대표님, [13.05.20 10:34]

create table ddl\_scripts

(

owner varchar2(30),

object\_type varchar2(30),

object\_name varchar2(30),

line number,

text varchar2(4000)

);

대표님, [13.05.20 10:35]

요렇게 table 만들어서

대표님, [13.05.20 10:35]

owner,table\_name input 값으로 받아

대표님, [13.05.20 10:35]

CREATE TABLE SCOTT.EMP

(

EMPNO NUMBER (4) NOT NULL,

ENAME VARCHAR2 (10),

JOB VARCHAR2 (9),

MGR NUMBER (4),

HIREDATE DATE,

SAL NUMBER (7,2),

COMM NUMBER (7,2),

DEPTNO NUMBER (2)

)

TABLESPACE USERS;

대표님, [13.05.20 10:36]

요렇게 insert되게 하는 procedure 만들어봐

대표님, [13.05.20 10:36]

index 도 함께

대표님, [13.05.20 10:36]

CREATE UNIQUE INDEX SCOTT.PK\_EMP

ON SCOTT.EMP

(

EMPNO ASC

)

TABLESPACE USERS;

대표님, [13.05.20 10:37]

owner,index\_name input 값으로 받아서

**Sql1) ddl\_scripts to procedure table**

create

or replace procedure proc\_ins\_tab\_ddlscript(p\_owner varchar2, p\_tab\_name varchar2) is v\_tname varchar2(30);

v\_towner varchar2(30);

v\_cname char(32);

v\_type char(20);

v\_null varchar2(10);

v\_maxcol number;

v\_virg varchar2(1);

v\_line number:=1;

begin select owner,

table\_name

into v\_towner,

v\_tname

from dba\_tab\_columns

where owner = p\_owner

and table\_name = p\_tab\_name

group by owner,

table\_name;

if v\_tname is null then raise NO\_DATA\_FOUND;

elsif v\_towner is null then raise NO\_DATA\_FOUND;

end if;

v\_virg:=',';

insert

into DDL\_SCRIPTS (OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)

values (p\_owner,

'table',

p\_tab\_name,

v\_line,

'CREATE TABLE '||p\_owner||'.'||p\_tab\_name||' (');

for rc in (select table\_name,

column\_name,

data\_type,

data\_length,

data\_precision,

data\_scale,

nullable,

column\_id

from user\_tab\_columns tc

where tc.table\_name=p\_tab\_name

order by table\_name,

column\_id) loop v\_cname:=rc.column\_name;

if rc.data\_type='VARCHAR2' then v\_type:='VARCHAR2('||rc.data\_length||')';

elsif rc.data\_type='NUMBER'

and rc.data\_precision is null

and rc.data\_scale=0 then v\_type:='INTEGER';

elsif rc.data\_type='NUMBER'

and rc.data\_precision is null

and rc.data\_scale is null then v\_type:='NUMBER';

elsif rc.data\_type='NUMBER'

and rc.data\_scale='0' then v\_type:='NUMBER('||rc.data\_precision||')';

elsif rc.data\_type='NUMBER'

and rc.data\_scale<>'0' then v\_type:='NUMBER('||rc.data\_precision||','||rc.data\_scale||')';

elsif rc.data\_type='CHAR' then v\_type:='CHAR('||rc.data\_length||')';

else v\_type:=rc.data\_type;

end if;

if rc.nullable='Y' then v\_null:='';

else v\_null:='NOT NULL';

end if;

select max(column\_id)

into v\_maxcol

from dba\_tab\_columns c

where c.table\_name=p\_tab\_name

and c.owner = p\_owner;

if rc.column\_id=v\_maxcol then v\_virg:='';

end if;

v\_line := v\_line+1;

insert

into DDL\_SCRIPTS(OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)

values(p\_owner,

'table',

p\_tab\_name,

v\_line,

v\_cname||v\_type||v\_null||v\_virg);

end loop;

v\_line := v\_line+1;

insert

into DDL\_SCRIPTS(OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)select p\_owner,

'table',

p\_tab\_name,

v\_line,

')'||'TABLESPACE '||TABLESPACE\_NAME||';'

from dba\_tables

where owner=p\_owner

and table\_name=p\_tab\_name;

exception

when NO\_DATA\_FOUND then dbms\_output.put\_line('ONWER : '||p\_owner||' 이나 TABLE : '||p\_tab\_name||' 에 해당하는 테이블이 존재하지 않습니다');

end proc\_ins\_ind\_ddlscript;

**Sql2) ddl\_scripts to procedure index**

create

or replace procedure proc\_ins\_ind\_ddlscript(p\_owner varchar2, p\_ind\_name varchar2) is v\_iname varchar2(30);

v\_iowner varchar2(30);

v\_cname char(32);

V\_DESCEND char(20);

v\_maxcol number;

v\_virg varchar2(1);

v\_line number:=1;

begin select index\_owner,

index\_name

into v\_iowner,

v\_iname

from dba\_ind\_columns

where index\_owner = p\_owner

and index\_name = p\_ind\_name

group by index\_owner,

index\_name;

if v\_iname is null then raise NO\_DATA\_FOUND;

elsif v\_iowner is null then raise NO\_DATA\_FOUND;

end if;

v\_virg:=',';

insert

into DDL\_SCRIPTS (OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)

select p\_owner,

'index',

p\_ind\_name,

v\_line,

'CREATE UNIQER INDEX '||p\_owner||'.'||p\_ind\_name||' ON '||index\_owner||'.'||table\_name||' ('

from dba\_ind\_columns

where INDEX\_NAME=p\_ind\_name

and INDEX\_OWNER = p\_owner

and rownum = 1;

for rc in (select INDEX\_OWNER,

INDEX\_NAME,

COLUMN\_NAME,

DESCEND,

COLUMN\_POSITION

from dba\_ind\_columns tc

where tc.index\_name=p\_ind\_name

order by table\_name,

column\_position) loop V\_DESCEND := rc.DESCEND;

select max(column\_position)

into v\_maxcol

from dba\_ind\_columns c

where c.index\_name=p\_ind\_name

and c.index\_owner = p\_owner;

if rc.column\_position=v\_maxcol then v\_virg:='';

end if;

v\_line := v\_line+1;

insert

into DDL\_SCRIPTS(OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)

values(p\_owner,

'index',

p\_ind\_name,

v\_line,

v\_cname||' '||V\_DESCEND||v\_virg);

end loop;

v\_line := v\_line+1;

insert

into DDL\_SCRIPTS(OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)select p\_owner,

'index',

p\_ind\_name,

v\_line,

')'||'TABLESPACE '||TABLESPACE\_NAME||';'

from dba\_indexes

where owner=p\_owner

and index\_name=p\_ind\_name;

exception

when NO\_DATA\_FOUND then dbms\_output.put\_line('ONWER : '||p\_owner||' 이나 TABLE : '||p\_ind\_name||' 에 해당하는 인덱스가 존재하지 않습니다');

end proc\_ins\_tab\_ddlscript;

**SQL3) ddl\_script to package all index and table**

**--패키지 선언부**

create or replace package pkg\_ins\_ind\_tab\_ddlscript is

procedure proc\_ins\_tab\_ddlscript(p\_owner varchar2,p\_tab\_name varchar2);

procedure proc\_ins\_ind\_ddlscript(p\_owner varchar2,p\_ind\_name varchar2);

procedure proc\_all\_ind\_ddlscripts(p\_owner varchar2,p\_tab\_name varchar2);

end pkg\_ins\_ind\_tab\_ddlscript;

**--패키지 바디부분**

create

or replace package body pkg\_ins\_ind\_tab\_ddlscript is procedure proc\_ins\_tab\_ddlscript(p\_owner varchar2, p\_tab\_name varchar2) is v\_tname varchar2(30);

v\_towner varchar2(30);

v\_cname char(32);

v\_type char(20);

v\_null varchar2(10);

v\_maxcol number;

v\_virg varchar2(1);

v\_line number:=1;

begin select owner,

table\_name

into v\_towner,

v\_tname

from dba\_tab\_columns

where owner = p\_owner

and table\_name = p\_tab\_name

group by owner,

table\_name;

if v\_tname is null then raise NO\_DATA\_FOUND;

elsif v\_towner is null then raise NO\_DATA\_FOUND;

end if;

v\_virg:=',';

insert

into DDL\_SCRIPTS (OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)

values (p\_owner,

'table',

p\_tab\_name,

v\_line,

'CREATE TABLE '||p\_owner||'.'||p\_tab\_name||' (');

for rc in (select table\_name,

column\_name,

data\_type,

data\_length,

data\_precision,

data\_scale,

nullable,

column\_id

from user\_tab\_columns tc

where tc.table\_name=p\_tab\_name

order by table\_name,

column\_id) loop v\_cname:=rc.column\_name;

if rc.data\_type='VARCHAR2' then v\_type:='VARCHAR2('||rc.data\_length||')';

elsif rc.data\_type='NUMBER'

and rc.data\_precision is null

and rc.data\_scale=0 then v\_type:='INTEGER';

elsif rc.data\_type='NUMBER'

and rc.data\_precision is null

and rc.data\_scale is null then v\_type:='NUMBER';

elsif rc.data\_type='NUMBER'

and rc.data\_scale='0' then v\_type:='NUMBER('||rc.data\_precision||')';

elsif rc.data\_type='NUMBER'

and rc.data\_scale<>'0' then v\_type:='NUMBER('||rc.data\_precision||','||rc.data\_scale||')';

elsif rc.data\_type='CHAR' then v\_type:='CHAR('||rc.data\_length||')';

else v\_type:=rc.data\_type;

end if;

if rc.nullable='Y' then v\_null:='';

else v\_null:='NOT NULL';

end if;

select max(column\_id)

into v\_maxcol

from dba\_tab\_columns c

where c.table\_name=p\_tab\_name

and c.owner = p\_owner;

if rc.column\_id=v\_maxcol then v\_virg:='';

end if;

v\_line := v\_line+1;

insert

into DDL\_SCRIPTS(OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)

values(p\_owner,

'table',

p\_tab\_name,

v\_line,

v\_cname||v\_type||v\_null||v\_virg);

end loop;

v\_line := v\_line+1;

insert

into DDL\_SCRIPTS(OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)select p\_owner,

'table',

p\_tab\_name,

v\_line,

')'||'TABLESPACE '||TABLESPACE\_NAME||';'

from dba\_tables

where owner=p\_owner

and table\_name=p\_tab\_name;

exception

when NO\_DATA\_FOUND then dbms\_output.put\_line('ONWER : '||p\_owner||' 이나 TABLE : '||p\_tab\_name||' 에 해당하는 테이블이 존재하지 않습니다');

end proc\_ins\_tab\_ddlscript;

procedure proc\_ins\_ind\_ddlscript(p\_owner varchar2, p\_ind\_name varchar2) is v\_iname varchar2(30);

v\_iowner varchar2(30);

v\_cname char(32);

V\_DESCEND char(20);

v\_maxcol number;

v\_virg varchar2(1);

v\_line number:=1;

begin select index\_owner,

index\_name

into v\_iowner,

v\_iname

from dba\_ind\_columns

where index\_owner = p\_owner

and index\_name = p\_ind\_name

group by index\_owner,

index\_name;

if v\_iname is null then raise NO\_DATA\_FOUND;

elsif v\_iowner is null then raise NO\_DATA\_FOUND;

end if;

v\_virg:=',';

insert

into DDL\_SCRIPTS (OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)select p\_owner,

'index',

p\_ind\_name,

v\_line,

'CREATE UNIQER INDEX '||p\_owner||'.'||p\_ind\_name||' ON '||index\_owner||'.'||table\_name||' ('

from dba\_ind\_columns

where INDEX\_NAME=p\_ind\_name

and INDEX\_OWNER = p\_owner

and rownum = 1;

for rc in (select INDEX\_OWNER,

INDEX\_NAME,

COLUMN\_NAME,

DESCEND,

COLUMN\_POSITION

from dba\_ind\_columns tc

where tc.index\_name=p\_ind\_name

order by table\_name,

column\_position) loop V\_DESCEND := rc.DESCEND;

select max(column\_position)

into v\_maxcol

from dba\_ind\_columns c

where c.index\_name=p\_ind\_name

and c.index\_owner = p\_owner;

if rc.column\_position=v\_maxcol then v\_virg:='';

end if;

v\_line := v\_line+1;

insert

into DDL\_SCRIPTS(OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)

values(p\_owner,

'index',

p\_ind\_name,

v\_line,

v\_cname||' '||V\_DESCEND||v\_virg);

end loop;

v\_line := v\_line+1;

insert

into DDL\_SCRIPTS(OWNER, OBJECT\_TYPE, OBJECT\_NAME, LINE, TEXT)select p\_owner,

'index',

p\_ind\_name,

v\_line,

')'||'TABLESPACE '||TABLESPACE\_NAME||';'

from dba\_indexes

where owner=p\_owner

and index\_name=p\_ind\_name;

exception

when NO\_DATA\_FOUND then dbms\_output.put\_line('ONWER : '||p\_owner||' 이나 TABLE : '||p\_ind\_name||' 에 해당하는 인데스가 존재하지 않습니다');

end proc\_ins\_ind\_ddlscript;

procedure proc\_all\_ind\_ddlscripts(p\_owner varchar2, p\_tab\_name varchar2) is

begin

proc\_ins\_tab\_ddlscript(p\_owner, p\_tab\_name);

for rt in (select index\_name

from dba\_ind\_columns

where table\_name = p\_tab\_name

group by index\_name) loop

proc\_ins\_ind\_ddlscript(p\_owner, rt.index\_name);

end loop;

end proc\_all\_ind\_ddlscripts;

end pkg\_ins\_ind\_tab\_ddlscript;