;
rollback ;
commit;

delete from dept where deptno = 40;
select * from dept;--40번 부서 삭제

select * from emp;--기존에 있던 40번 부서는 null값이 들어감

--ex25) 수정 (update) -emp테이블 장동건 사원의 부서번호를 30으로 수정하시오
update emp set deptno=30 where ename = '프로도';
select  * from emp;
commit ;
```