

## DataType (데이터타입)

- Data는 정보를 산출하기 위해 필요한 생명수
- DataType은 생명수를 담기 위한 그릇

## 1-1. Data Type 개요

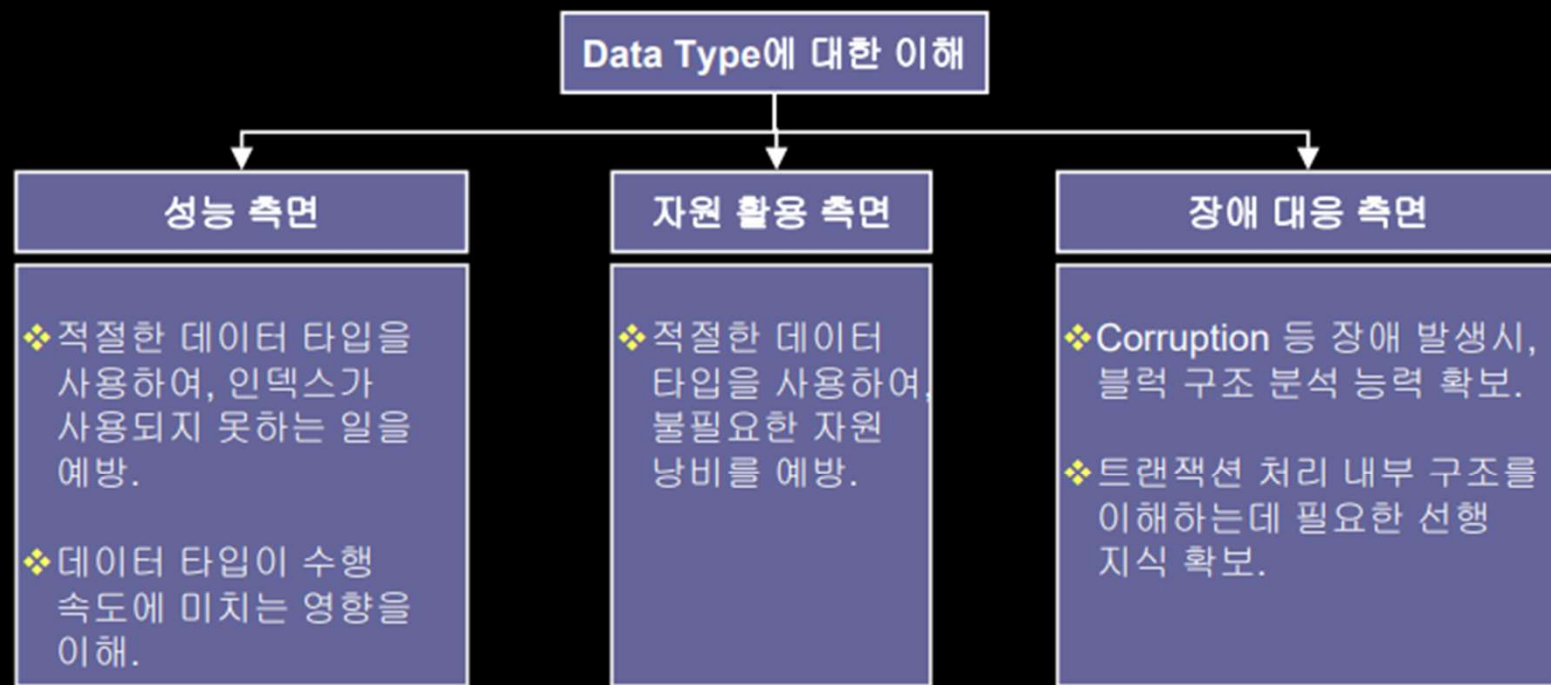
데이터 타입은, 테이블 컬럼을 정의하거나, 프로시저 / 함수의 인자에 사용되는 값이 저장되는 방식을 결정한다.



ORACLE

## 1-3. Data Type의 이해의 중요성

적절한 데이터 타입의 선정은 자원 활용 및 성능에 영향을 미치며, 내부 구현 방식에 대한 이해는, 장애 발생 또는 문제에 대한 해결 능력을 향상 시켜준다.

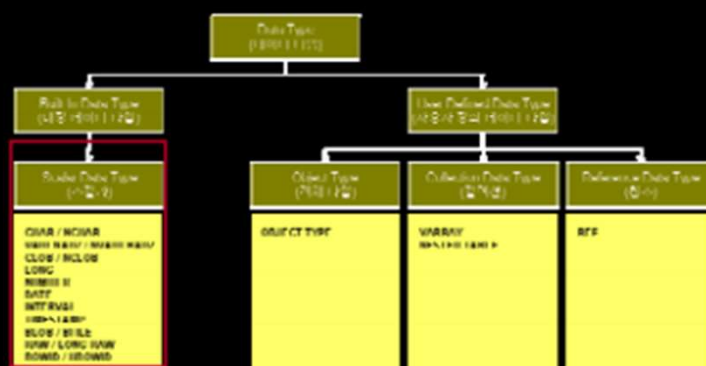


## 2. Scalar Data Type

2

### Scalar Data Type

- CHAR / NCHAR
- VARCHAR2 / NVARCHAR2
- CLOB / NCLOB
- LONG
- NUMBER
- DATE
- TIMESTAMP
- INTERVAL
- BLOB/BFILE
- RAW / LONG RAW
- ROWID / UROWID



ORACLE

## 2-11. ROWID, UROWID

ROWID 및 UROWID는 테이블의 ROW에 대한 고유 식별자로, 특정 ROW의 물리적 위치를 직접 저장하지는 않지만, 특정 행을 찾는 가장 빠른 메커니즘을 제공한다.

유형

내장 데이터 타입 ▶ 스칼라 타입 ▶ 데이터베이스 내부 처리용 데이터 타입

### ① 설명

- ❖ ROWID는 Database에 있는 각 행에 대한 고유 식별자임.
- ❖ ROWID는 명시적으로 컬럼 값으로서 저장되지 않음.
- ❖ ROWID는 행의 물리적 주소를 직접 부여하지는 않지만 행 위치를 지정하는 데 사용될 수 있음.
- ❖ ROWID를 사용하면 가장 빠르게 테이블의 행을 액세스할 수 있음.
- ❖ ROWID는 주어진 키 값의 집합을 가진 행을 지정하기 위해 인덱스에 저장됨.
- ❖ UROWID는 오라클 데이터베이스 이외 DB 테이블의 ROWID 및 IOT (Index Organized Table)의 ROWID를 처리할 수 있음.

### ② 특징

- ❖ ROWID는 3가지 타입이 존재함.
  - ❖ SHORT : RDBA와 ROW 정보로 구성
  - ❖ LONG : OBJECT ID와 RDBA, ROW 정보로 구성
  - ❖ UNIVERSAL
- ❖ UNIVERSAL ROWID (UROWID)에는 다시 3가지 타입이 존재함
  - ❖ PHYSICAL : 일반 테이블 및 클러스터, 파티션, 인덱스, 인덱스 파티션 및 서브파티션에 대한 주소
  - ❖ LOGICAL: Index Organized Table의 주소
  - ❖ REMOTE: 오라클 이외 데이터베이스에 대한 ROWID
- ❖ ROWID의 데이터 타입 코드는 69번이며 UROWID는 208번임.

### ③ Column Length / Default

- ❖ SHORT ROWID는 6바이트 16진수 문자열로 나타내짐.
- ❖ LONG ROWID는 10바이트 16진수 문자열로 나타내짐.
- ❖ UNIVERSAL ROWID는 최대 3950바이트까지를 차지할 수 있음.

### ④ 내부 저장 방식

#### ROWID LOG 타입 예시

```
Typ=69 Len=10: 0,0,2e,66,2,40,15,2,0,0
Object : 0x000e66 = 11878 (AAAC5m)
RDBA : 0x02401502 -> Relative File 9 (AAJ), Block 5378 (AAABUC)
Slot : 0x0000 = 0 (AAA)
```

ORACLE

## 4. 데이터 타입의 선정 지침

---

4

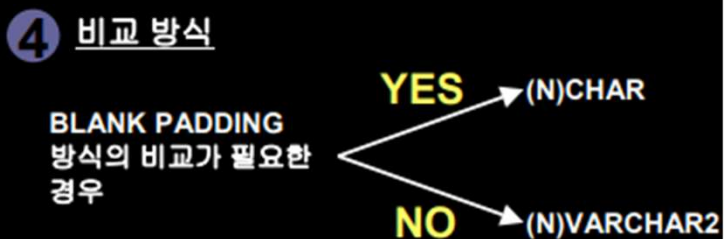
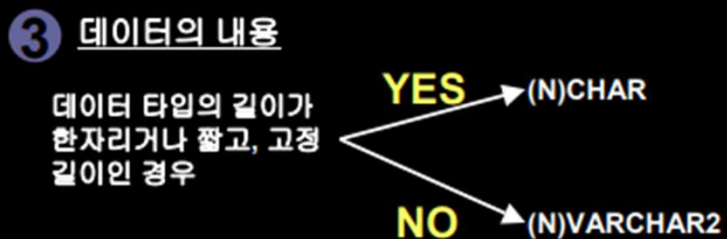
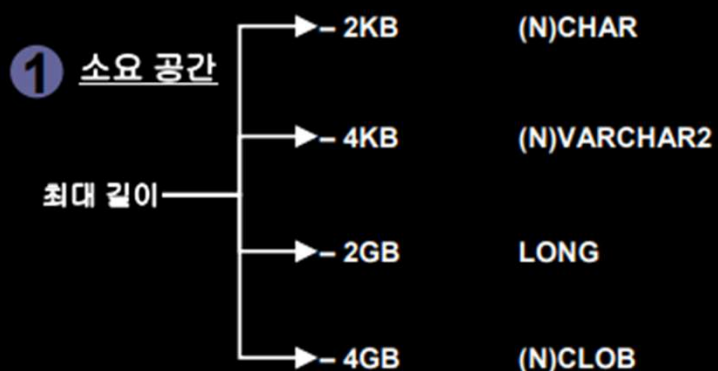
### 데이터 타입 선정 지침

- 문자열 관련 일반 지침
- 숫자/날짜 관련 일반 지침
- **LONG/LONG RAW** 제약 사항
- **BLOB/CLOB** 제약 사항

ORACLE®

## 4-1. 문자열 관련 일반 지침

문자 데이터와 관련된 데이터 타입 선정시, (1) 소요 공간, (2) 유니코드, (3) 데이터 내용, (4) 비교 방식 등을 고려해야 한다.



ORACLE



## 4-2. 숫자, 날짜 관련 일반 지침

오라클 9i 까지 모든 숫자는 내부적으로 NUMBER 타입에 저장된다. 날짜를 나타내는데는, DATE, TIMESTAMP를 사용할 수 있으며 경우에 따라 문자열 데이터 타입을 사용하는 경우도 있다.

### 숫자 관련 일반 지침

1. 반올림과 관련된 사항을 고려.
  - 예를 들어 유효 자릿수가 소수점 두자리 숫자에 대한 연산을 할 경우, 소수점 두자리 숫자끼리 연산을 해서 결과값 역시 소수점 두자리 값으로 계산되도록 할 것.
  - 나눗셈 연산시, (예 1/3) 유효 자릿수를 지정하지 않을 경우, 소수점 자릿수를 많이 소요하게 되므로, 자릿수 지정에 유의.  
Row-Chaining 유발.

### 날짜 관련 일반 지침

1. 초 이하 단위에 대한 저장이 필요할 경우, TIMESTAMP를 고려.
2. 날짜 관련 연산이 빈번할 경우, DATE 또는 TIMESTAMP를 고려할 것.
3. 년월일에 대한 LIKE 나 = 과 같은 조건을 자주 사용할 경우, CHAR, VARCHAR2 고려. 단 유효한 날짜인지 여부는 사용자 애플리케이션에서 확인 하여야 함.

참고 : 10g에서는 IEEE 754 형태의 BINARY\_FLOAT과, BINARY\_DOUBLE이 추가 되었음.

ORACLE



## 2-6. DATE

DATE 데이터 타입은 날짜와 시간을 고정 길이로 표현하는데 사용된다.

유형

내장 데이터 타입 ▶ 스칼라 타입 ▶ 날짜 데이터 타입

### ① 설명

- ❖ 세기 / 년 / 월 / 일 / 시 / 분 / 초 데이터를 저장할 수 있음.
- ❖ 초 이하 단위 (밀리초 등)과, Timezone과 관련된 정보를 저장하지 않음.

### ② 특징

- ❖ 전체 7바이트 가운데, 첫 두 바이트는 세기와 연도를 나타내고, 다음 두 바이트는 월, 일을 나타내며, 나머지 세 바이트는 시, 분, 초를 나타냄.
- ❖ DATE의 데이터 타입 코드는 12번임.

### ③ Column Length / Default

- ❖ 기원전 4712년 1월 1일부터, 기원후 9999년 12월 31일까지 표현 가능.
- ❖ 항상 7바이트의 고정 길이를 사용함.

### ④ 내부 저장 방식

Typ=12 Len=17: 120, 101, 10, 10, 14, 12, 58  
Century : 120 - 100 = 20  
Year : 101 - 100 = 01  
Month : 10  
Day : 10  
Hours : 14 - 1 = 13  
Minutes : 12 - 1 = 11  
Seconds : 58 - 1 = 57

ORACLE

-- 오라클 DATE 객체 정보를 바탕으로 쿼리 성능을 비교하는 경우 (

-- 쿼리 1 : to\_char() 반환 문자열 = '1981' 문자열 비교

```
SELECT empno, ename
FROM emp
WHERE to_char(hiredate, 'YYYY') = '1981'      -- 동일한 DataType ~ String
AND empno > 7700
;
```

-- 쿼리 2 : extract() 반환 정수 = 1981 정수 비교

```
SELECT empno, ename
FROM EMP
WHERE EXTRACT (year FROM hiredate) = 1981      -- 동일한 DataType ~ integer
AND empno > 7700
;
```

```

class ORACLE {
    class Schema {
        class Table;
        class Index;
        class View;
        class Sequence;

        class DATE { // 오라클이 설계한 DATE 객체
            int type;
            int len;
            int Year;
            int Month;
            int Day;
            int Hours;
            int Minutes;
            int Seconds;

            public DATE(void) { // 오라클 DATE 객체 생성자 예시 (SYSDATE 등의 날짜 객체)
                return get_datetime_from_os() // OS로부터 날짜일시 획득
            }

            public str to_char(date dt, string fmt) {
                return str_date // 포맷(fmt)에 따른 날짜일시를 문자열로 반환
            }

            Public int extract(string fmt) {
                return int_date // 포맷(룻)에 따른 날짜일일시의 구성요소를 반환
            }
        }
    }
}

```