

SQL 작성하기 정리

기본 SQL 작성하기 정리

1. DDL 활용 : 테이블의 구조와 제약 조건을 생성, 삭제하고 수정하는 DDL(Data Definition Language) 명령문을 작성할 수 있다.

- ✓ DDL(Data Definition Language)은 '데이터를 정의하는 언어'로서, 보다 엄밀하게 말하면 '데이터를 담는 그릇을 정의하는 언어'이며, 이러한 그릇을 DBMS에서는 오브젝트라고 한다. DDL을 통해 정의할 수 있는 대상, 오브젝트 유형은 다음과 같다.

<표 1-1> DDL 대상

DDL 대상	설명	비고
스키마(Schema)	- DBMS 특성과 구현 환경을 감안한 데이터 구조 - 직관적으로 하나의 데이터베이스로 이해 가능	DBMS마다 차이
도메인(Domain)	- 속성의 데이터 타입과 크기, 제약 조건 등을 지정한 정보 - 속성이 가질 수 있는 값의 범위로 이해 가능	예를 들어, 주소를 VARCHAR(120)로 정의
테이블(Table)	- 데이터 저장 공간	본 학습 대상
뷰(View)	- 하나 이상의 물리 테이블에서 유도되는 가상의 논리 테이블	학습 2-2 참조
인덱스(Index)	- 검색을 빠르게 하기 위한 데이터 구조	학습 2-1 참조

기본 SQL 작성하기 정리

1. DDL 활용

❖ DDL 조작 방법

- ✓ 오브젝트를 생성, 변경 그리고 제거하기 위해 다음과 같은 명령어를 사용한다.

<표 1-2> DDL 명령어

구분	DDL 명령어	내용
생성	CREATE	데이터베이스 오브젝트 생성
변경	ALTER	데이터베이스 오브젝트 변경
삭제	DROP	데이터베이스 오브젝트 삭제
	TRUNCATE	데이터베이스 오브젝트 내용 삭제

기본 SQL 작성하기 정리

1. DDL 활용

1) DDL 활용 : 데이터베이스를 구축하기 위해 스키마, 테이블, 도메인, 인덱스, 뷰와 같은 오브젝트에 대한 DDL 적용이 필요하나, 본 학습에서는 테이블만을 대상으로 한다.

① 테이블 생성

✓ 테이블 생성을 위한 DDL 사용 방법은 다음과 같이 두 종류로 분류할 수 있다.

<표 1-3> 테이블 생성 SQL문

구분	문법
신규 생성	<pre>CREATE TABLE 테이블이름 (열이름 데이터 타입 [DEFAULT 값] [NOT NULL] {,열이름 데이터 타입 [DEFAULT 값] [NOT NULL] }* [PRIMARY KEY (열 리스트),] {[FOREIGN KEY (열 리스트) REFERENCES 테이블이름 [(열이름)] [ON DELETE 옵션] [ON UPDATE 옵션]], }* [CHECK (조건식) UNIQUE(열이름)]) ;</pre>
다른 테이블 정보를 이용한 테이블 생성 ¹⁾	<pre>CREATE TABLE 테이블이름 AS SELECT 문;</pre>

기본 SQL 작성하기 정리

1. DDL 활용

1) DDL 활용 : 데이터베이스를 구축하기 위해 스키마, 테이블, 도메인, 인덱스, 뷰와 같은 오브젝트에 대한 DDL 적용이 필요하나, 본 학습에서는 테이블만을 대상으로 한다.

① 테이블 생성

✓ 테이블 생성을 위한 DDL 사용 방법은 다음과 같이 두 종류로 분류할 수 있다.

기본 SQL 작성하기 정리

1. Oracle

❖ 계정이 잠겨서 에러가 날 경우 해결방법

```
SQL> conn scott/1234  
ERROR:  
ORA-28000: the account is locked
```

경고: 이제는 ORACLE에 연결되어 있지 않습니다.
SQL> _

```
SQL> alter user scott  
2 identified by 1234  
3 account unlock;
```

사용자가 변경되었습니다.

```
SQL> _
```

기본 SQL 작성하기 정리

1. Oracle

❖ 사용자 만들기

```
SQL> create user hong1234 identified by 1234  
2 default tablespace users  
3 temporary tablespace temp;
```

사용자가 생성되었습니다.

```
SQL>
```

❖ 사용자 권한 부여

```
SQL> grant connect, resource to hong1234;
```

권한이 부여되었습니다.

```
SQL> grant unlimited tablespace to hong1234;
```

권한이 부여되었습니다.

```
SQL>
```

기본 SQL 작성하기 정리

2. Table 생성

```
SQL> CREATE TABLE EMP2 (  
  2     EMPNO          NUMBER    PRIMARY KEY,  
  3     NAME           VARCHAR2(30) NOT NULL,  
  4     BIRTHDAY       DATE,  
  5     DEPTNO         VARCHAR2(06) NOT NULL,  
  6     EMP_TYPE       VARCHAR2(30),  
  7     TEL            VARCHAR2(15),  
  8     HOBBY          VARCHAR2(30),  
  9     PAY            NUMBER,  
 10     POSITION        VARCHAR2(30),  
 11     PEMPNO         NUMBER  
 12 );
```

테이블이 생성되었습니다.

기본 SQL 작성하기 정리

3. DESC 로 테이블 구조 파악하기

SQL> desc emp2;

SQL> desc emp2;

이름

널?

유형

EMPNO

NOT NULL NUMBER

NAME

NOT NULL VARCHAR2(30)

BIRTHDAY

DATE

DEPTNO

NOT NULL VARCHAR2(6)

EMP TYPE

VARCHAR2(30)

TEL

VARCHAR2(15)

HOBBY

VARCHAR2(30)

PAY

NUMBER

POSITION

VARCHAR2(30)

PEMNO

NUMBER

기본 SQL 작성하기 정리

4. 해당 사용자가 만든 모든 테이블 조회하기

SQL>select * from tab; // mysql 은 mysql>show tables;

```
SQL> select * from tab;
```

TNAME	TABTYPE	CLUSTERID
BIN\$ni3s6PMTQJyAeh8et6Lwjw==\$0	TABLE	
BONUS	TABLE	
CAR_MEMBER	TABLE	
COURSE_TBL	TABLE	
DEPT	TABLE	
DEPTT	TABLE	
DEPT_TABLE	TABLE	
EMP	TABLE	
EMPLOYEES	TABLE	
LECTURER_TBL	TABLE	
MEMBER	TABLE	

TNAME	TABTYPE	CLUSTERID
ORDERTBL	TABLE	
ORDERTBL1	TABLE	
SALARIES	TABLE	
SALGRADE	TABLE	
TEST1	TABLE	

16 개의 행이 선택되었습니다.

```
SQL> _
```

기본 SQL 작성하기 정리

3. Data insert

```
SQL> INSERT INTO EMP2 VALUES (19900101,'Kurt Russell',TO_DATE('19640125','YYYYMMDD'),'0001','Permanent employee','054)223-0001','music',100000000,'Boss',null);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19960101,'AL Pacino',TO_DATE('19730322','YYYYMMDD'),'1000','Permanent employee','02)6255-8000','reading',72000000,'Department head',19900101);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19970201,'Woody Harrelson',TO_DATE('19750415','YYYYMMDD'),'1000','Permanent employee','02)6255-8005','Fitness',50000000,'Section head',19960101);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19930331,'Tommy Lee Jones',TO_DATE('19760525','YYYYMMDD'),'1001','Perment employee','02)6255-8010','bike',60000000,'Deputy department head',19960101);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19950303,'Gene Hackman',TO_DATE('19730615','YYYYMMDD'),'1002','Perment employee','02)6255-8020','Marathon',56000000,'Section head',19960101);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19966102,'Kevin Bacon',TO_DATE('19720705','YYYYMMDD'),'1003','Perment employee','052)223-4000','Music',75000000,'Department head',19900101);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19930402,'Hugh Grant',TO_DATE('19720815','YYYYMMDD'),'1004','Perment employee','042)998-7005','Climb',51000000,'Section head',19966102);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19960303,'Keanu Reeves',TO_DATE('19710925','YYYYMMDD'),'1005','Perment employee','031)564-3340','Climb',35000000,'Deputy Section chief',19966102);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19970112,'Val Kilmer',TO_DATE('19761105','YYYYMMDD'),'1006','Perment employee','054)223-4500','Swim',68000000,'Department head',19900101);
```

1 개의 행이 만들어졌습니다.

```
SQL> INSERT INTO EMP2 VALUES (19960212,'Chris O'Donnell',TO_DATE('19721215','YYYYMMDD'),'1007','Perment employee','054)223-4600',null,49000000,'Section head',19970112);
```

1 개의 행이 만들어졌습니다. ■

기본 SQL 작성하기 정리

4. SELECT 명령어

(1) 모든 칼럼 조회하기

SQL>select * from emp2;

SQL> select * from emp2;

EMPNO	NAME	BIRTHDAY	DEPTNO	EMP_TYPE	TEL	HOBBY	PAY	POSITION	PEMPNO
19900101	Kurt Russell	64/01/25	0001	Permanent employee	054)223-0001	music	10000000	Boss	
19960101	AL Pacino	73/03/22	0000	Permanent employee	02)6255-8000	reading	72000000	Department head	19900101
19930331	Tommy Lee Jones	76/05/25	0001	Permanent employee	02)6255-8010	bike	60000000	Deputy department head	19960101
19950303	Gene Hackman	73/06/15	0002	Permanent employee	02)6255-8020	Marathon	56000000	Section head	19960101
19930402	Hugh Grant	72/06/15	0004	Permanent employee	042)998-7005	Climb	51000000	Section head	19960102
19960212	Chris O'Donnell	72/12/15	0007	Permanent employee	054)223-4600		49000000	Section head	19970112
19970112	Val Kilmer	76/11/05	0006	Permanent employee	054)223-4500	Swim	68000000	Department head	19900101
19960303	Keanu Reeves	71/09/25	0005	Permanent employee	031)534-3340	Climb	35000000	Deputy Section chief	19960102
19960102	Kevin Bacon	72/07/05	0003	Permanent employee	052)223-4000	Music	75000000	Department head	19900101
19970201	Woody Harrison	75/04/15	0000	Permanent employee	02)6255-8005	Fitness	50000000	Section head	19960101

10 개의 행이 선택되었습니다.

SQL> █

기본 SQL 작성하기 정리

4. SELECT 명령어

(2) 원하는 칼럼만 조회하기

SQL>select empno,name from emp2;

```
SQL> select empno,name from emp2;
```

EMPNO	NAME
-------	------

19900101	Kurt Russell
19960101	AL Pacino
19930331	Tommy Lee Jones
19950303	Gene Hackman
19930402	Hugh Grant
19960212	Chris O'Donnell
19970112	Val Kilmer
19960303	Keanu Reeves
19966102	Kevin Bacon
19970201	Woody Harrelson

10 개의 행이 선택되었습니다.

```
SQL> _
```

기본 SQL 작성하기 정리

4. SELECT 명령어

(2) 원하는 칼럼만 조회하기

SQL>select empno,name from emp2;

```
SQL> select emp_type from emp2;
```

```
EMP_TYPE
```

```
-----  
Permanent employee  
Permanent employee  
Perment employee  
Perment employee  
Perment employee  
Perment employee  
Perment employee  
Perment employee  
Perment employee  
Permanent employee
```

10 개의 행이 선택되었습니다.

```
SQL>
```

기본 SQL 작성하기 정리

4. SELECT 명령어

(3) DISTINCT 명령어 - 중복된 값을 제거하고 출력하기

SQL>select distinct emp_type from emp2;

```
SQL> select distinct emp_type from emp2;
```

```
EMP_TYPE
```

```
-----
```

```
Perment employee
```

```
Permanent employee
```

```
SQL>
```

기본 SQL 작성하기 정리

4. SELECT 명령어

(4) BETWEEN 연산자를 사용하여 emp2 테이블에서 pay 가 30000000 과 60000000 사이인 사람들의 empno, name, pay 을 출력하세요

SQL>select empno, name, pay from emp2 where pay between 30000000 and 60000000;

```
SQL> select empno, name, pay from emp2 where pay between 30000000 and 60000000;
```

EMPNO	NAME	PAY
19930331	Tommy Lee Jones	60000000
19950303	Gene Hackman	56000000
19930402	Hugh Grant	51000000
19960212	Chris O'Donnell	49000000
19960303	Keanu Reeves	35000000
19970201	Woody Harrelson	50000000

6 개의 행이 선택되었습니다.

```
SQL> █
```


기본 SQL 작성하기 정리

4. SELECT 명령어

(5) IN 연산자로 여러 조건을 간편하게 검색하기

SQL>select empno, name, deptno from emp2 where deptno in(1000,1001);

```
SQL> select empno, name, deptno from emp2 where deptno in(1000,1001);
```

EMPNO	NAME	DEPTNO
19960101	AL Pacino	1000
19930331	Tommy Lee Jones	1001
19970201	Woody Harrelson	1000

```
SQL> _
```

기본 SQL 작성하기 정리

4. SELECT 명령어

(6) LIKE 연산자로 비슷한 것들 모두 찾기

SQL>select empno, name, pay from emp2 where pay like '5%';

```
SQL> select empno, name, pay from emp2 where pay like '5%';
```

EMPNO	NAME	PAY
19950303	Gene Hackman	56000000
19930402	Hugh Grant	51000000
19970201	Woody Harrelson	50000000

```
SQL>
```

기본 SQL 작성하기 정리

4. SELECT 명령어

(7) 정렬하여 출력하기 - ORDER BY 절 사용하기

- 한 글: 가 , 나 , 다 , 라
- 영 어: A , B , C , D.....
- 숫 자: 1 , 2 , 3 , 4.....
- 날 짜: 예전 날짜부터 시작해서 최근 날짜로 정렬됩니다.

SQL>select empno, name, pay from emp2 order by name;

```
SQL> select empno, name, pay from emp2 order by name;
```

EMPNO	NAME	PAY
19960101	AL Pacino	72000000
19960212	Chris O'Donnell	49000000
19950303	Gene Hackman	56000000
19930402	Hugh Grant	51000000
19960303	Keanu Reeves	35000000
19966102	Kevin Bacon	75000000
19900101	Kurt Russell	100000000
19930331	Tommy Lee Jones	60000000
19970112	Val Kilmer	68000000
19970201	Woody Harrelson	50000000

10 개의 행이 선택되었습니다.

```
SQL> _
```

기본 SQL 작성하기 정리

[다양한 오라클 명령어 종류들]

- DML (Data Manipulation Language) : INSERT(입력) , UPDATE(변경) , DELETE(삭제) , MERGE(병합)
- DDL (Data Definition Language) : CREATE (생성) , ALTER (수정) , TRUNCATE (잘라내기) , DROP (삭제)
- DCL (Data Control Language) : GRANT (권한 주기) , REVOKE (권한 뺏기)
- TCL (Transaction Control Language): COMMIT (확정) , ROLLBACK (취소)
- SELECT : 어떤 분류에서는 DQL (Data Query Language 라고 하기도 합니다

기본 SQL 작성하기 정리

❖ DDL 명령

1. CREATE - 새로 생성하라 !

1) 일반 테이블 생성하기

```
oracle@localhost:~  
SCOTT>  
SCOTT>CREATE TABLE new_table  
2 ( no      NUMBER(3),  
3   name   VARCHAR2(10),  
4   birth  DATE );  
  
Table created.  
SCOTT>
```



NEW_TABLE		
NO	NAME	BIRTH

기본 SQL 작성하기 정리

❖ DDL 명령

2) 기본 입력 값을 설정하면서 생성하기

```
oracle@localhost:~  
SCOTT>CREATE TABLE tt02  
2 ( no          NUMBER(3,1)    DEFAULT 0,  
3   name        VARCHAR2(10)   DEFAULT 'NO Name' ,  
4   hiredate     DATE           DEFAULT SYSDATE) ;  
  
Table created.  
SCOTT>
```

```
oracle@localhost:~  
SCOTT>  
SCOTT>INSERT INTO tt02 (no) VALUES(1);  
  
1 row created.  
  
SCOTT>SELECT * FROM tt02 ;  
  
      NO NAME      HIREDATE  
-----  
1 NO Name      01-AUG-13  
  
SCOTT>
```

기본 SQL 작성하기 정리

❖ DDL 명령

3) 테이블 복사하기 (CTAS 라고도 합니다)

(1) 모든 컬럼 다 복사하기 / 특정 컬럼 만 복사하기

```
SQL> create table emp3  
2 as  
3 select * from emp2;
```

테이블이 생성되었습니다.

```
SQL>
```

기본 SQL 작성하기 정리

❖ DDL 명령

3) 테이블 복사하기 (CTAS 라고도 합니다)

(2) 테이블의 구조(컬럼)만 가져오고 데이터 안 가져오기

```
SQL> create table emp4  
2 as  
3 select * from emp2  
4 where 1=2;
```

테이블이 생성되었습니다.

```
SQL>
```

```
SQL> select * from emp4;
```

선택된 레코드가 없습니다.

```
SQL>
```

주로 데이터는 필요 없이 테이블 구조만 가져올 때 많이 사용하는 방식이며 문법은 거의 동일한데 4번 라인에 보면 WHERE 절에 틀린 조건을 적어 주는 부분만 다르다.

기본 SQL 작성하기 정리

❖ DDL 명령

테스트용 테이블 생성

```
SQL> create table temp01(  
2   id number not null primary key,  
3   name varchar2(20),  
4   gender varchar(10),  
5   address varchar2(80));
```

테이블이 생성되었습니다.

```
SQL> desc temp01;
```

이름	널?	유형
ID	NOT NULL	NUMBER
NAME		VARCHAR2(20)
GENDER		VARCHAR2(10)
ADDRESS		VARCHAR2(80)

```
SQL>
```

기본 SQL 작성하기 정리

❖ DDL 명령

4) ALTER : 테이블 구조 변경

(1) 새로운 컬럼 추가하기

```
SQL> alter table temp01  
2 add (hp varchar2(13));
```

테이블이 변경되었습니다.

```
SQL> desc temp01
```

이름	널?	유형
ID	NOT NULL	NUMBER
NAME		VARCHAR2(20)
GENDER		VARCHAR2(10)
ADDRESS		VARCHAR2(80)
HP		VARCHAR2(13)

```
SQL>
```

기본 SQL 작성하기 정리

❖ DDL 명령

4) ALTER : 테이블 구조 변경

(1) 테스트용 데이터 입력

```
SQL> insert into temp01 values (1, 'park', '1', 'pusan', '010-1111-1111');  
1 개의 행이 만들어졌습니다.  
SQL> insert into temp01 values (2, 'kim', '2', 'seoul', '010-2222-1111');  
1 개의 행이 만들어졌습니다.  
SQL> insert into temp01 values (3, 'jo', '1', 'daegu', '010-2222-3333');  
1 개의 행이 만들어졌습니다.  
SQL>
```

```
SQL> select * from temp01;
```

ID	NAME	GENDER	ADDRESS	HP
1	park	1	pusan	010-1111-1111
2	kim	2	seoul	010-2222-1111
3	jo	1	daegu	010-2222-3333

```
SQL> ■
```

기본 SQL 작성하기 정리

❖ DDL 명령

4) ALTER : 테이블 구조 변경

(2) 컬럼 추가하면서 기본값 지정하기

```
SQL> alter table temp01  
2 add (location varchar2(10) default 'pusan');
```

테이블이 변경되었습니다.

```
SQL> select * from temp01;
```

ID	NAME	GENDER	ADDRESS	HP	LOCATION
1	park	1	pusan	010-1111-1111	pusan
2	kim	2	seoul	010-2222-1111	pusan
3	jo	1	daegu	010-2222-3333	pusan

```
SQL> █
```

기본 SQL 작성하기 정리

❖ DDL 명령

4) ALTER : 테이블 구조 변경

(3) 테이블의 컬럼 이름 변경하기

```
SQL> alter table temp01 rename column location to loc;  
(테이블이 변경되었습니다.)
```

```
SQL> select * from temp01;
```

ID	NAME	GENDER	ADDRESS	HP	LOC
1	park	1	pusan	010-1111-1111	pusan
2	kim	2	seoul	010-2222-1111	pusan
3	jo	1	daegu	010-2222-3333	pusan

```
SQL>
```

기본 SQL 작성하기 정리

❖ DDL 명령

4) ALTER : 테이블 구조 변경

(4) 컬럼의 데이터 크기 조정하기

```
SQL> desc temp01
```

이름	널?	유형
ID	NOT NULL	NUMBER
NAME		VARCHAR2(20)
GENDER		VARCHAR2(10)
ADDRESS		VARCHAR2(80)
HP		VARCHAR2(13)
LOC		VARCHAR2(10)

```
SQL> alter table temp01  
2 modify(loc varchar(20));
```

테이블이 변경되었습니다.

```
SQL> desc temp01
```

이름	널?	유형
ID	NOT NULL	NUMBER
NAME		VARCHAR2(20)
GENDER		VARCHAR2(10)
ADDRESS		VARCHAR2(80)
HP		VARCHAR2(13)
LOC		VARCHAR2(20)

```
SQL> _
```

기본 SQL 작성하기 정리

❖ DDL 명령

4) ALTER : 테이블 구조 변경

(5) 컬럼 삭제하기

```
SQL> alter table temp01 drop column loc;
```

테이블이 변경되었습니다.

```
SQL> desc temp01
```

이름	널?	유형
ID	NOT NULL	NUMBER
NAME		VARCHAR2(20)
GENDER		VARCHAR2(10)
ADDRESS		VARCHAR2(80)
HP		VARCHAR2(13)

```
SQL>
```

기본 SQL 작성하기 정리

❖ DDL 명령

5) TRUNCATE 명령

```
SQL> truncate table temp01;  
테이블이 잘렸습니다.  
SQL> select * from temp01;  
선택된 레코드가 없습니다.  
SQL> _
```

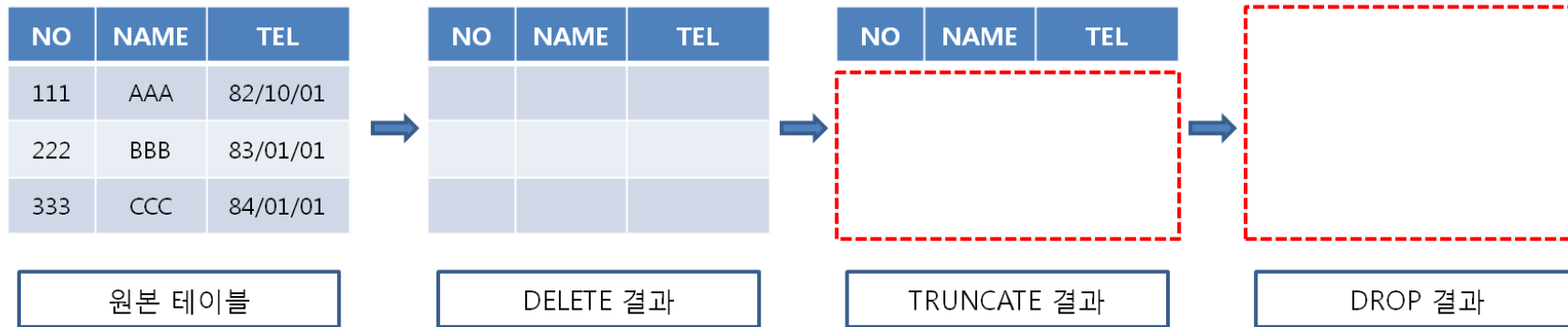
6) Drop 명령

```
SQL> drop table temp01;  
테이블이 삭제되었습니다.  
SQL> select * from temp01;  
select * from temp01  
          *  
1행에 오류:  
ORA-00942: 테이블 또는 뷰가 존재하지 않습니다  
  
SQL>
```


기본 SQL 작성하기 정리

❖ DDL 명령

7) DELETE , TRUNCATE , DROP 명령어의 차이점 비교



기본 SQL 작성하기 정리

□ 실습하기

- ✓ 'yangjung'이라는 사용자를 등록 후 비밀번호는 '1234'로 설정 후 'yangjung'이라는 사용자로 접속하여 다음 작업을 진행 하시오.

기본 SQL 작성하기 정리

❖ 실습하기

1. 아래와 같은 구조의 일반 테이블을 생성하세요.

new_emp

NO	NAME	HIREDATE	BONUS
NUMBER(5)	VARCHAR2(20)	DATE	NUMBER(6,2)

기본 SQL 작성하기 정리

❖ 실습하기

2. 위 1번 문제에서 생성한 new_emp 테이블에서 NO , NAME , HIREDATE 컬럼만 가져와서 아래 그림과 같이 new_emp2 테이블을 생성하는 쿼리를 쓰세요.

new_emp2

NO	NAME	HIREDATE
NUMBER(5)	VARCHAR2(20)	DATE

기본 SQL 작성하기 정리

❖ 실습하기

3. 위 2번 문제에서 생성한 new_emp2 테이블과 동일한 구조의 테이블을 new_emp3 이름으로 생성하되 테이블 구조만 가져오고 데이터는 가져오지 않도록 하는 쿼리를 쓰세요.

기본 SQL 작성하기 정리

❖ 실습하기

4. 위 2번 문제에서 생성한 new_emp2 테이블에 DATE 타입을 가진 BIRTHDAY 컬럼을 추가하는 쿼리를 쓰세요. 단 해당 컬럼이 추가될 때 기본값으로 현재날짜 (SYSDATE) 가 자동으로 입력되도록 하세요.

new_emp2

NO	NAME	HIREDATE	BIRTHDAY
NUMBER(5)	VARCHAR2(20)	DATE	2014/10/30
			2014/10/30
			2014/10/30

기본 SQL 작성하기 정리

❖ 실습하기

5. 위 4번 문제에서 생성한 new_emp2 테이블의 BIRTHDAY 컬럼 이름을 BIRTH 로 변경하는 쿼리를 쓰세요.

기본 SQL 작성하기 정리

❖ 실습하기

6. 위 4번 문제에서 생성한 new_emp2 테이블의 NO 컬럼의 길이를 NUMBER(7) 로 변경하는 쿼리를 쓰세요.

기본 SQL 작성하기 정리

❖ 실습하기

7. new_emp2 테이블의 컬럼 중에서 BIRTH 컬럼을 삭제하는 쿼리를 쓰세요.

기본 SQL 작성하기 정리

❖ 실습하기

8. new_emp2 테이블의 컬럼은 남겨 놓고 데이터만 지우는 쿼리를 쓰세요.

기본 SQL 작성하기 정리

❖ 실습하기

9. new_emp2 테이블을 완전히 삭제하는 쿼리를 쓰세요.

작성한 SQL문을 CTRL+A ,, CTRL+C 한 후 메모장에 CTRL+V 하여 0917_실습_이름.txt 제출