

```

1  REM Column 의 옵션
2  --1. DEFAULT OPTION
3  --2. Constraints
4
5  REM DEFAULT OPTION
6  --1. 열에 기본값을 부여할 수 있다.
7  --2. 이 옵션은 열 값이 없는 행을 삽입할 경우 열에 널 값이 입력되는 것을 방지
8  --3. 리터럴, 표현식 또는 SYSDATE 와 같은 SQL 함수는 기본값으로 사용할 수 있다.
9  --4. 기본 표현식은 해당 열의 데이터 유형과 일치해야 한다.
10 --5. ex)
11 --hiredate DATE DEFAULT SYSDATE
12 --6. 테이블을 생성할 때 혹은 스키마를 변경할 때 칼럼에 직접 적용
13 --CREATE TABLE table name
14 --(column datatype [DEFAULT expression][,...]);
15 --7. 실습1
16 /*
17 CREATE TABLE Department
18 AS
19 SELECT * FROM dept
20 WHERE 0 > 1;
21
22 ALTER TABLE Department
23 ADD
24 ( hiredate DATE DEFAULT SYSDATE);
25
26 INSERT INTO Department(deptno, hiredate)
27 VALUES(10, DEFAULT);
28 */
29 --8. 실습2
30 /*
31 CREATE TABLE Jusorok
32 (
33     bunho NUMBER(2),
34     gender CHAR(4) DEFAULT '남자'
35 );
36
37 INSERT INTO Jusorok VALUES (1, '여자');
38 INSERT INTO Jusorok VALUES (2, DEFAULT);
39 SELECT * FROM jusorok;
40 */
41
42 REM Constraints(제약조건)
43 /*
44 INSERT INTO dept
45 VALUES (10, 'TEST', 'SEOUL'); --무결성 제약조건 위배(ORA-00001)
46 --Dept 테이블에 이미 10번 부서가 존재하고 있기 때문
47
48 INSERT INTO dept
49 VALUES (NULL, NULL, 'SEOUL'); --오류(ORA-01400)
50 --NULL 을 부서번호에 삽입할 수 없다.
51 */
52
53 --1. 특징
54 --1) 사용자에게 의해 발생한 잘못된 DML 문이 실행되지 않는다.
55 --2) 제약 조건에 대한 모든 정보가 자료사전에 저장된다.
56 --3) 원할 때는 언제든지 기능을 비활성화할 수 있고, 또한 활성화 할 수 있다.
57 --4) 처리 결과가 즉시 사용자에게 보여진다.
58 --5) 하나의 칼럼에 여러 개의 제약 조건을 설정할 수 있다.
59 --6) 부적절한 데이터의 입력, 수정, 삭제를 방지할 목적
60 --7) 제약조건은 테이블레벨/컬럼레벨 제약조건이 있다.
61 --8) 제약조건은 종속성이 있을 경우 삭제를 방지한다.
62 --9) 테이블에서 삽입, 수정, 삭제를 할 때 마다 조건에 대한 규칙을 적용한다.
63 --10) 제약조건은 각각의 RDBMS 마다 다를 수 있다.
64 --11) Oracle 에서는 제약 조건 이름을 지정하지 않으면, 자동으로 SYS-Cnumber 형식으로 생성한다.
65 --12) 사용자가 해당 제약조건에 위배되는 명령을 요청했을 때, 오라클은 제약조건 이름과 함께 에러를 내보낸다.
66 --13) DESC 명령어는 NOT NULL 제약 조건을 확인할 수 있지만, 다른 제약조건을 확인할 수 없다.
67
68 --2. 제약조건의 종류
69 --1) Column Level Constraint
70 --컬럼단위로 제약조건을 부여할 때
71 --5가지의 제약조건이 모두 가능
72 --NOT NULL은 컬럼레벨에서만 부여가능
73 --Syntax
74 /*
75 column name data type
76 [CONSTRAINT constraint name] constraint type
77 */
78 --id VARCHAR2(10) CONSTRAINT testpk_id_pk PRIMARY KEY,
79 --id VARCHAR2(10) PRIMARY KEY
80
81 /*
82 CREATE TABLE Student
83 (
84     hakbun CHAR(4) CONSTRAINT Student_hakbun_pk PRIMARY KEY,
85     name VARCHAR2(20) CONSTRAINT Student_name_nn NOT NULL,
86     kor NUMBER(3) NOT NULL
87 )
88
89 SELECT OWNER, CONSTRAINT_NAME, CONSTRAINT_TYPE, TABLE_NAME
90 FROM user constraints
91 WHERE TABLE_NAME = UPPER('student');
92 */
93
94 --2) Table Level Constraint
95 --각각의 칼럼의 정의와 개별적으로 정의한다.

```

```

96      --하나 이상의 칼럼을 지정가능하다.
97      --NOT NULL을 제외한 나머지 제약조건의 정의가 가능
98      --Syntax
99  /*
100      column name  data type,
101      ...,
102      ...,
103      [CONSTRAINT constraint_name]  constraint_type
104  */
105  /*
106      id      VARCHAR2(10),
107      name    VARCHAR2(20),
108      age     NUMBER(3),
109      CONSTRAINT testpk_id_pk PRIMARY KEY(id)
110  */
111  /*
112      CREATE TABLE Student
113      (
114          hakbun    CHAR(4),
115          name      VARCHAR2(20)  CONSTRAINT Student name nn NOT NULL,
116          kor       NUMBER(3) NOT NULL,
117          CONSTRAINT Student_hakbun_pk  PRIMARY KEY(hakbun)
118      );
119  */
120
121  --3. 데이터 무결성 제약조건 5가지
122      --PRIMARY KEY(PK)
123      --FOREIGN KEY(FK)
124      --UNIQUE(UK)
125      --NOT NULL(NN)
126      --CHECK(CK)
127
128  --4. Constraints Naming Convention guide lines
129      --constraint name : tablename columnname constrainttype
130      --ex : emp empno pk,  emp deptno fk
131
132  REM PRIMARY KEY
133
134  /*
135      EMPNO      ENAME      JOB      DEPTNO
136      7499      ALLEN      SALESMAN      30
137              JONES      MANAGER      20
138              JONES      SALESMAN      10
139      --현재 EMPNO 가 UNIQUE 로 설정된 상태임
140      --UNIQUE 는 NULL 이 허용되기 때문에 추가적으로 NOT NULL이 필요
141      --그래서 PRIMARY KEY 가 필요함.
142  */
143  --1. 기본적으로 테이블마다 주 식별자는 있어야 한다.
144  --2. NOT NULL 제약조건과 NO DUPLICATE 제약조건(UNIQUE)이 부여된다.
145  --3. UNIQUE INDEX 가 자동으로 생성
146  --4. 테이블 레벨 제약조건과 컬럼 레벨 제약조건 모두 가능
147  --5. Syntax
148      --1) Column Level Constraints
149  /*
150      CREATE TABLE Test
151      (
152          id  NUMBER(3)      CONSTRAINT  test id pk  PRIMARY KEY,
153          name  VARCHAR2(20)
154      );
155  */
156  --2) Table Level Constraints
157  /*
158      CREATE TABLE Test1
159      (
160          id  NUMBER(3),
161          name  VARCHAR2(20),
162          CONSTRAINT test1_id_pk PRIMARY KEY(id)
163      );
164  */
165
166  REM FOREIGN KEY
167  --1. 자식테이블(참조하는 쪽)에서 정의한다.
168  --2. MASTER TABLE(부모테이블) vs DETAIL TABLE(자식테이블)
169      --MASTER TABLE(parent) 은 참조 당하는 쪽(예:dept table)을 의미하고 DETAIL TABLE(child)은 참조하는
170      쪽(예:emp table)을 의미한다.
171  --3. 외래키가 바라보는(참조하는) 부모테이블(마스터테이블)의 키는 PRIMARY KEY, UNIQUE KEY로 정의된 열을
172      지정할 수 있으며, 데이터타입이 일치해야 하고, NULL 일 수 있다.
173  --4. ON DELETE CASCADE를 지정하면, 부모테이블의 레코드를 삭제할 때, 참조된 행을 삭제할 수 있다.
174  --5. Syntax
175      --1) Column Level Constraint
176  /*
177      column_name  data_type
178      [CONSTRAINT constraint name]
179      REFERENCES table_name (column_name)
180      [ON DELETE CASCADE]
181  */
182      --ex) deptno  NUMBER(2) REFERENCES dept (deptno);
183
184  --2) Table Level Constraint
185  /*
186      column_name  data_type,
187      ...,
188      ...,
189      [CONSTRAINT constraint name]  FOREIGN KEY(column name)
190      REFERENCES table_name (column_name) [ON DELETE CASCADE]

```

```

189 */
190 --ex) deptno NUMBER(2),
191      --FOREIGN KEY(deptno) REFERENCES dept (deptno)
192
193 --6. FOREIGN KEY 의 주의점
194 --1) FOREIGN KEY 값은 MASTER TABLE에서 존재하는 값과 일치해야하거나 NULL이 되어야 한다.
195 --2) MASTER TABLE을 먼저 생성해야 한다.
196 --3) MASTER TABLE에 PRIMARY KEY 또는 UNIQUE KEY 로 설정된 열을 DETAIL TABLE에서 참조해야 한다.
197 --4) MASTER TABLE 과 DETAIL TABLE 에서 참조하는 쪽과 참조 당하는 쪽의 열은 자료형이 서로 일치해야 한다.
198
199 --7. 실습
200 /*
201      --부모 테이블 먼저 생성
202      CREATE TABLE dept10
203      (
204      deptno NUMBER(2) CONSTRAINT dept10_deptno_pk PRIMARY KEY,
205      dname VARCHAR2(15),
206      LOCAL VARCHAR2(1)
207      );
208
209      --자식테이블 생성
210      CREATE TABLE emp10
211      (
212      empno NUMBER(4) CONSTRAINT emp10_empno_pk PRIMARY KEY,
213      ename VARCHAR2(15),
214      deptno NUMBER(2) CONSTRAINT emp10_deptno_fk REFERENCES dept10(deptno)
215      );
216
217      CREATE TABLE dept_copy
218      (
219      deptno      NUMBER(2),
220      dname      VARCHAR2(14),
221      loc        VARCHAR2(13),
222      CONSTRAINT dept_copy_deptno_pk PRIMARY KEY(deptno)
223      );
224
225      CREATE TABLE emp_copy
226      (
227      empno      NUMBER(4),
228      ename      VARCHAR2(10),
229      hiredate   DATE,
230      deptno     NUMBER(2),
231      CONSTRAINT emp_copy_empno_pk PRIMARY KEY(empno),
232      CONSTRAINT emp_copy_deptno_fk FOREIGN KEY(deptno)
233      REFERENCES dept_copy (deptno)
234      );
235 */
236
237 REM UNIQUE KEY
238
239 /*
240      EMPNO      ENAME      JOB      DEPTNO
241      7499      ALLEN      SALESMAN      30
242      7499      JONES      MANAGER      20
243 */
244 --1. 열 또는 열 집합 모든 값들의 유일성을 보장하기 위한 키
245 --2. 중복된 값을 가질 수 없다는 것을 보증한다.
246 --3. 하나의 테이블에서 여러 칼럼에 명시할 수 있다.
247 --4. PRIMARY KEY와 유사하지만, NULL 허용이 된다는 것이 차이다.
248 --5. 자동으로 INDEX가 부여된다.
249 --6. Syntax
250 --1) Column Level Constraint
251      --column_name data_type [CONSTRAINT constraint_name] UNIQUE,
252 --2) Table Level Constraint
253      --column_name data_type,
254      --, , ,
255      --[CONSTRAINT constraint_name] UNIQUE(column_name)
256 --7. 실습
257 /*
258      CREATE TABLE dept_clone
259      (
260      deptno NUMBER(2),
261      dname VARCHAR2(20),
262      loc VARCHAR2(20),
263      CONSTRAINT dept_clone_deptno_uk UNIQUE(deptno)
264      );
265
266      CREATE TABLE unitest
267      (
268      deptno NUMBER(2) CONSTRAINT unitest_deptno_uk UNIQUE,
269      dname CHAR(14),
270      loc CHAR(13);
271      );
272      --OR
273      CREATE TABLE unitest
274      (
275      deptno NUMBER(2),
276      dname CHAR(14),
277      loc CHAR(13),
278      CONSTRAINT unitest_deptno_uk UNIQUE (deptno),
279      );
280 */
281
282 REM NOT NULL
283 --1. 값이 NULL 이 되지 않는다는 것을 보장한다.

```

```

284 --2. INSERT, UPDATE 시 NULL 을 허용하지 않겠다는 의미
285 --3. NOT NULL 제약조건이 없는 열은 기본적으로 NULL 이 허용된다.
286 --4. 반드시 컬럼레벨 제약조건에서만 지정가능
287 --5. PRIMARY KEY 는 기본적으로 NOT NULL 을 가지고 있음.
288 --6. Syntax
289 --column datatype [CONSTRAINT constraint_name] NOT NULL
290 --7. 실습
291 /*
292 CREATE TABLE dept_copy1
293 (
294     deptno    NUMBER(2),
295     dname     VARCHAR2(20) DEFAULT 'Accounting'
296             CONSTRAINT dept_copy1_dname_nn NOT NULL,
297     loc       VARCHAR2(20),
298     CONSTRAINT dept_copy1_deptno_pk PRIMARY KEY(deptno),
299     CONSTRAINT dept_copy1_loc uk UNIQUE(loc)
300 );
301
302 INSERT INTO dept_copy1(deptno, loc)
303 VALUES(20, 'Pusan');
304
305 CREATE TABLE dept14
306 (
307     deptno NUMBER(2),
308     dname  VARCHAR2(15),
309     LOCAL CHAR(1),
310     CONSTRAINT dept14_dname_nn NOT NULL(dname)
311 ); -- ERROR 발생
312
313 CREATE TABLE NULLDemo
314 (
315     deptno NUMBER(2),
316     dname VARCHAR2(10) constraint nulldemo_dname_nn NOT NULL,
317     loc VARCHAR2(10),
318     CONSTRAINT nulldemo_deptno_pk PRIMARY KEY(deptno)
319 );
320 INSERT INTO nulldemo VALUES (1, 'aaa', '');
321 INSERT INTO nulldemo VALUES (2, 'bbb', null);
322 INSERT INTO nulldemo VALUES (3, NULL, 'seoul');
323 */
324
325 REM CHECK
326 --1. 행이 만족해야 하는 조건을 정의하는 것이다.
327 --2. 다음과 같은 표현식(expression)은 허용되지 않는다.
328     --1) CURRVAL, NEXTVAL, LEVEL, ROWNUM
329     --2) SYSDATE, UID, USER 함수에 대한 호출
330     --3) 다른 행에 있는 값을 참조할 수 있는 질의
331     --4) 예약어
332 --3. Syntax
333     --1) Column Level Constraint
334         --column_name data_type [CONSTRAINT constraint_name] CHECK(condition)
335     --2) Table Level Constraint
336         --column name data type,
337         --,,,,
338         --[CONSTRAINT constraint_name] CHECK(condition)
339 --4. 실습
340 /*
341 CREATE TABLE dept_copy
342 (
343     deptno    NUMBER(2),
344     dname     VARCHAR2(20),
345     loc       VARCHAR2(20),
346     CONSTRAINT dept_copy_deptno_pk PRIMARY KEY(deptno),
347     CONSTRAINT dept_copy_deptno_ck CHECK(deptno IN(10,20,30,40,50))
348 );
349
350 --사원번호, 사원명, 급여, 성별 4개의 칼럼을 갖는 테이블을 생성하시오,
351 --단 사원번호가 기본키로, 사원명은 NOT NULL로, 급여는 500에서 5000사이의 값만
352 --저장할 수 있고, 성별은 남자는 M, 여자는 F 둘중의 하나만 저장할 수 있어야 한다.
353 CREATE TABLE emp2
354 (
355     empno    NUMBER(4) CONSTRAINT emp_empno_pk PRIMARY KEY,
356     ename    VARCHAR2(10) CONSTRAINT emp_ename_nn NOT NULL,
357     sal      NUMBER(7,2) CONSTRAINT emp_sal_ck
358             CHECK(sal BETWEEN 500 AND 5000),
359     gender   CHAR(1) CONSTRAINT emp_gender_ck
360             CHECK(gender IN('M', 'F'))
361 );
362
363 CREATE TABLE zippost
364 (
365     post1    CHAR(3),
366     post2    CHAR(3),
367     address  VARCHAR2(100)
368             CONSTRAINT zippost_address_nn NOT NULL,
369     CONSTRAINT zippost_post_pk PRIMARY KEY(post1, post2)
370 );
371
372 CREATE TABLE member
373 (
374     id       NUMBER(4),
375     name     VARCHAR2(10) CONSTRAINT member_name_nn NOT NULL,
376     gender   CHAR(1),
377     jumin1   CHAR(6),
378     jumin2   CHAR(7),

```

```

379         tel          CHAR(13),
380         post1        CHAR(3),
381         post2        CHAR(3),
382         address VARCHAR2(100),
383         CONSTRAINT member_id_pk PRIMARY KEY(id),
384         CONSTRAINT member_gender_ck CHECK(gender IN ('1', '2')),
385         CONSTRAINT member_tel_uk UNIQUE(tel),
386         CONSTRAINT member_post_fk FOREIGN KEY(post1, post2)
387             REFERENCES zippost(post1, post2)
388     );
389
390 */
391
392 REM DICTIONARY 에서 제약 조건 검색하기
393 --1. DESC 명령어는 NOT NULL 제약 조건을 확인할 수 있지만, 다른 제약조건을 확인할 수 없다.
394 --2. 오라클은 USER CONSTRAINTS 데이터 디렉터리 뷰로 제약 조건에 관한 정보를 알려준다.
395
396 --DESC USER_CONSTRAINTS;
397 --OWNER : 소유한 사용자명
398 --CONSTRAINT NAME
399 --CONSTRAINT TYPE
400 --P : PRIMARY KEY
401 --R : FOREIGN KEY
402 --U : UNIQUE
403 --C : CHECK, NOT NULL --> NULL 값을 체크하기 때문에 C 이다.
404 --TABLE NAME
405 --SEARCH CONDITION : CHECK 제약 조건일 때 어떤 내용이 조건으로 사용되었는지 설명
406 --R_CONSTRAINT_NAME : FOREIGN KEY인 경우 어떤 PRIMARY KEY를 참조했는지에 대한 정보
407
408 --SELECT constraint_name, table_name, status, constraint_type
409 --FROM user constraints;
410
411 --USER CONS COLUMNS
412 --어떤 칼럼에 어떤 제약 조건이 지정되었는지 알려주는 데이터 디렉터리
413 --DESC USER CONS COLUMNS
414 --COL OWNER FOR A10
415 --COL TABLE_NAME FOR A15
416 --COL COLUMN_NAME FOR A15
417 --SELECT * FROM USER CONS COLUMN;
418
419 REM 제약 조건 추가
420 --1. 제약 조건의 추가, 삭제는 가능하지만, 변경은 불가능
421 --2. 제약 조건의 활성화, 비활성화 가능
422 --3. NOT NULL 제약 조건은 MODIFY 절을 이용하여 추가
423 --4. NOT NULL 은 Column Level 에서만 가능
424 --5. Syntax
425 --ALTER TABLE table_name
426 --ADD [CONSTRAINT constraint_name] constraint_type(column);
427 --6. 실습
428 /*
429
430     DROP TABLE emp1;
431
432     CREATE TABLE emp1
433     AS
434     SELECT empno, ename, job, deptno
435     FROM emp;
436
437     ALTER TABLE emp1
438     ADD
439     (
440         CONSTRAINT emp1_deptno_fk FOREIGN KEY(deptno)
441         REFERENCES dept(deptno)
442     );
443
444     CREATE TABLE emp_clone
445     AS
446     SELECT empno, ename, job
447     FROM emp
448     WHERE deptno = 10;
449
450     ALTER TABLE emp_clone
451     ADD CONSTRAINT emp_clone_empno_pk PRIMARY KEY(empno);
452
453     ALTER TABLE emp_clone
454     MODIFY
455     (
456         ename VARCHAR2(10) CONSTRAINT emp_clone_nn NOT NULL
457     );
458 */
459
460 --1. 아래와 같이 dept 및 emp 테이블의 제약조건의 이름을 나열하는 질의를 작성하시오.
461 -- CONSTRAINT NAME
462 -- -----
463 -- dept_deptno_pk
464 -- emp empno pk
465 -- emp deptno fk
466 -- emp_mgr_fk
467
468 --SELECT constraint name
469 --FROM user_constraints
470 --WHERE table_name IN ('EMP', 'DEPT');
471
472 --2. 사원테이블을 emp_copy라는 테이블을 생성하시오. 사원이름에 UNIQUE키를 부여하시오.
473 --CREATE TABLE emp_copy

```

```

474      --AS
475      --SELECT empno, ename, sal
476      --FROM emp
477      --WHERE deptno = 20;
478
479      --ALTER TABLE emp_copy
480      --ADD CONSTRAINT emp_copy_ename_uk UNIQUE (ename);
481
482 REM 제약조건의 삭제
483 --1. ALTER TABLE의 DROP 절을 사용하여 기존의 테이블에 있는 제약조건을 삭제할 수 있다.
484 --2. Syntax
485      --ALTER TABLE table name
486      --DROP CONSTRAINT constraint name
487      --OR
488      --DROP [{ PRIMARY KEY(column_name) | FOREIGN KEY(column_name) | UNIQUE(column_name) }]
489 --3. Guide Lines
490      --1) 제약 조건을 삭제하면 USER CONSTRAINTS, USER CONS COLUMN 데이터 사전 뷰에서 제약 조건 이름이 삭제.
491      --2) CASCADE 는 모든 종속적인 제약 조건을 모두 삭제한다.
492 --4. 실습
493 /*
494      ALTER TABLE emp_copy
495      DROP CONSTRAINT emp_copy_ename_uk;
496
497      ALTER TABLE emp copy
498      ADD PRIMARY KEY(empno);
499
500      ALTER TABLE emp copy
501      DROP PRIMARY KEY;
502
503      ALTER TABLE emp1
504      DROP CONSTRAINT emp1_ename_nn;
505 */
506
507
508 REM 제약 조건 비활성화/활성화
509
510 /*
511      --1)DEPT1 table 생성 in reference with DEPT table
512      --2)emp1 table 생성 in reference with EMP table
513      --3)DEPT1 table의 deptno 에 PK 줄 것
514      --4)emp1 table 의 deptno 에 FK 줄 것
515      --5)dept1 table에서 10번 부서 삭제 -- 에러
516      --부서번호 10번을 dept1 테이블에서 삭제하기 위해서는 아래와 같이 해야한다.
517      --1)emp1에서 10번 부서에 근무하는 사원을 삭제
518      --2)dept1에서 10번 부서를 삭제
519      --OR
520      --1)emp1에서 Foreign Key 삭제 후 dept1에서 10번 부서 삭제
521 */
522 --1. ALTER TABLE을 이용하여 삭제 없이 제약 조건을 비활성화(DISABLE)/활성화(ENABLE) 가능
523 --2. 종속적인 무결성 제약 조건을 비활성화 하기 위하여 CASCADE를 사용
524      --부모 테이블의 제약 조건을 비활성화시키면서 이를 참조하고 있는 자식 테이블의 제약 조건까지 함께
525      비활성화시키기 위한 방법
526      --ex) dept 테이블의 기본 키를 비활성화하려고 해도 emp 테이블에서 이를 외래 키로 설정하고 있기 때문에
527      비활성화가 안됨. emp 테이블의 외래 키를 비활성화한 후에 dept 테이블의 기본 키를 비활성화해야 함. 하지만
528      CASCADE 를 사용하면 두번에 걸쳐서 비활성화할 필요 없음.
529 --3. 제약조건 비활성화 뿐만 아니라 삭제에도 CASCADE 이용 가능
530 --4. Syntax
531      --ALTER TABLE table_name
532      --DISABLE CONSTRAINT constraint_name [CASCADE];
533
534      --ALTER TABLE table_name
535      --ENABLE CONSTRAINT constraint_name;
536 --4. 실습
537      --1) 위의 문제에서 추가했던 emp_ename_uk 제약조건을 비활성화하시오.
538      --ALTER TABLE emp
539      --DISABLE CONSTRAINT emp_ename_uk;
540
541      --SELECT CONSTRAINT_NAME, CONSTRAINT TYPE, TABLE_NAME,
542      --      R_CONSTRAINT_NAME, STATUS
543      --FROM USER_CONSTRAINTS
544      --WHERE TABLE_NAME = 'EMP';
545
546      --2)
547      --ALTER TABLE emp_copy
548      --ADD CONSTRAINT emp_copy_ename_uk UNIQUE (ename);
549
550      --ALTER TABLE emp_copy
551      --DISABLE CONSTRAINT emp_copy_ename_uk;
552
553      --ALTER TABLE emp_copy
554      --ENABLE CONSTRAINT emp_copy_ename_uk;
555
556      --3)
557      --CREATE TABLE emp_clone
558      --AS
559      --SELECT * FROM emp;
560
561      --CREATE TABLE dept_clone
562      --AS
563      --SELECT * FROM dept;
564
565      --ALTER TABLE dept_clone
566      --ADD CONSTRAINT dept_clone_deptno_pk PRIMARY KEY(deptno);
567
568      --ALTER TABLE emp_clone

```

```

566      --ADD CONSTRAINT emp_clone empno pk PRIMARY KEY(empno);
567
568      --ALTER TABLE emp_clone
569      --ADD CONSTRAINT emp_clone_deptno_fk FOREIGN KEY(deptno)
570      --REFERENCES dept_clone(deptno);
571
572      --SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE, TABLE_NAME,
573      --      R_CONSTRAINT_NAME, STATUS
574      --FROM USER CONSTRAINTS
575      --WHERE TABLE_NAME = 'EMP_CLONE';
576
577      --INSERT INTO emp_clone(empno, ename, sal, deptno)
578      --VALUES(9999, 'SUJAN', 5000, 10);
579
580      --DELETE FROM dept_clone
581      --WHERE deptno = 10;      --error
582
583      --ALTER TABLE emp_clone
584      --DISABLE CONSTRAINT emp_clone_deptno_fk;
585
586      --SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE, TABLE_NAME,
587      --      R_CONSTRAINT_NAME, STATUS
588      --FROM USER CONSTRAINTS
589      --WHERE TABLE_NAME = 'EMP_CLONE';
590
591      --DELETE FROM dept_clone
592      --WHERE deptno = 10;      --가능
593
594      --다시 활성화
595      --ALTER TABLE emp_clone
596      --ENABLE CONSTRAINT emp_clone_deptno_fk; --error
597      --부서 테이블 10번 부서가 삭제된 상태에서 외래키 제약 조건 활성화 안됨
598
599      --INSERT INTO dept_clone
600      --VALUES (10, 'ACCOUNTING', 'NEW YORK');
601
602      --ALTER TABLE emp_clone
603      --ENABLE CONSTRAINT emp_clone_deptno_fk;
604
605      --CASCADE 실습
606      --dept_clone 에서 기본 키 제약 조건 비활성화
607      --ALTER TABLE DEPT_CLONE
608      --DISABLE PRIMARY KEY;      --error
609
610      --ALTER TABLE DEPT_CLONE
611      --DISABLE PRIMARY KEY CASCADE;
612
613      --SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE, TABLE_NAME,
614      --      R_CONSTRAINT_NAME, STATUS
615      --FROM USER CONSTRAINTS
616      --WHERE TABLE_NAME = 'EMP_CLONE';
617
618      --ALTER TABLE DEPT_CLONE
619      --DROP PRIMARY KEY;      --error
620
621      --ALTER TABLE DEPT_CLONE
622      --DROP PRIMARY KEY CASCADE;

```