1. 매주 1회 구매하는 마스크 관련 Transaction 처리를 위한 Table을 설계하시요.

- 마스크를 구매하는 구매자 전국민 관련 Table(이름,출생년도(주민번호) ,장애여부,대리구매 가능여부,주소) , 구매 이력을 관리하는 Table(이름,주민번호,구매여부,) 2개 설계

- Table 이름,필요한 컬럼,컬럼 이름,PK 등은 본인이 알아서 할 것.

2. 위 설계된 Table을 기초로 구매하는 Transaction을 Procedure 로 작성하시요

- 월요일~ 금요일까지 출생년도 끝자리기준 5부제에 구해 구매하도록 함((1,6)월,(2,7)화,(3,8)수,(4,9)목,(5,0)금).

- 만 10살 이하 어린이와 만 80살 노인의 마스크 구매는 가족 등 동거인이 대리 구매 가능 함.

(1940 년도이상 출생자,2010년도 이하 출생자)

- 장애인은 연령에 관계없이 대리 구매 가능.

- 대리구매 자격은 주민등록부상 동거인인 경우만 가능.

- 주중에 구매하지 못한 경우 주말에 출생년도에 관계없이 구매 가능.

- 총 3장 구매 가능. 분할 구매 가능, 예를 들어 주중 2장,주말 1장

**- 이번에는 결과물을 파워포인트로 작성할 것.**

**- 파워포인트 작성 후 직원들 앞에서 발표 예정.**

**Table 생성)**

**drop table nation\_tab;**

**create table nation\_tab(**

**birthno varchar2(14) primary key not null,**

**name varchar2(50)not null,**

**obstacle varchar2(1),**

**address varchar2(50) not null,**

**lordno varchar2(14)not null**

**);**

**drop table sal\_mask\_tab;**

**create table sal\_mask\_tab(**

**name varchar2(50),**

**birthno varchar2(14)not null,**

**sal\_mask number,**

**proxy\_sal\_mask varchar2(10),**

**sal\_date date,**

**payment varchar2(50),**

**store\_name varchar2(100)**

**);**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('050101-3484634','leejinsang','T','jongro.apt','220101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('150101-4484634','leejinho','F','jongro.apt','220101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('220101-1484634','leejinhan','F','jongro.apt','220101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('330101-2484634','kimcumji','F','guro.apt','440101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('440101-1484634','kimchunsam','F','guro.apt','440101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('550101-2484634','kimjisun','F','guro.apt','440101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('660101-1484634','parkjihun','T','seoul.apt','660101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('770101-2484634','parkyeonseo','T','seoul.apt','660101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('880101-1484634','parkjisung','T','incheon.apt','660101-1484634');**

**insert into nation\_tab(birthno,name,obstacle,address,lordno)values('990101-2484634','chiohyeonjun','T','paju.apt','990101-2484634');**

**commit;**

**mask 구매 procedure)**

**create**

**or replace procedure proc\_sal\_mask(p\_name varchar2, p\_birthno varchar2, p\_sal\_mask number, p\_proxy\_sal\_mask varchar2, p\_payment varchar2, p\_store\_name varchar2) is v\_tot\_mask number ;**

**v\_cnt number;**

**v\_day varchar2(10);**

**v\_cus\_day varchar2(10);**

**more\_mask exception;**

**dont\_sal\_mask exception;**

**begin**

**select count(\*) into v\_cnt from NATION\_TAB where name = p\_name and birthno= p\_birthno;**

**if v\_cnt !=1 then raise NO\_DATA\_FOUND; end if;**

**for tr in(select name,**

**birthno,**

**case**

**when substr(BIRTHNO, 2, 1) in(1,6) then '월'**

**when substr(BIRTHNO, 2, 1) in(2,7) then '화'**

**when substr(BIRTHNO, 2, 1) in(3,8) then '수'**

**when substr(BIRTHNO, 2, 1) in(4,9) then '목'**

**when substr(BIRTHNO, 2, 1) in(5,0) then '금'**

**end sal\_date\_dy,**

**to\_char(sysdate, 'dy')dy**

**from NATION\_TAB**

**where birthno=p\_birthno**

**and name = p\_name) loop**

**v\_day := tr.dy;**

**v\_cus\_day := tr.sal\_date\_dy;**

**for re in (select sum(nvl(sal\_mask, 0))tot\_mask**

**from SAL\_MASK\_TAB**

**where birthno =p\_birthno ) loop if p\_sal\_mask+nvl(re.tot\_mask,0)>3**

**then v\_tot\_mask := nvl(re.tot\_mask,0);**

**raise more\_mask;**

**end if;**

**end loop;**

**if tr.sal\_date\_dy in ('토',**

**'일',**

**tr.dy) then**

**insert**

**into SAL\_MASK\_TAB(NAME, BIRTHNO, SAL\_MASK, PROXY\_SAL\_MASK, SAL\_DATE, PAYMENT, STORE\_NAME)**

**values(p\_name,**

**p\_birthno,**

**p\_sal\_mask,**

**p\_proxy\_sal\_mask,**

**sysdate,**

**p\_payment,**

**p\_store\_name);**

**else raise dont\_sal\_mask ;**

**end if;**

**end loop;**

**exception**

**when NO\_DATA\_FOUND then RAISE\_APPLICATION\_ERROR(-20001, 'Name 이나 BIRTHNO 정보가 정확하지 않습니다');**

**when more\_mask then RAISE\_APPLICATION\_ERROR(-20003, '한주에 구매할 수 있는 마스크는 3개입니다 '||'현재'||v\_tot\_mask||'개 구매하였습니다');**

**when dont\_sal\_mask then RAISE\_APPLICATION\_ERROR(-20004, '오늘은 '||v\_day||'요일 이고'||p\_name||'님이 구매 가능한 날짜는 '||v\_cus\_day||'요일 입니다');**

**end proc\_sal\_mask;**

**대리구매 procedure)**

**create**

**or replace procedure proc\_proxy\_purchashing(p\_name varchar2, p\_birthno varchar2, p\_proxy\_nm varchar2, p\_proxy\_no varchar2, p\_sal\_mask number, p\_payment varchar2, p\_store\_name varchar2) is v\_lordno varchar2(100);**

**v\_proxy\_lordno varchar2(100);**

**v\_cnt number;**

**v\_age number;**

**v\_obstacle varchar2(10);**

**not\_proxy exception;**

**begin select obstacle,**

**l\_cnt**

**into v\_obstacle,**

**v\_cnt**

**from nation\_tab,**

**(select count(lordno)l\_cnt**

**from NATION\_TAB**

**where BIRTHNO in(p\_birthno,**

**p\_proxy\_no)**

**and NAME in(p\_name,**

**p\_proxy\_nm))**

**where BIRTHNO= p\_birthno ;**

**if v\_cnt !=2 then raise NO\_DATA\_FOUND;**

**END if;**

**select**

**case**

**when substr(birthno, 8, 1) in (1,2) then 19||substr(birthno, 1, 6)**

**else 20||substr(birthno, 1, 6)**

**end age**

**into v\_age**

**from NATION\_TAB**

**where name = p\_name**

**and birthno = p\_birthno;**

**v\_age:= trunc(months\_between(trunc(sysdate), to\_date(v\_age, 'yyyymmdd'))/12);**

**if v\_age between 10 and 80**

**and v\_obstacle = 'F' then raise not\_proxy;**

**end if;**

**proc\_sal\_mask(p\_name, p\_birthno, p\_sal\_mask, p\_proxy\_no, p\_payment, p\_store\_name);**

**exception**

**when not\_proxy then raise\_application\_error(-20005, '** **대리구매 해당자가 아닙니다.');**

**when NO\_DATA\_FOUND then raise\_application\_error(-20006, '정보가 일치하지 않습니다.');**

**end proc\_proxy\_purchashing;**