

오늘 만큼은...

오늘만큼은

"기분 좋게 살자"

남에게 상냥한 미소를 짓고,

어울리는 복장으로 조용히 이야기하며,

예절 바르게 행동하고, 아낌없이 남을 칭찬하자.

오늘만큼은

"이 하루가 보람되도록 하자"

인생의 모든 문제는 한꺼번에 해결되지 않는다.

하루가 인생의 시작인 것 같은 기분으로 오늘을 보내자.

오늘만큼은

"계획을 세우자"

매 시간의 예정표를 만들자.

조급함과 망설임이라는 두 가지 해충을 없애도록 마음을 다지자.

할 수 있는데 까지 해 보자.

오늘 만큼은...

오늘만큼은

"30분 정도의 휴식을 갖고 마음을 정리해 보자"

깊이 생각하고 인생을 관조해 보자.

자기 인생에 대한 올바른 인식을 얻도록 하자.

오늘만큼은

"그 무엇도 두려워하지 말자"

특히, 아름다움을 즐기며 사랑하도록 하자.

사랑하는 사람이 나를 사랑한다는 믿음을 의심하지 말자.

백업과 복원

데이터베이스 백업

백업의 유형

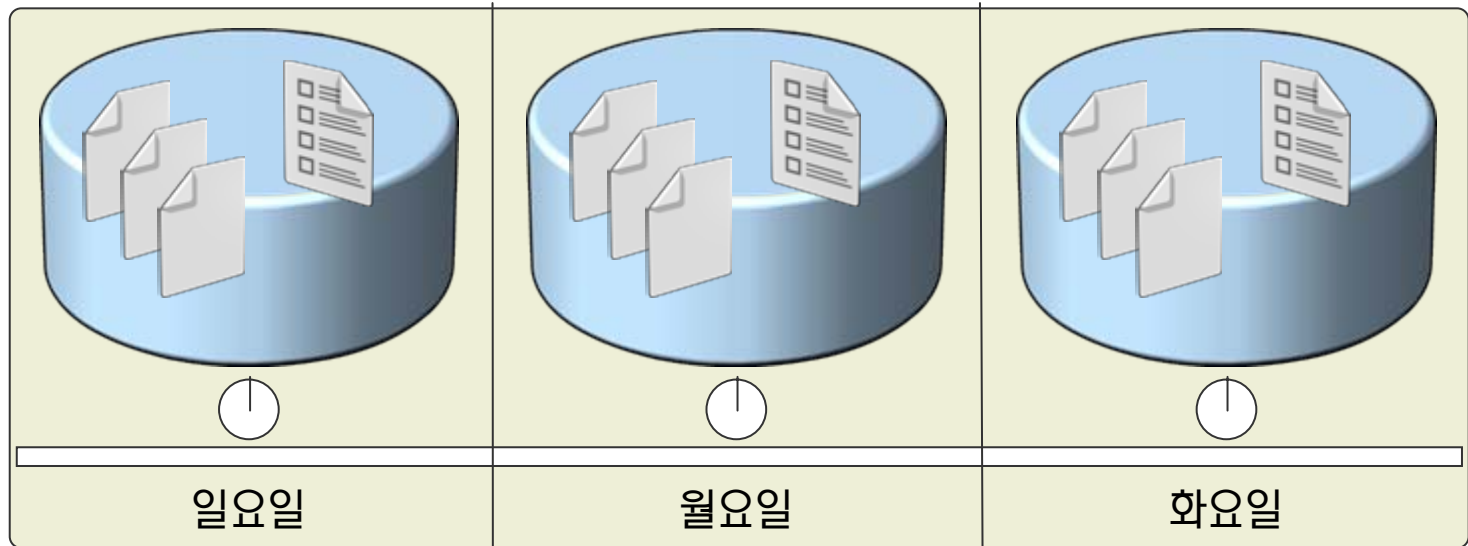
백업 유형	설명
전체	모든 데이터 파일과 트랜잭션 로그 일부
트랜잭션 로그	로그 파일에 있는 데이터베이스에 대한 변경 내용
꼬리 로그	백업 받지 않은 남아 있는 로그
차등	최종 전체 백업 이후에 변경된 내용 및 로그 일부
파일/파일그룹	특정 파일 또는 파일 그룹
부분(Partial) *	PRIMARY 파일그룹과 읽기/쓰기 파일그룹, 또는 지정된 파일 그룹
복사 전용 *	데이터베이스와 로그 (기존 백업 시퀀스에는 영향을 주지 않음)

*) MS SQL 서버 2005에서 지원

복구 모델과 백업

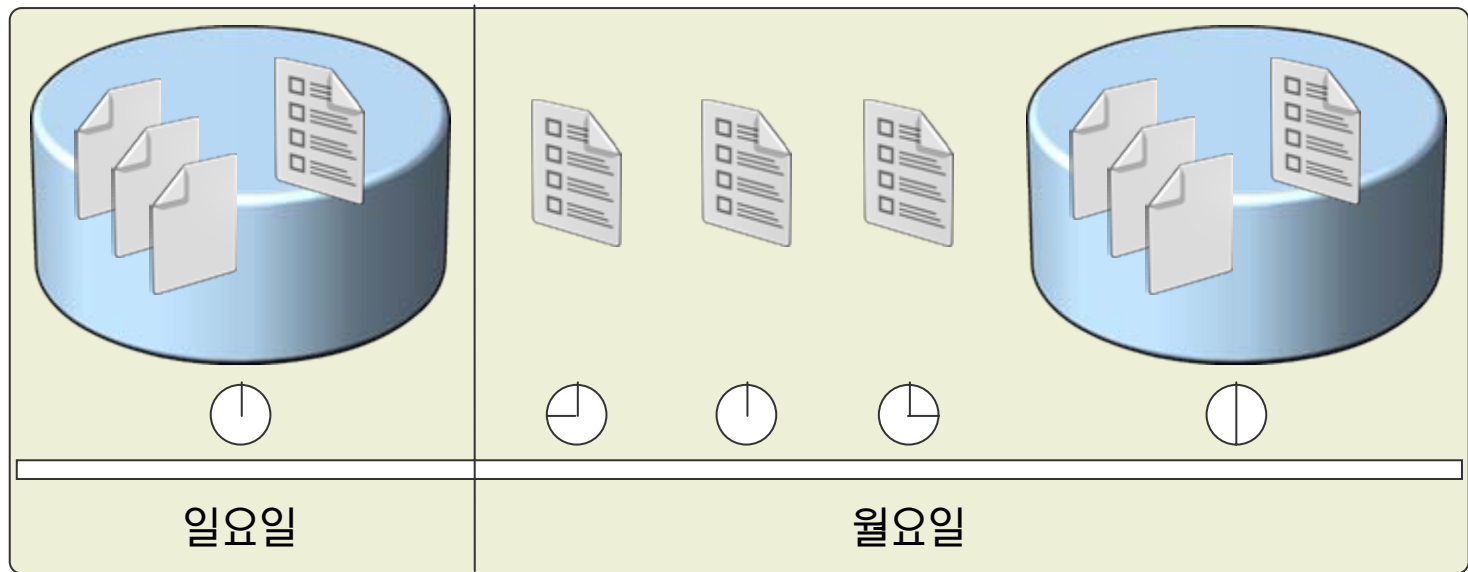
복구 모델	설명
단순 (Simple)	전체 또는 차등 백업이 가능함. 트랜잭션 로그는 삭제되므로 백업 불가.
전체 (Full)	전체, 차등 백업과 트랜잭션 로그 백업이 가능함
대량 로그 (Bulk-logged)	전체, 차등 백업과 트랜잭션 로그 백업이 가능함. 몇 몇 작업에 대해서는 로그를 최소로 남기게 됨

전체 백업 전략



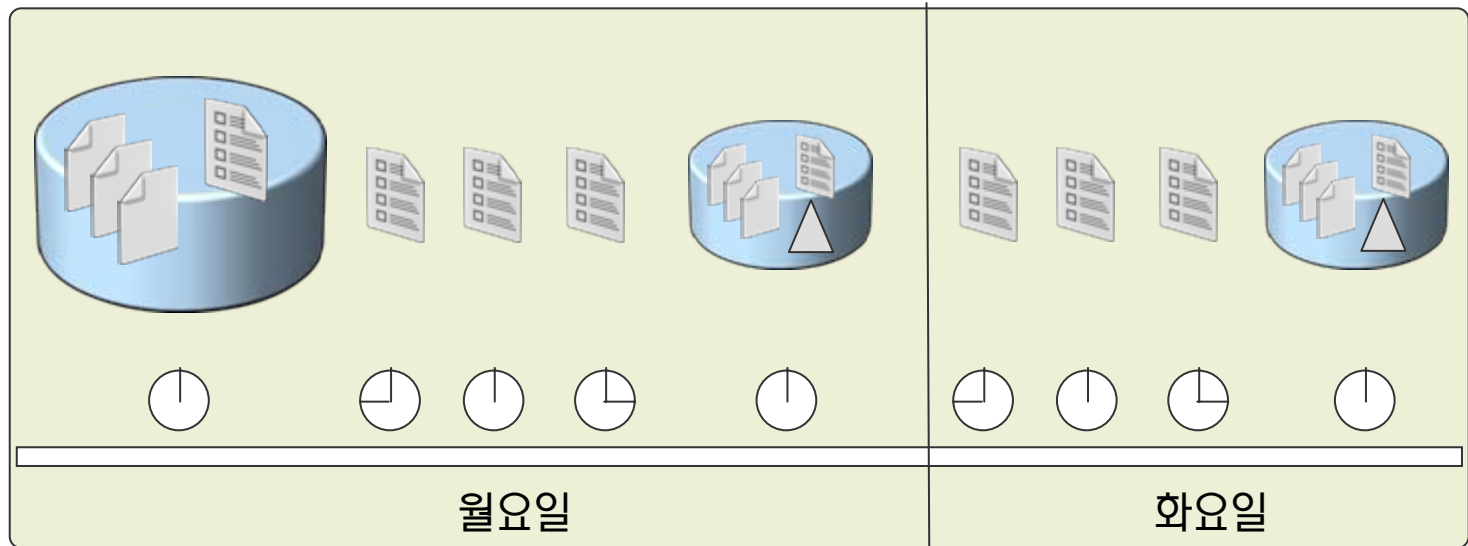
- 전체 백업을 수행하는 경우:
 - 데이터베이스 사이즈가 작은 경우
 - 데이터베이스 변경이 거의 없거나, 읽기 전용인 경우
- 복구 모델이 전체(Full)인 경우 수시로 로그를 제거해 주어야 함

트랜잭션 로그 백업 전략



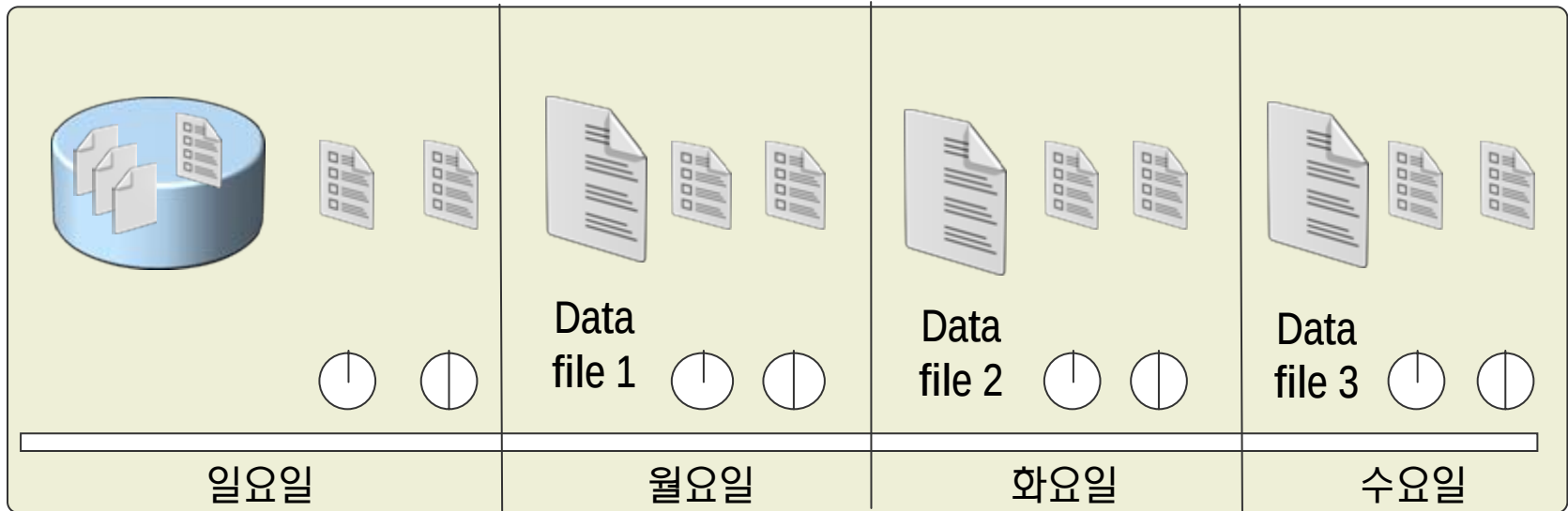
- 데이터베이스 백업과 트랜잭션 로그 백업을 같이 하는 경우:
 - 데이터베이스가 자주 변경 되는 경우
 - 전체 백업 시간이 너무 긴 경우

차등 백업 전략



- 차등 백업을 사용하는 경우:
 - 데이터베이스가 수시로 변경 되는 경우
 - 백업 시간을 단축하고자 하는 경우
- 트랜잭션 로그를 별도로 백업 받아야 함

파일/파일 그룹 백업 전략



- 파일 및 파일 그룹 백업을 받는 경우:
 - 데이터베이스 사이즈가 아주 큰 경우
 - 전체 백업을 수행하는데 시간이 많이 걸리는 경우
- 트랜잭션 로그를 별도로 백업 받아야 함
- 관리가 번거로울 수 있음

전체 백업 수행

- 데이터베이스 전체를 백업 받음
- 트랜잭션 로그 파일의 일부가 백업에 포함됨

```
BACKUP DATABASE {database_name | @database_name_var}  
  TO <backup_device> [, ...n]  
  [WITH  
    [FORMAT]  
    [[,] {INIT | NOINIT}]]  
  ]
```

트랜잭션 로그 백업 수행

- 장애가 발생한 시점으로 복원이 가능하도록 함
- 복구모델이 전체 또는 대량로그인 경우 트랜잭션 로그를 별도로 백업 받도록 함
- 수시로 트랜잭션 로그를 백업 받도록 함

```
BACKUP LOG {database_name | @database_name_var}  
  TO <backup_device> [, ...n]  
  [WITH  
    [{INIT | NOINIT}]  
  ]
```

차등 백업 수행

- 전체 백업 이후로 변경된 내용을 백업 받음
- 전체 백업 보다 백업 파일의 사이즈가 작고 수행 시간이 빠름

```
BACKUP DATABASE {database_name | @database_name_var}  
    TO <backup_device> [, ...n]  
    [WITH  
        [DIFFERENTIAL]  
    ]
```

DEMO

1. EM을 이용한 데이터베이스 백업
2. Transact-SQL을 이용한 백업
3. 여러 파일에 동시에 백업
4. 백업 파일에 암호 설정
5. 미리 백업

데이터베이스 복원

복원 진행 과정

- 복원 진행 과정
 - Data copy
 - Redo
 - Undo
- NORECOVERY 옵션과 RECOVERY 옵션을 사용함

복원 방법

- 전체 백업 또는 차등 백업으로부터의 복원
 - 데이터베이스 파일을 복원하게 됨
 - 데이터베이스 개체를 새로 생성함
- 복원 시 RECOVERY 또는 NORECOVERY 옵션을 이용해 복원 과정을 제어
- 파일의 위치를 변경하고자 하는 경우 MOVE...TO 옵션을 사용함
- 존재하는 데이터베이스를 교체하고자 하는 경우 REPLACE 옵션을 이용

```
USE master
RESTORE DATABASE AdventureWorks
FROM AWBack
WITH NORECOVERY
```

```
RESTORE DATABASE AdventureWorks
FROM AWBackDiff
WITH RECOVERY
```

트랜잭션 로그를 이용한 복원

- 트랜잭션 로그 백업으로부터 복원
- 트랜잭션 로그에 기록되어 있는 데이터베이스에 대한 변경 사항을 복원함
- 최종 로그 복원 시에만 RECOVERY 옵션을 사용해야 함

```
RESTORE DATABASE AdventureWorks  
FROM AWBack  
WITH NORECOVERY
```

```
RESTORE LOG AdventureWorks  
FROM AWBackLog  
WITH FILE = 1, NORECOVERY
```

```
RESTORE LOG AdventureWorks  
FROM AWBackLog  
WITH FILE = 2, RECOVERY
```

DEMO

1. 일반적인 백업과 복원
2. 데이터베이스 파일 복원
3. 특정 시점으로 복원

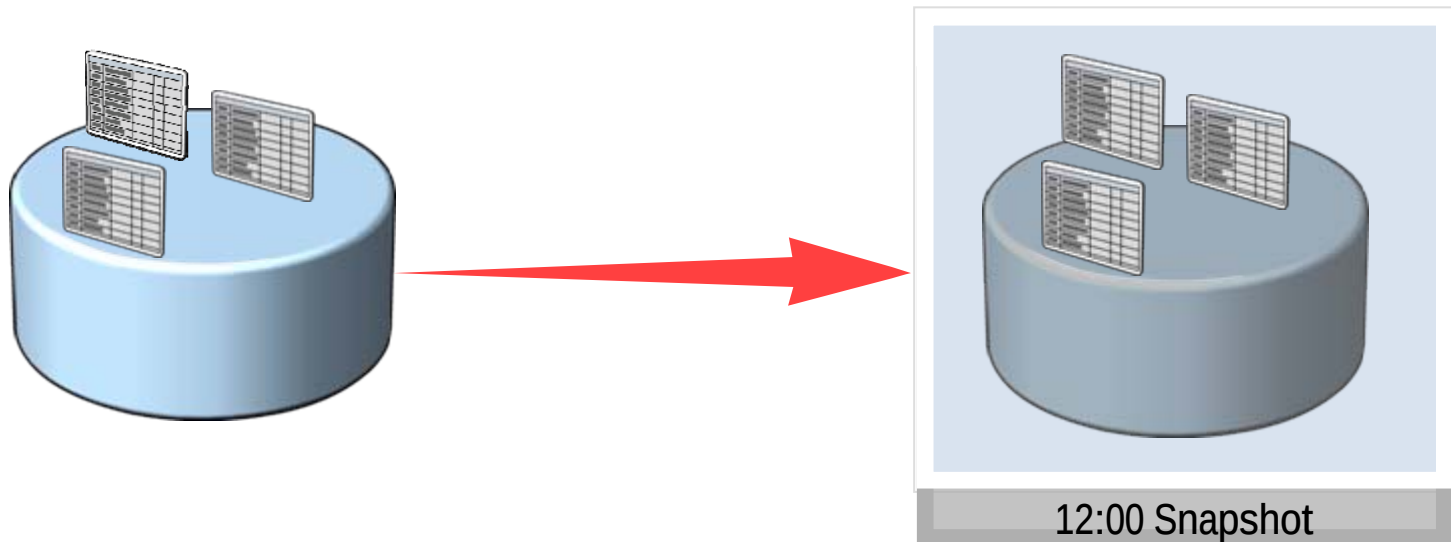
조각(Piecemeal) 복원

- 파일 그룹을 기반으로 데이터베이스를 복원함
- PRIMARY 파일 그룹 복원부터 시작함
- 이 후 보조 파일 그룹을 복원함
- 복원이 진행되는 과정에서 각각의 파일 그룹이 사용 가능하게 되어짐
- 아직 복원되지 않는 파일 그룹과 연관된 트랜잭션은 잠금을 유지함

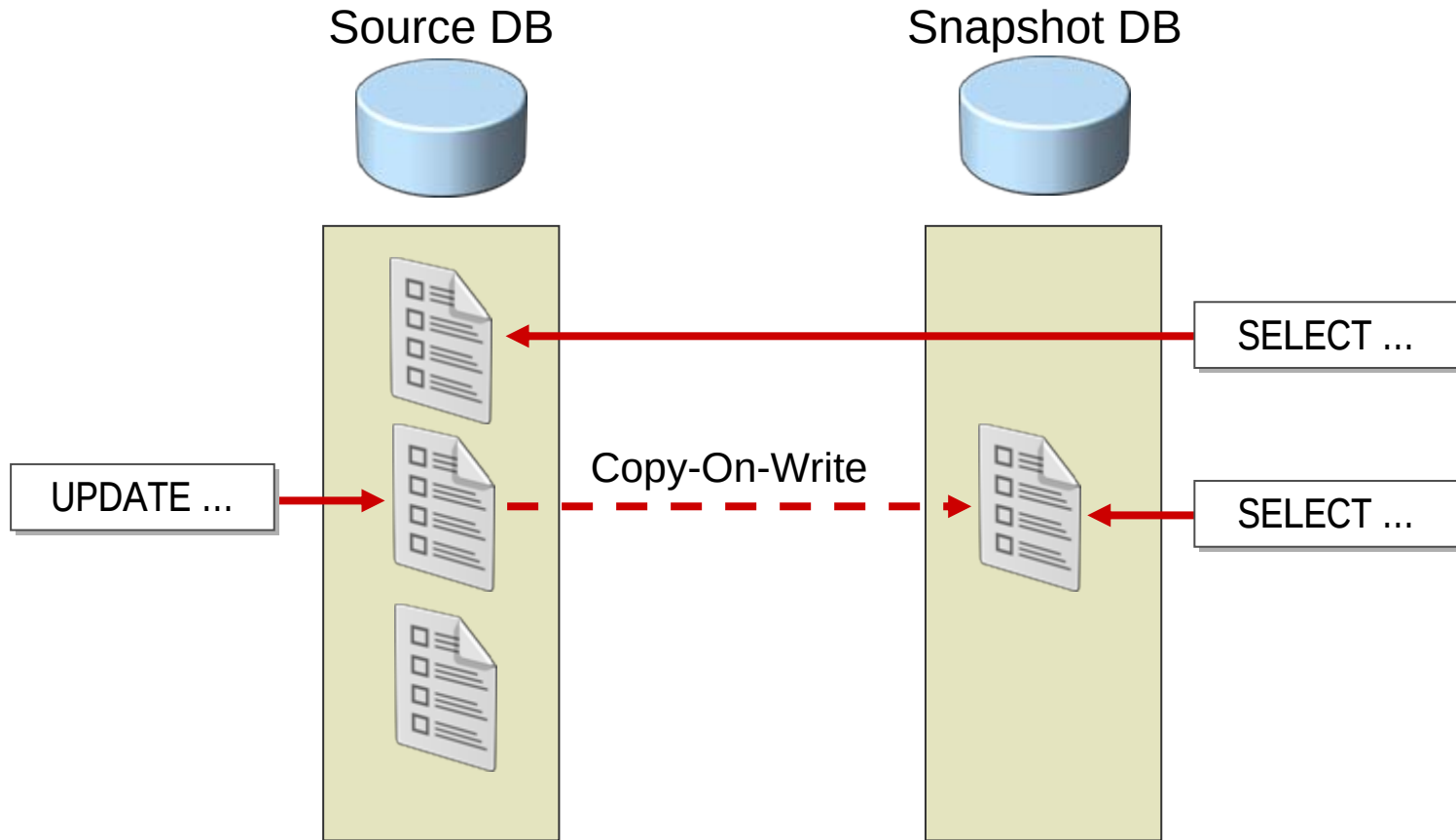
데이터베이스 스냅샷을 이용한 복원

데이터베이스 스냅샷이란

- 데이터베이스에 대한 특정 시점의 읽기전용, 일관성을 갖는 뷰
- 테스트 및 개발, 또는 보고서 작성에 활용될 수 있음
- 원본 데이터베이스와 동일한 서버에 존재해야 함



데이터베이스 스냅샷 작동 원리



데이터베이스 스냅샷을 이용한 복원

삭제 행 복원	스냅샷으로부터 원래의 데이터 복원
변경 복원	스냅샷으로부터 원래의 값 복원
삭제된 개체 복원	스냅샷으로부터 스크립트 생성 및 데이터 복사
스냅샷 시점으로 복원	스냅샷으로부터 데이터베이스 복원

DEMO

1. 조각(Piecemeal) 복원
2. 데이터베이스 스냅샷을 이용한 복원

END