

🌸 19. time base 불완전 복구 수행

완전 복구와 불완전 복구의 차이

- 완전 복구는 장애가 나기 전에 마지막 commit 시점으로 복구하는 것
- 불완전 복구는 과거의 특정 시점으로 db를 되돌리는 것

불완전 복구를 해야 할 때

- 완전 복구를 못했을 때
- 고객의 요청으로 특정 시점으로 db를 되돌려야 할 때
- 실수로 유저를 drop 했을 때
- 실수로 테이블 스페이스를 rop 했을 때
- current와 active 상태의 redo log group 의 멤버가 모두 삭제되었을 때

특정 시간으로 불완전 복구 하기

- 만약 금요일 3시에 scott 유저를 drop을 했고, 3시 이전으로 db를 되돌리고 싶다. 그러려면 과거에 백업 받았던 파일들 중에 모든 data file들을 원본 위치에 가져온다. 기존에 있는 data file들은 모두 지우고 가져온다. 그리고 나서 2시 59분으로 모든 data file들을 복구한다. 복구하고 나면 data file 들은 2시 59분인데 control file과 redo log file은 현재 4시 (4시라고 가정) 상태의 파일들이다. 그러면 서로 싱크가 안 맞아서 db가 안 올라간다. 그래서 ****resetlogs**** 라는 옵션을 써서 db를 올린다. 그래서 db 전체 2시 59분 상태가 된다.

실습

1. cold backup을 수행한다.

```
-> /home/oracle 밑에 coldbackup3라는 폴더 생성
(psh)
mkdir /home/oracle/coldbackup3
```

```
-> datafile과 control file과 redo log file의 위치 확인
(psh sys)
@datafile
@controlfile
@logfile
```

⇒ 모두 다 psh 파일 아래에 있다. (정상)

```
psh(SYS) > @datafile

FILE_NAME
-----
/u01/app/oracle/oradata/psh/users01.dbf
/u01/app/oracle/oradata/psh/undotbs01.dbf
/u01/app/oracle/oradata/psh/sysaux01.dbf
/u01/app/oracle/oradata/psh/system01.dbf
/u01/app/oracle/oradata/psh/example01.dbf
```

```
psh(SYS) > @controlfile

NAME
-----
/u01/app/oracle/oradata/psh/control01.ctl
/u01/app/oracle/oradata/psh/control02.ctl
/u01/app/oracle/oradata/psh/control03.ctl
```

```

psh(SYS) > @logfile

      GROUP#
-----
MEMBER
-----
          1
/u01/app/oracle/oradata/psh/redo01.log

          1
/u01/app/oracle/oradata/psh/redo01b.log

          2
/u01/app/oracle/oradata/psh/redo02b.log

      GROUP#
-----
MEMBER
-----
          2
/u01/app/oracle/oradata/psh/redo02.log

          3
/u01/app/oracle/oradata/psh/redo03b.log

          3
/u01/app/oracle/oradata/psh/redo03.log

```

shutdown immediate

-> 원본 datafile, control file, redo log file 모두 /home/oracle/coldbackup3에 백업
cp /u01/app/oracle/oradata/psh/* /home/oracle/coldbackup3
아니면
cd /home/oracle/coldbackup3
cp /u01/app/oracle/oradata/psh/* .
-> 한 칸 띄고 '.'은 현재 디렉토리라는 뜻

2. db를 startup 하고 log switch를 여러 번 일으킨다.
(psh sys)
startup

@logsw (3번 정도 실행)

3. emp400 테이블을 만들고 log switch를 여러번 일으킨다.
(psh scott)
create table emp400
tablespace users
as
select * from emp;

@logsw (3번 정도 실행)

4. 현재 시간을 확인한다.
alter session set nls_date_format = 'RRRR/MM/DD:HH24:MI:SS';

select sysdate from dual; -> 시간 따로 적어두기

@logsw
@ckpt

```

psh(SCOTT) > select sysdate from dual;

SYSDATE
-----
2024/03/05:15:23:58

```

5. emp400을 drop 한다.

```
drop table emp400 purge;
```

- emp400 이 drop 되기 전으로 db를 불완전 복구한다.

6. db를 정상 종료한다.

```
(psh sys)
```

```
shutdown immediate
```

7. mount 상태로 db를 올린다.

```
startup mount
```

8. 기존의 data file들을 모두 지운다. (controlfile과 redo log file은 그대로 둔다.)

```
(psh)
```

```
cd /u01/app/oracle/oradata/psh
```

```
rm *.dbf
```

9. 백업 받은 모든 datafile들을 복원한다.

```
cp /home/oracle/coldbackup3/*.dbf .
```

10. emp400이 drop 되기 전으로 불완전 복구를 수행한다.

```
set autorecovery on
```

-> 복구할 때 archive log file을 적용하는거 안 물어보고 자동으로 다 적용 되겠끔 하는 명령어

```
alter session set nls_date_format = 'RRRR/MM/DD:HH24:MI:SS';
```

```
recover database until time '2024/03/05:15:23:58';
```

```
psh(SYS) > recover database until time '2024/03/05:15:23:58';
ORA-00279: change 2686801 generated at 03/05/2024 15:14:34 needed for thread 1
ORA-00289: suggestion :
/u01/app/oracle/flash_recovery_area/PSH/archivelog/2024_03_05/o1_mf_1_46_lyfgd76
4_.arc
ORA-00280: change 2686801 for thread 1 is in sequence #46

ORA-00279: change 2686908 generated at 03/05/2024 15:19:51 needed for thread 1
ORA-00289: suggestion :
/u01/app/oracle/flash_recovery_area/PSH/archivelog/2024_03_05/o1_mf_1_47_lyfgd94
5_.arc
ORA-00280: change 2686908 for thread 1 is in sequence #47

ORA-00279: change 2686911 generated at 03/05/2024 15:19:53 needed for thread 1
ORA-00289: suggestion :
/u01/app/oracle/flash_recovery_area/PSH/archivelog/2024_03_05/o1_mf_1_48_lyfgdbp
n_.arc
ORA-00280: change 2686911 for thread 1 is in sequence #48

ORA-00279: change 2686915 generated at 03/05/2024 15:19:54 needed for thread 1
ORA-00289: suggestion :
/u01/app/oracle/flash_recovery_area/PSH/archivelog/2024_03_05/o1_mf_1_49_lyfglsy
z_.arc
ORA-00280: change 2686915 for thread 1 is in sequence #49

ORA-00279: change 2687018 generated at 03/05/2024 15:23:21 needed for thread 1
ORA-00289: suggestion :
/u01/app/oracle/flash_recovery_area/PSH/archivelog/2024_03_05/o1_mf_1_50_lyfglvo
9_.arc
ORA-00280: change 2687018 for thread 1 is in sequence #50

Log applied.
Media recovery complete.
```

11. resetlogs 옵션을 써서 db를 올린다.

```
alter database open resetlogs;
```

12. scott 유저의 emp400이 보이는지 확인하시오.

```
(psh scott)
select count(*) from emp400;
```

문제 1. log_status를 써서 db가 올라갔는지 확인하시오.
@log_status

문제 2. 로그 스위치를 일으켜서 unused를 없애시오.
@logsw

* reset logs로 open 했으면 반드시 다시 full backup을 수행해야 한다.
예전 백업본은 이제 필요가 없는 상태가 되었다.

문제 1. cold backup을 수행하고 나서 이번에는 scott 유저를 drop 하고 scott 유저에서 drop 되기 전 시점으로 불완전 복구하시오.

```
(psh)
mkdir /home/oracle/coldbackup4
```

```
(psh sys)
shutdown immediate
```

```
(psh)
cd /home/oracle/coldbackup4
cp /u01/app/oracle/oradata/psh/* .
```

```
(psh sys)
startup
```

@logsw (3번 정도 실행)

```
alter session set nls_date_format = 'RRRR/MM/DD:HH24:MI:SS';
select sysdate from dual;
```

```
psh(SYS) > select sysdate from dual;

SYSDATE
-----
2024/03/05:16:03:31
```

```
drop user scott cascade;
```

```
shutdown immediate
startup mount
```

```
(psh)
cd /u01/app/oracle/oradata/psh
rm *.dbf
```

```
cp /home/oracle/coldbackup4/*.dbf .
```

```
set autorecovery on
alter session set nls_date_format = 'RRRR/MM/DD:HH24:MI:SS';
recover database until time '2024/03/05:16:03:31';
```

```
alter database open resetlogs;
```

```
connect scott/tiger;
```

```
psh(SYS) > recover database until time '2024/03/05:16:03:31';
ORA-00279: change 2688127 generated at 03/05/2024 16:00:14 needed for thread 1
ORA-00289: suggestion :
/u01/app/oracle/flash_recovery_area/PSH/archivelog/2024_03_05/o1_mf_1_5_lyfjxs1c
_arc
ORA-00280: change 2688127 for thread 1 is in sequence #5

Log applied.
Media recovery complete.
psh(SYS) > alter database open resetlogs;
Database altered.
psh(SYS) > connect scott/tiger;
Connected.
psh(SCOTT) > █
```

문제 2. redo log group을 총 6개로 생성하고 그룹당 멤버를 2개씩 생성하시오.

```
alter database add logfile group 4 '/u01/app/oracle/oradata/psh/redo04.log' size 10m;
alter database add logfile group 5 '/u01/app/oracle/oradata/psh/redo05.log' size 10m;
alter database add logfile group 6 '/u01/app/oracle/oradata/psh/redo06.log' size 10m;
```

```
@log_status
@logfile
```

```
alter database add logfile member '/u01/app/oracle/oradata/psh/redo04b.log' to group 4;
alter database add logfile member '/u01/app/oracle/oradata/psh/redo05b.log' to group 5;
alter database add logfile member '/u01/app/oracle/oradata/psh/redo06b.log' to group 6;
```

```
select group#, members
from v$log;
```

```
psh(SYS) > select group#, members
from v$log; 2
```

GROUP#	MEMBERS
1	2
2	2
3	2
4	2
5	2
6	2

```
6 rows selected.
```