1. 아래와 같이 Table 생성,Data를 입력한 후 /질문에 대한 SQL을 최소 각각 3개씩 작성하시요!!!!

drop table cust\_status;  
create table cust\_status  
 (cust\_id      char(1)      not null,  
  cust\_id\_seq  number       not null,  
  status       varchar2(10) not null)  
 ;

insert into cust\_status values ('A',1,'정상');  
insert into cust\_status values ('A',2,'위험');  
insert into cust\_status values ('B',1,'정상');  
insert into cust\_status values ('B',2,'정상');  
insert into cust\_status values ('C',1,'위험');  
insert into cust\_status values ('C',2,'위험');  
insert into cust\_status values ('D',1,'위험');  
insert into cust\_status values ('D',2,'위험');  
insert into cust\_status values ('D',3,'정상');  
insert into cust\_status values ('E',1,'정상');  
commit;

**Q1>  cust\_id 별로 status 값이  한 종류만 가진  cust\_id 만 출력**

**<1번답>**

select cust\_id

from(

select cust\_id,count(\*) over (partition by cust\_id) as cnt

from

(select cust\_id,status from cust\_status group by cust\_id,status) cust

)

where cnt=1;

**<2번답>**

select distinct cust\_id

from cust\_status

where cust\_id not in

(

select cust\_id

from(

select cust\_id,row\_number() over (partition by cust\_id order by cust\_id) as cnt

from

(select cust\_id,status from cust\_status group by cust\_id,status) cust

)

where cnt=2

);

**<3번답>**

select distinct cust\_id

from cust\_status

where cust\_id not in

(

select cust\_id

from(

select cust.\* ,max(status)over(partition by cust\_id) as max\_cust

from cust\_status cust

)

where max\_cust!=status

)

;

**Q2> ﻿ cust\_id 별로 status 값이  한 종류만 가진  Row 전체를 출력**

**<1번답>**

select cust2.\*

from(

select cust\_id,count(\*) over (partition by cust\_id) as cnt

from

(select cust\_id,status from cust\_status group by cust\_id,status)

) cust

, cust\_status cust2

where cust.cust\_id = cust2.cust\_id and cnt=1;

**<2번답>**

select cust2.\*

from(

select distinct cust\_id

from cust\_status

where cust\_id not in

(

select cust\_id

from(

select cust\_id,row\_number() over (partition by cust\_id order by cust\_id) as cnt

from

(select cust\_id,status from cust\_status group by cust\_id,status) cust

)

where cnt=2

) ) cust

, cust\_status cust2

where cust.cust\_id = cust2.cust\_id

;

**<3번답>**

select cust2.\*

from(

select distinct cust\_id

from cust\_status

where cust\_id not in

(

select cust\_id

from(

select cust.\* ,max(status)over(partition by cust\_id) as max\_cust

from cust\_status cust

)

where max\_cust!=status

) ) cust

, cust\_status cust2

where cust.cust\_id = cust2.cust\_id

;

2. 아래와 같은 Table을 생성 ,Data를 입력한 후 20190101~20191231까지 총 365건 이 조회되도록 일자별집계를 구하시요.

TABLE명 : REPAY\_TEST

**1) 상환 후 잔액 거래 내역**

|  |  |  |  |
| --- | --- | --- | --- |
| **REPAY\_DATE** | **DETR\_NM** | **RBNO** | **LOAN\_BAL\_AMT** |
| 20190103 | 홍길동 | 1234567-1234567 | 1500000 |
| 20190906 | 홍길동 | 1234567-1234567 | 1000000 |
| 20190909 | 홍길동 | 1234567-1234567 | 500000 |

**2) 상환 후 잔액 일별 집계**

|  |  |  |  |
| --- | --- | --- | --- |
| **TOT\_DATE** | **DETR\_NM** | **RBNO** | **LOAN\_BAL\_AMT** |
| 20190101 | 홍길동 | 1234567-1234567 | 0 |
| 20190102 | 홍길동 | 1234567-1234567 | 0 |
| 20190103 | 홍길동 | 1234567-1234567 | 1500000 |
| 20190104 | 홍길동 | 1234567-1234567 | 1500000 |
| 20190105 | 홍길동 | 1234567-1234567 | 1500000 |
| 20190106 | 홍길동 | 1234567-1234567 | 1500000 |
| 상동 | 상동 | 상동 | 상동 |
| 20190905 | 홍길동 | 1234567-1234567 | 1500000 |
| 20190906 | 홍길동 | 1234567-1234567 | 1000000 |
| 20190907 | 홍길동 | 1234567-1234567 | 1000000 |
| 20190908 | 홍길동 | 1234567-1234567 | 1000000 |
| 20190909 | 홍길동 | 1234567-1234567 | 500000 |
| 20190910 | 홍길동 | 1234567-1234567 | 500000 |
| 20190911 | 홍길동 | 1234567-1234567 | 500000 |
| 상동 | 상동 | 상동 | 상동 |

select tot\_date,detr\_nm,rbpo,

nvl(amt,

(

select loan\_bal\_amt

from REPAY\_TEST

where repay\_date=

(

select max(repay\_date)

from repay\_test

where REPAY\_date<tot\_date and loan\_bal\_amt is not null

)

)

) as amt

from(

select

tot\_date,detr\_nm,rbpo,

(select loan\_bal\_amt from REPAY\_TEST where repay\_date=tot\_date) as amt

from(

select

to\_char((to\_date('20190101','yyyymmdd') + (level - 1)) ,'yyyymmdd') as tot\_date,

r\_t2.detr\_nm,

r\_t2.rbpo

from dual,

(select max(detr\_nm) detr\_nm,max(rbpo) as rbpo from repay\_test r\_t) r\_t2

connect by level <= 365

order by tot\_date

) from2 --통합테이블

);