

Index

아래와 같이 인덱스를 생성해보고, 질의를 수행했을때 인덱스를 사용하는지의 여부를 살펴보자.
그리고, 각 SQL 질의가 자신이 생각하고 있는 방법으로 수행되는지 확인하자.

질의가 수행되는 방법을 확인하기 위해 Trace를 통해 질의 수행시 시간이나 질의 계획 등을 살펴볼 수 있도록 설정하자.

DBA로 접속, Trace권한 부여

```
SQL> conn /as sysdba
```

test/tahiti 로 계정을 생성한다.

```
SQL> create user test identified by tahiti;
```

```
SQL> grant connect, resource to test;
```

(이미 생성되어있으면 생략)

계정에 Trace를 할 수 있는 권한을 부여한다.

```
SQL> grant plustrace to test;
```

```
SQL> conn test
```

#강의 사이트에서 users.dmp 파일을 다운받아서 import 한다.

(다운경로 : c:\나 window 사용자 root 폴더 추천)

콘솔창에서,

```
C:\> imp userid=test/tahiti file=users.dmp
```

```
C:\> sqlplus
```

```
User-name : test
```

```
Password : tahiti
```

```
SQL> select count(*) from users;
```

#50000 row 가 들어있는 users 테이블을 확인할수 있다.

```
SQL> desc users;
```

```
#####
# SQL 문장 실행 계획 및 통계 정보 보기
#####
```

1. Autotrace On

```
SQL> SET AUTOTRACE ON
```

참고, on, off로 켜고 끌 수 있음.

#2. 다음 SQL 문은 각각 어떻게 수행되는가?

```
SQL> SET AUTOTRACE TRACE EXPLAIN ;
```

#QUERY 결과는 생략하고 실행계획 및 통계정보만 확인한다.

#다음 질의는 어떻게 수행되는가?

```
SQL> SELECT * FROM users;
```

```
SQL> SELECT * FROM users ORDER BY name;
```

```
SQL> SELECT name FROM users ORDER BY name;
```

```
SQL> SELECT * FROM users ORDER BY name DESC;
```

```
SQL> SELECT name FROM users ORDER BY name DESC;
```

```
SQL> SELECT * FROM users WHERE val1 = 628;
```

```
SQL> SELECT * FROM users WHERE val1 < 628;
```

```
SQL> SELECT * FROM users WHERE val1 > 628;
```

```
SQL> SELECT COUNT(*) FROM users WHERE val1 = 628;
```

```
SQL> SELECT COUNT(*) FROM users WHERE val1 > 628;
```

```
SQL> SELECT COUNT(*) FROM users WHERE val1 < 628;
```

```
SQL> SELECT * FROM users WHERE name = 'uxysv';
```

```
SQL> SELECT * FROM users WHERE val2 = 171 and val3 = 68324;
```

```
SQL> SELECT * FROM users WHERE val2 = 171;
```

```
SQL> SELECT * FROM users WHERE val3 = 68324;
```

```
SQL> SELECT id FROM users WHERE name='uxysv';
```

#3. 인덱스 생성

```
SQL> CREATE INDEX idx_val1 ON users(val1);
```

```
SQL> CREATE INDEX idx_grade ON users(grade);
```

#Unique Index 생성

```
SQL> CREATE UNIQUE INDEX idx_name ON users(name);
```

#Composite Index 생성

```
SQL> CREATE INDEX idx_val2_val3 ON users(val2, val3);
```

#4. 다음 SQL 문은 각각 어떻게 수행되는가?

```
SQL> SELECT * FROM users;
SQL> SELECT * FROM users ORDER BY name;
SQL> SELECT name FROM users ORDER BY name;
SQL> SELECT * FROM users ORDER BY name DESC;
SQL> SELECT name FROM users ORDER BY name DESC;
#table Full scan 하는 이유는 무엇인가?
```

Index Range Scan

#다음 질의는 어떻게 수행되는가?

```
SQL> SELECT * FROM users WHERE val1 = 628;
SQL> SELECT * FROM users WHERE val1 < 628;
SQL> SELECT * FROM users WHERE val1 > 628;
SQL> SELECT COUNT(*) FROM users WHERE val1 = 628;
SQL> SELECT COUNT(*) FROM users WHERE val1 > 628;
SQL> SELECT COUNT(*) FROM users WHERE val1 < 628;
```

```
SQL> SELECT * FROM users WHERE grade = 1;
SQL> SELECT COUNT(*) FROM users WHERE grade = 1;
```

Unique Index : Composite인덱스로 처리할 수 있는 질의와 처리할 수 없는 질의는??

#다음 질의는 어떻게 수행되는가?

```
SQL> SELECT * FROM users WHERE name = 'uxysv';
SQL> SELECT id FROM users WHERE name = 'uxysv';
```

Composite Index : Composite인덱스로 처리할 수 있는 질의와 처리할 수 없는 질의는??

#다음은 어떤 차이가 있는가? 차이가 있다면 그 이유는 무엇인가?

```
SQL> SELECT * FROM users WHERE val2 = 171 and val3 = 68324;
SQL> SELECT val2, val3 FROM users WHERE val2 = 171 and val3 = 68324;
```

```
SQL> SELECT * FROM users WHERE val2 = 171;
SQL> SELECT val2, val3 FROM users WHERE val2 = 171;
```

```
SQL> SELECT val2, val3 FROM users WHERE val3 = 68324;
SQL> SELECT * FROM users WHERE val3 = 68324;
```

```
SQL> SELECT val3 FROM users WHERE val2 = 171;
```

```
SQL> SELECT val2 FROM users WHERE val3 = 68324;
```

#5. HINT

HINT INDEX: 강제로 특정 인덱스를 사용하게 해보자.

다음 질의는 어떻게 수행되는가? 결과는 어떻게 나오나?

```
SQL> SELECT /*+ INDEX (users idx_grade) */ * FROM users WHERE grade = 1;
```

```
SQL> SELECT /*+ FULL(users) */ * FROM users WHERE grade = 1;
```

```
SQL> SELECT /*+ INDEX (users idx_val1) */ * FROM users WHERE val1= 628;
```

```
SQL> SELECT /*+ FULL(users) */ * FROM users WHERE grade = 1;
```

```
SQL> SELECT /*+ INDEX (users idx_val1) */ COUNT(*) FROM users WHERE val1 = 628;
```

```
SQL> SELECT /*+ FULL (users) */ COUNT(*) FROM users WHERE val1 = 628;
```

결과의 순서는? 정렬이 되나? 수행 시간은?