

|  |
| --- |
| 데이터표준정의서  Version 1.0  2018.04.06 |

|  |
| --- |
|  |

제·개정 이력

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 제·개정일 | Version | 제·개정 내용1) | 개정위치 | 작성자 | 검토자 | 승인자 |
| 2018.04.06 | 1.0 | 최초 작성 | 전체 | 김한수 | 김은범 | 이재근 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

1) 제 ·개정 내용 : 최초작성, 수정, 추가, 삭제로 구분하여 작성함. 수정, 추가, 삭제의 경우 괄호 안에 해당 내용 기술함.

목 차

[1. 문서 개요 6](#_Toc509503822)

[1.1 목적 6](#_Toc509503823)

[1.2 대상 6](#_Toc509503824)

[2. 데이터 표준화 개요 6](#_Toc509503825)

[2.1 데이터 표준화 정의 6](#_Toc509503826)

[2.2 데이터 표준화 목적 6](#_Toc509503827)

[2.3 데이터 표준화 대상 7](#_Toc509503828)

[2.4 데이터 표준 정의를 위한 용어 정의 8](#_Toc509503829)

[3. 단어 표준 9](#_Toc509503830)

[3.1 개요 9](#_Toc509503831)

[3.2 한글 단어 표준 정의 원칙 9](#_Toc509503832)

[3.2.1 한글 단어 표준 기본 원칙 9](#_Toc509503833)

[3.2.2 영문 및 외래어의 사용 원칙 9](#_Toc509503834)

[3.2.3 합성어 사용의 원칙 10](#_Toc509503835)

[3.2.4 합성어 사용의 예외 원칙 10](#_Toc509503836)

[3.2.5 접두(미)사의 사용 원칙 11](#_Toc509503837)

[3.3 한글단어에 대한 영문약어 표준 정의 원칙 12](#_Toc509503838)

[3.4 단어사전 관리 원칙 13](#_Toc509503839)

[4. 용어 표준 13](#_Toc509503840)

[4.1 개요 13](#_Toc509503841)

[4.2 한글 용어(논리명) 표준정의 원칙 14](#_Toc509503842)

[4.2.1 한글용어(논리명) 기본원칙 14](#_Toc509503843)

[4.2.2 주제영역 명명규칙 14](#_Toc509503844)

[4.2.3 엔티티명 명명규칙 15](#_Toc509503845)

[4.2.4 속성명 명명규칙 16](#_Toc509503846)

[4.2.5 유형별 용어 사용의 원칙 16](#_Toc509503847)

[4.2.5.1 숫자가 포함된 경우 16](#_Toc509503848)

[4.2.5.2 금액 관련 용어 17](#_Toc509503849)

[4.2.5.3 금액 관련 용어의 어순 17](#_Toc509503850)

[4.2.5.4 날짜 관련 용어 18](#_Toc509503851)

[4.2.5.5 총 vs 합계 18](#_Toc509503852)

[4.2.5.6 현재 vs 전체 18](#_Toc509503853)

[4.3 영문 용어(물리명) 표준정의 원칙 18](#_Toc509503854)

[4.3.1 물리 데이터 오브젝트 명명규칙 18](#_Toc509503855)

[4.3.2 테이블명 명명규칙 18](#_Toc509503856)

[4.3.3 컬럼명 명명규칙 19](#_Toc509503857)

[4.3.4 기타 물리 데이터 오브젝트 명명규칙 19](#_Toc509503858)

[5. 도메인 개요 20](#_Toc509503859)

[5.1 도메인 정의 20](#_Toc509503860)

[5.2 도메인의 사용 목적 21](#_Toc509503861)

[5.2.1 용어 표준화 21](#_Toc509503862)

[5.2.2 인포타입의 일괄 부여/변경 21](#_Toc509503863)

[5.2.3 속성값 유효성 제한 21](#_Toc509503864)

[5.3 도메인 분류 기준 21](#_Toc509503865)

[5.4 도메인 구성 요소 21](#_Toc509503866)

[5.4.1 도메인그룹 21](#_Toc509503867)

[5.4.2 도메인 유형 22](#_Toc509503868)

[5.4.3 도메인명 22](#_Toc509503869)

[5.4.4 인포타입(Information Type) 22](#_Toc509503870)

[6. 도메인 분류 23](#_Toc509503871)

[7. 제한형 도메인 표준 24](#_Toc509503872)

[7.1 명 24](#_Toc509503873)

[7.1.1 물리속성명(프로그램명 VS 프로그램영문명 VS 프로그램한글명 등) 24](#_Toc509503874)

[7.1.2 ID 24](#_Toc509503875)

[7.1.3 주소 24](#_Toc509503876)

[7.2 내용 25](#_Toc509503877)

[7.2.1 전문내용 25](#_Toc509503878)

[7.3 금액 25](#_Toc509503879)

[7.3.1 금액 VS 금 25](#_Toc509503880)

[7.3.2 가격 VS 가 25](#_Toc509503881)

[7.3.3 세금 VS 세 25](#_Toc509503882)

[7.3.4 이익 VS 손실 26](#_Toc509503883)

[7.3.5 원화금액 VS 외화금액 26](#_Toc509503884)

[7.4 수량 26](#_Toc509503885)

[7.4.1 건수 VS 수 26](#_Toc509503886)

[7.4.2 시간 VS 시각 26](#_Toc509503887)

[7.5 날짜 26](#_Toc509503888)

[7.5.1 일 VS 일자 26](#_Toc509503889)

[7.6 율 27](#_Toc509503890)

[7.6.1 금리 VS 이율 27](#_Toc509503891)

[7.6.2 율 VS률 27](#_Toc509503892)

[8. 번호도메인표준 27](#_Toc509503893)

[8.1.1 번호 VS 일련번호 28](#_Toc509503894)

[8.1.2 실명번호 28](#_Toc509503895)

[9. 코드 도메인 표준 28](#_Toc509503896)

[9.1.1 코드 유형명 28](#_Toc509503897)

[9.1.2 코드 도메인 명명규칙 29](#_Toc509503898)

[9.1.3 여부VS 유무 30](#_Toc509503899)

[9.1.4 구분 VS 유형 VS 분류 VS 종류 30](#_Toc509503900)

[9.1.5 외부기관 정의 코드 31](#_Toc509503901)

[10. 인포타입(Information Type) 31](#_Toc509503902)

[10.1 인포타입 명명 규칙 31](#_Toc509503903)

[10.2 인포타입 부여 규칙 32](#_Toc509503904)

[10.3 인포타입 데이터유형 32](#_Toc509503905)

[10.4 표준 인포타입 32](#_Toc509503906)

[11. 코드 표준화 34](#_Toc509503907)

[11.1 코드 표준화 개요 34](#_Toc509503908)

[11.1.1 코드 표준화 목적 34](#_Toc509503909)

[11.1.2 코드 표준화 원칙 34](#_Toc509503910)

[11.2 코드의 종류 35](#_Toc509503911)

[11.2.1 코드 종류 35](#_Toc509503912)

[11.2.2 코드종류별 관리방법 35](#_Toc509503913)

[11.2.3 목록성 코드 35](#_Toc509503914)

[11.2.4 코드 부여방식 36](#_Toc509503915)

[11.3 코드 명명규칙 37](#_Toc509503916)

[11.3.1 코드관련 용어 정의 37](#_Toc509503917)

[11.3.2 코드ID 명명규칙 37](#_Toc509503918)

[12. 코드 부여 표준화 37](#_Toc509503919)

[12.1 코드 부여원칙 37](#_Toc509503920)

[12.1.1 코드 부여원칙 38](#_Toc509503921)

[12.1.2 코드 부여원칙 예외 38](#_Toc509503922)

[12.2 코드표준 세부작업 방안 39](#_Toc509503923)

[12.2.1 코드 분류 39](#_Toc509503924)

[12.2.2 중복 코드의 제거 39](#_Toc509503925)

[12.2.3 동일 의미 코드의 통합 40](#_Toc509503926)

[12.2.4 코드 표준화 40](#_Toc509503927)

# 문서 개요

## 목적

본 문서는 비티씨코리아닷컴㈜의 단어 표준 및 용어 표준을 정의하고 관리하기 위한 목적 및 용도로 작성한다.

## 대상

본 문서는 다음과 같은 사용자를 대상으로 작성되었다.

* 데이터베이스 설계자
* 데이터베이스 운영자 및 사용자
* 메타데이터시스템 운영자 및 사용자

# 데이터 표준화 개요

## 데이터 표준화 정의

데이터표준화는 여러 시스템에 산재해 있는 데이터에 대한 한글(논리)명/영문(물리)명 명명규칙, 데이터타입, 데이터길이 등에 대한 기준을 마련하여 데이터의 품질을 향상 시키고자 하는 활동을 말한다.

## 데이터 표준화 목적

* **용어 사용의 혼란 방지**
* 표준화된 명칭을 부여함으로써 데이터의 중복 정의 방지와 불일치를 제거하여 의사소통을 원활하게 한다.
* **전사 데이터 모델의 일관성 유지**
* 전사적으로 표준화된 용어를 사용함으로써 모델을 구성하는 오브젝트명칭의 일관성을 유지할 수 있다.
* **데이터의 정확성 및 품질수준 확보**
* 표준화를 통해 데이터의 구조와 흐름 등 정합성 검증이 용이하여 데이터의 품질수준을 확보할 수 있다.
* **공통의 표준화된 데이터 이해**
* 같은 의미를 나타내는 단어는 하나의 단어로만 표현 하고 단어 간의 의미의 경계를 명확히 하여 공통의 이해를 기반으로 상호간 의사소통을 명확히 함
* 시스템에 대한 이해(Readability)를 향상시켜 시스템 상호 운용성을 향상
* **개발 생산성 향상**
  + 공통의 용어 정의를 통해 재사용 기회를 높이고 항목명 작명 시 단어 선택에 대한 시간을 단축시킴으로써 개발 생산성을 향상
* **데이터 관리의 일관성과 효율성을 높임**

## 데이터 표준화 대상

표준화 대상 오브젝트는 원천 시스템으로부터 Data가 작성되어 사용자의 Data활용 때까지 일련의 전 과정이 포함되며, 논리 설계로써 표준화 원칙과 명명규칙뿐만 아니라, 물리 설계에 해당하는 표준화 원칙과 물리적인 오브젝트의 명명규칙도 포함된다.

* **논리 데이터 모델 오브젝트**
* 주제영역: 기업모델의 데이터 개념을 보다 관리 하기 쉽도록 특정 주제를 가지고 분류한 영역으로서 그 하위에 엔티티 집합을 가지고 있다. 예를 들어 ‘고객’이라는 주제영역은 고객에 관련된 엔티티들을 집합시켜 놓은 것이다.
* 엔티티: 데이터의 기본 단위로서 업무에서 관리하고자 하는 대상에 대한 정보를 가지고 있는 단위이다. 예를 들어 ‘회원기본’이라는 엔티티는 회원의 시스템 이용에 필요한 기본 정보를 가지고 있다.
* 속성: 속성은 엔티티들이 가지는 특징을 분류하는 가장 작은 단위의 데이터 구성 요소이다. 예를 들어 ‘회원기본’이라는 엔티티는 성명, 이메일 주소, 휴대전화번호, 성별 등과 같은 특징을 갖게 되는데 이러한 특징이 속성에 해당 된다.
* **물리 데이터 모델 오브젝트**
  + 테이블: 논리 설계 단계의 엔티티에 대응하는 오브젝트
  + 컬럼: 논리 설계 단계의 속성에 대응하는 오브젝트
  + 인덱스: 테이블에서 특정조건의 데이터를 효율적으로 검색하기 위한 색인데이터로 대표적인 인덱스 대상으로는 기본키(Primary Key), 유일키(Unique Key)등이 있다.
  + 뷰: 테이블에 대한 재 정의로 물리적으로 테이블의 특정 Column, 특정 Row를 뷰로 정의하여 특정 사용자만 접근이 가능하도록 할 수 있다.
  + 데이터베이스: 논리적으로 연관된 하나 이상의 테이블의 모음을 저장하는 오브젝트
  + 테이블스페이스: 테이블을 저장하는 물리적 공간을 지칭하는 오브젝트

## 데이터 표준 정의를 위한 용어 정의

[표 2‑1] 데이터 표준 정의를 위한 용어

|  |  |  |
| --- | --- | --- |
| 용어 | | 내용 |
| 단어 | | 테이블과 컬럼의 한글명 및 영문명을 구성하는 최소한의 원소를 가리키며 용어는 반드시 이러한 단어의 조합으로 이루어 짐 |
|  | 단일어 | 다른 단어와의 조합이 아닌 순수한 단어 |
| 합성어 | 일반 단어나 접두[미]사의 조합으로 이루어진 단어 |
| 유사어 | 표준 한글 단어와 함께 사용이 허락된 단어. 즉 사용 할 수는 있으나 제시된 표준단어를 사용하도록 권장 함 |
| 금칙어 | 사용이 허락되지 않는 단어 |
| 분류어 | 개체의 유형을 명확하게 나타내기 위한 단어로서 표준 용어의 요소이며 도메인의 근간이 됨. |
| 용어 | | 단어의 조합으로 이루어져 테이블 및 컬럼의 명칭으로 사용되는 용어 |

* 하나의 단어는 용어이거나 도메인이 될 수 있다.
* 하나의 도메인은 단어이거나 용어 이어야 한다.
* 합성어는 주로 아래와 같은 목적으로 생성된다.
* 업무적으로 중요한 의미를 가지며 전사적으로 합성된 단어를 주로 사용하는 것
* 물리 명이 길어지는 것을 방지하기 위해 단어들을 축약 표현하기 위한 목적으로 결합시켜 생성한 것 (예 : 신불자, 등록자)
* 엔티티 분류어 : 엔티티 유형을 명확하게 하기 위해 엔티티명 뒤에 붙여 그 의미를 명확하게 한다. (예: 기본, 상세, 이력, 내역 등)
* 속성 분류어 : 속성 값의 유형을 명확하기 위해 속성명 뒤에 붙여 그 의미를 명확하게 한다. (예: 코드, 번호, 금액, ID, 명, 일자, 일시, 주소, 내용 등)
* 분류어는 아래 사항을 참고하여 단어 중에서 선정한다.
  + 개체의 유형을 명확하게 나타내기 위한 단어로 선정한다.
  + 도메인은 [수식어]+분류어로 구성되어지나 '~금(금액)', '~세(세금)', '~가(가격)', '~수(건수/개수)'와 같은 단축 단어형 분류어를 별도의 단어로 등록하는 경우에는 ①동음이의어 생성 가능성이 높고, ②도메인의 이중관리로 인해서 발생하는 도메인의 불일치로 일관성이 결여될 수 있으며 ③물리명 생성과정에서도 일관성이 결여될 수 있으므로 번호도메인과 코드도메인을 제외한 그룹(제한형)도메인은 수식어를 제외하고 합성어를 포함한 단어로 등록하고 분류어로 생성한다.

예① : 부가가치세--> 부가가치세\_금액

예② : 종가--> 종가\_가격(X)/종가

# 단어 표준

## 개요

단어란 테이블과 컬럼의 한글명 및 영문명을 이루는 원소를 가리키며 테이블명, 컬럼명으로 사용되는 용어는 반드시 이러한 단어의 조합으로 이루어 진다.

## 한글 단어 표준 정의 원칙

### 한글 단어 표준 기본 원칙

* 명사형으로 정의하며 동사, 접속사, 복수 표시 또는 소유격형태의 단어는 사용하지 않는다.

(예) 회원 이전 사용만가능 금액, 마지막 초기화시간, 과, 와

* 띄어쓰기를 허용하지 않는다.

(예) 이체 은행코드 (X) -> 이체은행코드

* 특수문자(/, \_, -, +, (, ))는 사용하지 않는다.

(예) 판/구매등록구분 -> 거래등록구분, 오입금수량-당일입금수량(X),

받는BTC->KRW금액(X)

* 업무에서 사용하는 관용어를 우선 사용한다. (코인, 충전(입금) 등 고려)
* 동음이의어(한글 단어명 기준)는 되도록이면 사용하지 않으나 대체 단어가 없는 경우 사용할 수 있다.

(예) 수입(Import), 수입(Earning)

* 동음이의어는 실체명/속성명 등을 대상으로 표준화 작업을 할 때 사용자의 직접(수동)적인 선택이 필요하다.
* 동일한 의미의 단어를 한글과 영문으로 중복해서 정의하지 않는다

(예)COIN, 코인(X), BTC, 비트코인(X)

한글약어는 되도록 피하며, 예외적으로 범용으로 쓰이는 약어나, 공식적으로 사용이 승인된 약어는 표준에 의거하여 길이 제한을 맞추기 의해 특별히 사용할 수 있다.

(예) 주민번호(X) -> 주민등록번호, 사번(X)-> 사원번호

### 영문 및 외래어의 사용 원칙

* 한글단어에는 가급적 영문을 사용하지 않으며 한글화를 원칙으로 하나 대체할 한글이 없거나 범용적으로 사용되는 영문은 사용할 수 있다.

(예) FAX(X) -> 팩스, EMAIL(X) -> 이메일, SQL, CRON, URL

* 영문을 한글화 할 경우 의미를 파악할 수 있도록 한글화 하는 것을 원칙으로 하나 발음대로 한글화 한 명칭이 더 관용적일 경우 이를 사용할 수 있다. 의미로 한글화 한 경우와 발음대로 한글화 한 경우 둘 다 자주 사용될 경우 의미로 한글화 한 단어를 표준으로 한다.

(예) PROCESS(X) -> 프로세스, THREAD(X) -> 스레드

* 영문단어를 사용하는 경우 반드시 1Byte 대문자로 정의한다.

(예) b/s(X)-> BS

* 외래어의 경우 한글을 우선으로 하나 한글명 보다 더 친숙하게 사용되는 외래어의 경우에는 그 외래어를 한글용어로 사용한다.

### 합성어 사용의 원칙

* 단독으로 사용 가능한 단일어의 조합에 의해 생성된 단어를 합성어로 등록한다.
* 합성어를 구성하는 단어는 반드시 단어 사전에 등록되어 있어야 한다.

(예) ‘계좌번호’합성어 등록 시 ‘계좌’, ‘번호’가 각각 단어로 등록되어 있어야 함

* 단어를 관리하는 차원에서는 단일어만 관리하여도 문제가 없으나 아래의 경우에는 단일어+단일어의 형태를 가지는 합성어로 등록하여 관리 할 수 있다.
  + 합성어에 대한 영문을 등록하는 방법은 2가지가 있다. 즉, 단일어에 의해 합성된 영문약어를 그대로 사용하는 방법(ACCT\_NO)과 합성어로 등록할 시 영문약어를 축약하여(ANO) 등록하는 방법이 있다. 당사에서는 합성어 등록 시 축약된 영문약어로 등록함을 원칙으로 한다.
  + 단, 관용적으로 합성된 단어를 자주 사용하는 경우 등 일관된 물리 명 사용을 위해서 각 단일어의 영문약어를 ‘\_’를 사용하여 조합하여 등록해서 사용할 수 있다.

(예) 계좌번호(ACCT\_NO),

* 단어의 조합으로 생성된 컬럼영문명의 길이가 30자를 초과하는 경우 영문약어명을 30자 이내로 축약한 별도의 영문단축명을 반드시 부여하여 사용한다.

### 합성어 사용의 예외 원칙

* 합성어는 이미 등록된 단일어의 조합에 의해 생성된 단어를 지칭하는 것이나 영문변환을 고려하여 아래의 경우 단일어로 취급하여 해당 단어에 맞는 영문약어를 정의한다.
* 접두(미)사와 결합된 단어는 단일어로 등록한다.

(예) 입금자, 송금인, 세금액, 송금율 등

* [~액]형태로만 사용되는 경우는 단일어로 등록하고 [~금액], [~액]형태 모두 사용하는 경우는 [~금액]형태로 사용한다.

(예) 잔액(X) -> 잔고+금액

* 회사 고유명 또는 서비스명을 정의하기 위해 사용한 단어는 단일어로 등록한다.

(예) 빗썸(Bithumb) : BTHB

* 단일어의 조합에 의해 생성된 영문명의 의미가 더 모호하거나 달라질 경우 단일어로 등록하고 해당 단일어에 적합한 영문약어를 정의하여 등록한다.

(예) 세금(tax)+계산서(accounts sheet): TAX\_ACSH (X), 세금계산서(invoice): INVC, 휴대(Portable)+전화(Telephone): PRTB\_TEL(X), 휴대전화(Mobile Phone): MBPH

* 합성어를 이루는 단일어가 당사에서 별도로 사용될 가능성이 없는 경우 개별 단일어를 등록하지 않고 합성된 단어를 단일어로 등록한다.

(예) ‘잡수익’의 ‘잡’은 당사에서 별도의 단어로 사용될 가능성이 없으므로 ‘잡수익’을 단일어로 등록한다.

### 접두(미)사의 사용 원칙

* 접두(미)사 자체는 반드시 단일어로 등록한다.

(예) 대, 중, 소, 전, 후

* 일반적으로 단일어와 결합 시 영문이 변경되는 접두사는 접두사와 결합된 단어를 단일어로 등록하고, 접두사와 결합된 단어의 영문이 별도로 존재하지 않는 경우 등록하지 않는다.

(예)정상(normal: NRML), 비정상(abnormal)의 경우 ‘비정상’이 비+정상: N\_NRML으로 사용되지 않도록 ‘비정상(abnormal: ABNRML)’을 단일어로 등록한다.

* 접미사는 일반단어와 결합 시 대부분 별도의 영문이 존재하므로 일반단어와 결합된 단어를 등록하여 사용한다.

[표 2] 접두(미)사 사용 원칙

|  |  |  |
| --- | --- | --- |
| 구분 | | 사용 원칙 |
| 접두사 | 단일어와 결합 시 일반적으로 별도의 영문명이 존재하는 접두사  (예) 부(不), 재(再), 미(未), 비(非), 부(副), 선(先) | 접두사와 결합된 단어를 단일어로 등록  (예)’비과세’ |
| 단일어와 결합 시 일반적으로 별도의 영문이 존재하지 않는 접두사  (예) 신(新), 구(舊), 전(前), 후(後), 모(母), 자(子), 유(有), 무(無), 별(別), 총(總), 원(原), 대(大), 중(中), 소(小), 주(主), 부(副) | 접두사와 결합된 단어를 등록하지 않음. 단, 관용적으로 사용하는 단어는 단일어로 등록함.  (예) ‘구계좌번호’(X)  -> ‘구’, ‘계좌번호’ |
| 접미사 | 2단일어와 결합 시 별도의 영문명이 존재하는 접미사  (예) 사(社), 처(處), 자(者), 인(人), 증(證), 지(地), 료(料), 서(書), 장(長), 금(金), 비(賁), 권(權), 성(性), 소(所), 형(型), 표(表), 식(式), 화(化), 도(度) | 접미사와 결합된 단어를 단일어로 등록함.  (예) ‘지급처’, ‘개설인’, ‘교환소’ |
| 단일어와 결합 시 일반적으로 별도의 영문명이 존재하지 않아 단일어와 분리해서 사용하는 접미사  (예) 용(用) | 접미사와 결합된 단어를 단일어로 등록하지 않음.  (예) ‘프로세스용’(X)  -> ‘프로세스^용’ |

## 한글단어에 대한 영문약어 표준 정의 원칙

* 영문약어는 한글명에 대한 영문명을 어두 이외의 모음을 제거하여 각 음절의 대표자음으로 가급적 4자 이내로 정의한다.
* 한글명에 대한 영문명이 4자 이내인 경우는 영문명을 영문약어로 사용한다.

단, 영문약어 명명의 일관성을 고려하여 예외를 둘 수 있다.

예① : 은행(BANK)-->BANK, 코드(CODE)-->CODE

예② : 연-->YY(YEAR), 월-->MM, 일-->DD(DAY), 시-->HH(HOUR),

분-->MI, 초-->SS, 일자-->DT, 일시-->DTM, 시각 -->TIME

* 반드시 대문자로 정의하고 물리모델 반영시 해당 DBMS특성에 따라 변환은 허용한다.
* 영문약어명 사용시 한글 발음식의 약어는 사용하지 않으나 순수 한글이거나 대체할 영문이 없는 경우 사용할 수 있다.

(예) 주소: JUSO (X) -> ADDR, 구분: GUBUN (X) -> KIND 시: SI, 군: GUN, 구: GU -> 허용

* 두 개의 자음이 연속할 경우, 가급적 한 개의 자음은 생략하는 형태로 표현한다.
* 영문약어는 유일성을 유지하도록 관리한다. 단, 동일한 영문명을 쓰는 것이 타당하다고 판단된 유사어의 경우에만 동일한 영문약어를 사용할 수 있다. 동일한 영문 Full Name에 대해서는 동일한 약어를 가지도록 권장한다.

(예) 가(Price: PRC) - 가격(Price: PRC), 율(Rate: RT) – 률(Rate: RT), 금(Amount: AMT) – 금액(Amount: AMT)

* SQL(Structured Query Language)와 같이 범용적으로 사용되는 두문자어(Acronyms)가 있는 경우 되도록 두문자어를 사용하며 자릿수가 지나치게 긴 경우 새로운 영문약어를 작성한다.

(예) SQL, VAN, OTP 등

## 단어사전 관리 원칙

* 단어 등록 시 기존에 사용한 단어의 의미를 토대로 하여 사전과 단어 사용자를 통해 단어의 의미를 명확하게 관리함
* 단어는 사용하고자 하는 목적에 맞게 표준단어(표준 논리명)와 표준단어 영문명(표준 물리명)을 모두 관리해야 함

(예: 사원 – EMP(EMPLOYEE), 계정 – ACCT(ACCOUNT))

* 표준단어 중 속성 분류어 혹은 엔티티 분류어를 통합하여 관리함
* 표준단어가 아니더라도 해당 단어를 자주 사용하는 경우는 비표준 단어로써 표준단어와 이음동의어 그룹으로 묶어 표준을 제시 받을 수 있도록 관리함
* 동음이의어인 경우 금융권에서 사용하는 단어를 중심으로 정의함

(예: 전문 - [電文] 전보의 내용이 되는 글)

* 동음이의어라도 두 개 이상의 의미가 모두 사용되는 경우는 등록 가능함.

(예: 수입 - [輸入] 다른 나라로부터 물품을 사들임. 수입 - [收入] 일이나 사업(경제활동) 등을 통하여 이익을 거두어 들임 또는 그 이익)

* 다른 명사와 결합되어 사용되는 접두사, 접미사는 비표준 단어로서 관리함

(예: 현~ - 현재, ~액 - 금액, ~가 – 가격, ~비 - 비용, ~비 - 비율)

* 접두사, 접미사와 결합된 단어들은 되도록이면 별도로 등록하지 않고 표준단어로 순화하여 사용하나 표준단어로 순화가 어려운 경우는 그 자체를 표준단어로서 등록함 (🡪 별도의 합성어를 등록하여 관리하도록 권고)

(예: 기준가 -> 기준가격 (기준+가격), 매출액 -> 매출금액 (매출+금액))

(예: 사업주, 교환소, 미사용)

* 경제 용어, 세무 용어, 법률 용어 등의 전문 용어이거나 낱개 단어로 분리하기 어렵고 분리 시 의미 전달이 모호한 경우는 해당 용어를 표준단어로 정의함

(예: 해외사업환산대, 감가상각부인액 등)

* 축약된 단어는 의미가 모호하므로 되도록이면 사용하지 않음

(예: 사번 -> 사원+번호, 주민번호 -> 주민+등록+번호)

* 일반적으로 축약하여 자주 사용되거나 용어의 길이가 길어져 영문명이 지나치게 길어질 수 있는 것들은 축약된 단어를 표준단어로 정의함

(예: 차변대변 -> 차대변, 상수도하수도-> 상하수도)

# 용어 표준

## 개요

한글용어는 물리적으로 구현되는 영문 테이블/컬럼명을 생성하기 위한 기준이 되는 용어로 주제영역, 엔티티(Entity), 속성(Attribute)명 등의 논리 데이터 항목에 대한 표준용어사용 지침을 정의하며, 물리영문명 표준은 논리 데이터 모델의 물리 데이터모델 전환시 필요한 테이블, 컬럼, 인덱스, 뷰 등의 데이터베이스 오브젝트에 대한 표준을 제공한다.

## 한글 용어(논리명) 표준정의 원칙

### 한글용어(논리명) 기본원칙

[표준단어 + … + 표준단어 + 분류어 (표준단어) ] 형태의 명명규칙을 준수한다.

* 누구나 이해하기 쉽도록 간결하고 명확한 표현을 사용한다.
* 반드시 단어사전에 등록된 단어를 사용하며 단어사전에 등록되어 있지 않은 경우, 단어

등록이 선행되어야 한다.

* 한글과 영문 약어를 이어서 사용하는 경우에 띄어쓰기를 하지 않고 Under Bar(\_) 및 특수문자를 사용하지 않는다. 즉, 단어와 단어의 조합시 띄어쓰기를 허용하지 않는다.

(예) 거래 일자(X), 거래\_일자(X), 거래일자 OTP 인증번호(X), OTP\_인증번호(X), OTP인증번호

* 하나의 컬럼을 여러 의미를 가지는 용도로 사용하지 않음을 원칙으로 하나, 불가피할 경우 ‘및’을 사용하여 명명할 수 있다. 영문명 변환시 ‘및’은 ‘N’으로 변환한다. 3개 이상의 의미를 가지는 컬럼명의 경우 주로 테이블을 구분하는 식별번호성 으로 사용되므로 도메인을 ‘식별번호’로하여 용어를 정의한다.

(예) 하나의 컬럼에 주민번호, 사업자번호가 관리되는 경우 관용적으로 사용되는 ‘주민사업자번호’ 라고 명명하여도 무방하나 두개 이상의 컬럼이 동시에 관리되는 컬럼의 경우 의미가 모호하게 변경될 가능성이 있는 것은 명확하게 ‘00번호및XX번호’로 명명할 수 있다.

* 용어는 의미가 명확히 전달되도록 함축하여 정의하고 동일한 의미를 가진 용어는 하나만 등록하여 사용한다. 복수의 정보를 관리할 필요가 있는 경우에는 엔티티 정의 단계에서 속성1명,속성2명… 등으로 용어 사이에 일련번호를 부여하여 사용한다.

(예) 용어: 배송주소, 속성명1: 배송1주소 속성명2: 배송 + 2+주소

* 용어는 60 Byte(한글 30자) 이내로 정의한다.
* 개체의 유형을 명확하게 하기 위해서 분류어를 지정하여 사용한다.

### 주제영역 명명규칙

주제영역명은 표준주제영역명을 사용한다.

* 주제영역명은 주제영역대분류명, 주제영역중분류명, [주제영역소분류명] 으로 분류한다.
* 주제영역대분류명은 필수로 정의하고 주제영역중분류명, 주제영역소분류명은 필요에 따라 정의한다.
* 표준주제영역명은‘주제영역대분류명\_주제영역중분류명\_주제영역소분류명’형식으로 표현한다.
* 표준주제영역명은 상위 및 자신의 주제영역명을 표현하고 하위의 주제영역명은 표현하지 않는다. (예) 주제영역대분류명: ‘고객’, 주제영역중분류명: ‘회원’, 주제영역소분류명: ‘부가정보’ 인 경우

표준주제영역명: 고객\_회원부가정보\_부가정보

* 주제영역소분류명 하위의 세부 주제영역이 존재할 경우 ‘\_+(세부)주제영역소분류명’을 추가하여 상세 표현한다.
* 각 주제영역은 중복을 배제한 데이터 그룹으로 분류하고 정의를 명확히 한다.

### 엔티티명 명명규칙

* 엔티티명은 [수식어 + 주제어 + 엔티티분류어] 형태의 명명규칙을 따른다.
* 수식어와 주제어는 반복될 수 있다.
* 단수의 명사 혹은 명사구로 정의하며 띄어쓰기를 하지 않는다.
* 엔티티고유명은 표준데이터 사전에 등록되어 있는 단어를 조합하여 사용한다.
* 엔티티고유명은 동일시스템 내에서 중복되지 않도록 함축적으로 정의한다.
* 엔티티명(엔티티고유명+{엔티티분류어})은 60 Byte(한글30자) 이내로 정의한다.
* 백업엔티티나 분할엔티티의 경우 엔티티명 +’\_’ + 분할(백업)단위를 사용하여 정의한다. (예) 엔티티명\_YYYYMMDD, 엔티티명\_일련번호

[표 4‑1] 엔티티 유형명

|  |  |
| --- | --- |
| 유형 | 내용 |
| 기본  (BASE) | 현재나 최종의 상태를 표시하는 기본적인 정보를 담은 엔티티타입으로 각 주제영역의 대표성 엔티티 기본과 이력이 혼합되어 있는 경우에는 기본을 사용한다.  (예) 국가기본, 회원기본, 업체기본 |
| 이력  (LOG) | 기본 엔티티가 존재하고 별도로 과거 정보를 시점별로 보관하는 엔티티인 경우. 즉 정보의 변경사항을 관리하기 위한 목적으로 생성하는 엔티티  예) 연락처이력 |
| 상세  (DTL) | 기본 상위엔티티가 존재하고 기본엔티티에 대하여 일대다의 형식으로 상세 정보를 관리하는 엔티티  (예) 업체상세 |
| 내역  (LIST) | 작업 또는 거래시점에 생성되고, 동일 형식의 정보가 낱개 형태로 변경/추가되는 엔티티(transaction성격의 데이터).  (예) 거래내역, 거래상세내역, 입금내역 |
| 관계  (REL) | 엔티티와 엔티티 사이의 관계를 정의하기 위하여 생성하는 엔티티  예) 회원관계자관계 |
| 집계  (AGRG) | 이미 존재하는 엔티티로부터 데이터 값을 추출하여 통계, 합계 등을 산출하기 위한 속성으로 구성된 엔티티타입 |
| 코드  (CD) | 코드 정보를 관리하는 엔티티타입 |
| 채번(SEQN) | 일련번호 채번용 엔티티 |
| 백업(BKUP) | 데이터의 보관을 위하여 별도로 저장하는 엔티티 |
| 임시(TEMP) | 임시로 사용하는 엔티티 |
| 이행(MIG) | 데이터 이행용 엔티티 |
| 스냅샷  (SNST) | 정기적 또는 부정기적으로 특정 시점의 데이터의 상태를 보관하기 위한 엔티티. ‘이력’ 과의 차이는 거래와 관계 없이 전산상의 목적으로 관리하는 엔티티 |
| 차원  (DMSN) | 데이터웨어하우스를 설계하기위해 사용되는 다차원모델링의 차원 엔티티  (DW에서만 허용되는 엔티티유형명) |
| 팩트  (FACT) | 데이터웨어하우스를 설계하기위해 사용되는 다차원모델링의 팩트 엔티티  (DW에서만 허용되는 엔티티유형명) |

### 속성명 명명규칙

속성명은 반드시 [수식어] + 도메인명을 사용하여 정의한다.

* 글자수는 물리명 전환을 고려하여 가급적 4개 이내의 단어 조합으로 구성한다.
* 경우에 따라 수식어 없는 도메인 명으로만 정의할 수 있으나 특수한 경우를 제외하고는 허용하지 않는다.
* 일반적으로 사용하는 용어를 축약하지 않는다.

(예) 주민번호(X) -> 주민등록번호,사번(X) -> 사원번호

* 속성명에 특정 부서명을 사용 하지 않도록 한다. 이는 부서의 변동 발생시 컬럼명의 변경을 방지 하기 위함이다.
* ‘코드’도메인에 대한 상세 수식어의 원칙은 도메인 표준에서 정의한다.

### 유형별 용어 사용의 원칙

#### 숫자가 포함된 경우

* 숫자 자체(1,2,3……)는 단어로 등록하지 않고 단일어와 조합하여 용어로 사용한다.

(예) 1^개월, 1^등급, 1^차

* 용어에 의미 없는 일련번호를 부여하기 위한 숫자를 사용하지 않고 의미를 명확하게 하는 수식어를 사용하여 정의하는 것을 원칙으로 한다. 단 불가피 하게 일련번호를 사용해야 하는 경우 1부터 순차적으로 부여 한다.
* ‘첫번째’,’두번째’ 등의 서수는 ‘1차’,’2차’ 등 차수로 변환하여 사용한다, 차수로 변환하여 의미가 모호할 경우 예외적으로 서수의 표기를 허용하나, 영문명은 차수와 동일한 단어를 사용한다.

#### 금액 관련 용어

* 원화금액만 발생하는 용어는 ‘금액’으로 정의하고 원화와 외화가 모두 발생하고 이를 구분할 필요가 있는 경우에는 ‘원화+금액’, ‘외화+금액’으로 정의한다.
* 금액이 원단위가 아닐 경우 수식어를 부여하여 컬럼명을 명확히 한다.

(예) 백만+단위+금액(백만단위금액 일 경우), 십만+단위+금액(십만단위금액 일 경우)

* 데이터를 저장할 때 'USD570'과 같이 통화 코드와 함께 사용하는 외화금액의 경우 ‘통화코드-외화금액’과 같이 속성을 원자화하고, ‘통화금액-통화별금액' 등과 같이 '통화'를 반복적으로 사용하지 않는다. 단, ‘미화금액’, '엔화금액'과 같이 통화가 확정된 경우에는 의미를 명확히 하기 위해 사용할 수 있다.

#### 금액 관련 용어의 어순

* ‘기간의수식어 + 기간, 시간 + 대상 + 장소 + 전(후) + 총 + 누적 + 금액의수식어 + 원화(외화) +도메인’ 을 기본 어순으로 한다.

기간의 수식: 최근, 과거, 이전, 최초, 최종

기간, 시간: 당월, 3개월, 1일, 전년말

대상: 해외, 국내

장소: 해외, 국내

(예)

|  |  |  |  |
| --- | --- | --- | --- |
| 전월말^총^매도^금액 | | | |
| 전월말 | 총 | 매도 | 금액 |
| <기간> | <금액범위> | <금액의수식어> | <도메인> |

* 금액 관련 용어 사용시 수식어는 용어의 실제 도메인을 유지하도록 부여 한다.
* 외화, 원화, 미화 수식어는 도메인의 직전에 사용하며, ‘환산’은 이미 내포하고 있는 의미 이므로 사용하지 않는다.

(예) 원화^물품^금액(X), 원화^물품^환산^금액(X), 원화^환산^잔액(X) -> 원화^물품^금액

* 차변, 대변 수식어는 실제도메인의 직후에 ‘차변금액’, ‘대변금액’으로 사용한다.

(예) 금액^차변(X) -> 차변^금액

수수료^차변(X) -> 수수료^차변^금액

미지급금^대변(X) -> 미지급금^대변^금액

#### 날짜 관련 용어

* ‘개월’이 수식어로 사용될 경우 기간을 포함한 의미로 사용한다. 기간의 의미를 포함하는 경우 ‘개월간’으로 사용하지 않고 ‘개월’로 사용하며, ‘이전’의 의미가 있을 경우는 반드시‘개월전’으로 사용한다.

(예) 최근^1개월간^접속^여부(X) -> 최근^1개월^접속^여부

* ‘일자’의 사용에 있어서 시간은 숫자 형태의 어떠한 작업이나 시간이 경과한 수를 의미할 경우만 사용한다. HH24MISS형태로 표시되는 정확한 시각을 나타낼 ‘일시’를 사용한다.

#### 총 vs 합계

합계 보다는 ‘총’이 금액 및 수량 관련 용어에 더 범용적으로 사용 될 수 있으므로 ‘합계’ 보다는 ‘총’을 사용 한다.

#### 현재 vs 전체

용어의 사용시 ‘현재주소’, ‘업체전체명’, ‘현재잔액’ 등으로 현재 시점을 나타내거나 기본적인 사항을 기술하지 않는다. 즉, 수식어가 없는 용어를 기본 용어로 취급한다

(예) 업체명 – 현재 업체 전체명의 의미를 내포함

## 영문 용어(물리명) 표준정의 원칙

### 물리 데이터 오브젝트 명명규칙

* 정의된 한글명에 대하여 등록된 단어 단위로 영문약어를 사용하여 자동 전환하며 단어 구분자로 Under Bar(\_)를 사용한다.
* 합성어를 단일어 보다 우선 적용한다.
* 영문용어는 단어에 대한 영문약어의 조합으로 사용하되 25자를 초과하는 경우 별도의 영문단축명을 부여할 수 있다.(특정DBMS 오브젝트 최대 자리 수 및 접두어를 고려하여 25자 이내로 제한)

### 테이블명 명명규칙

* 엔티티명에 사용된 단어의 영문약어를 '\_'를 구분자로 하여 자동 변환된 명칭을 사용한다.
* 테이블명은 엔티티영문약어명 형식으로 정의한다.
* 엔티티명에 대한 영문약어명은 표준데이터 사전에 등록된 단어의 영문약어 조합으로 구성한다.
* 테이블명은 25Byte이내에서 명명하되 테이블명이 25Byte를 초과하는 경우 별도의 단축명을 부여하여 사용할 수 있다

### 컬럼명 명명규칙

* 속성명에 사용된 단어의 영문약어를 '\_' 를 구분자로 하여 자동 변환된 명칭을 사용한다.
* 전환된 물리명은 Under Bar(\_)를 포함하여 25Byte 이내로 구성한다.
* 생성된 영문명이 25자를 초과하는 경우에는 반드시 영문단축명을 적용한다.
* 컬럼명은 테이블 내에서 유일성을 유지해야 한다

### 기타 물리 데이터 오브젝트 명명규칙

기타 물리 오브젝트의 경우, [구분자 접두어 + '\_' + 테이블명 + ’\_’ + 일련번호 ] 형식을 기본으로 한다. 구분자 접두어는 데이터베이스, 테이블, 프로시저 등과 같은 물리 오브젝트의 개체 형식을 구분하기 위한 접두어를 말한다.

[표 4‑2] 물리 오브젝트 별 명명 규칙

| **구분** | **설명** |
| --- | --- |
| DB명(서비스명) | 형식: 접두어(필요시)||서비스명||DB  DB명은 각 시스템별로 DB가 하나일 경우 가급적 시스템의 3자리 영문약어를 사용한다.(1개 이상일 경우 해당 DB의 특성에 맞는 영문약어를 사용한다.)  (예) HKD\_DEALDB |
| Table Space명 | 형식: TBS\_ || Table Space명 |
| DB Link명 | 형식: O\_ (운영)/S\_(스테이징)/D\_(개발)/T\_(테스트)+ 대상 서비스명  (예) O\_DEAL /S\_DEAL/ D\_DEAL / T\_DEAL |
| INDEX명 | 형식: IDX\_ || 테이블명||\_||UNIQUE : U||Serial(01~99)  (예) IDX\_ACS\_CUST\_M\_U01 |
| CONSTRAINT명 | 형식: PRIMARY Key: PK\_||테이블명  FOREIGN Key: FK\_||Child테이블명 || \_ Serial(01~99)  CHECK: CK\_||테이블명 || \_Serial(00~99)  (예) PK\_ACS\_CUST\_M , FK\_ACS\_CUST\_M\_01 , CK\_ACS\_CUST\_M\_01 |
| VIEW명 | 형식: VIEW\_|| 뷰영문약어명 || \_ Serial(01~99)  (예) VW\_ACS\_CUST\_REL  원본 테이블과 동일한 형태의 뷰로 정의 할 경우에는 뷰 영문약어명을 테이블명을 기준으로 구분자 접두어 'VIEW'와 함께 작성한다. |
| Package/  Procedure/  Function명 | 형식: Package : PKG\_ || 패키지영문약어명 || \_Serial(01~99)  Procedure : SP\_ || 프로시져영문약어명 || \_Serial(01~99)  Function : FN\_ || 함수영문약어명 || \_Serial(01~99)  (예) PKG\_GET\_CUST\_SMRY  SP\_CUST\_SMRY  FN\_GET\_CUST\_SMRY |
| Trigger명 | 형식: TRG || 테이블명 || \_Serial(01~99)  (예) TRG\_ACS\_CUST\_M\_01 |

# 도메인 개요

## 도메인 정의

* 도메인은 각 속성의 일반적인 특징을 나타내는 데이터의 성격을 분류한 것으로써 속성이 취할 수 있는 값의 집합이라 할 수 있다.
* 일반적으로 엔티티에 존재하는 속성들의 실제 값들이 나타내는 공통적인 특징을 그룹화 하고 특정 값을 제한함으로써 값에 대한 표준화와 명명에 대한 표준화를 준수 할 수 있다.

[표 5‑1] 도메인 예

|  |  |  |
| --- | --- | --- |
| 용어 | 구성 | 도메인 |
| 입금계좌번호  환불계좌번호 | 입금 + 계좌번호  환불 + 계좌번호 | 계좌번호 |
| 결재요청금액  총매도금액  정산대상금액 | 결재 + 요청 + 금액  총 + 매도 + 금액  정산 + 대상 + 금액 | 금액 |
| 서비스시작일자  이체신청일자 | 서비스 + 시작 + 일자  이체 + 신청 + 일자 | 일자 |

## 도메인의 사용 목적

### 용어 표준화

용어의 의미를 명확히 하고 해당 속성이 가질 수 있는 데이터의 범위를 명확히 하기 위하여 도메인을 적용한다. 또한 이음동의어가 발생할 소지가 있는 경우를 위해 해당 용어에 알맞은 도메인을 사용한다. (예, 년월일-일자, 순번-일련번호)

### 인포타입의 일괄 부여/변경

인포타입(Information Type : Data Type / Length)은 해당 도메인이 가질 수 있는 데이터 타입과 길이를 지정하는 것으로써, 향후 컬럼의 데이터 타입과 길이를 부여하고 변경하는 기준이 되는 단위 별로 도메인 혹은 인포타입(Information Type) 을 지정한다.

### 속성값 유효성 제한

속성값에 범위가 있거나 규칙이 있어서 향후 속성값 유효성 제한을 위하여 도메인을 부여한다.

## 도메인 분류 기준

용어는 무엇에 관한(수식어) 어떤 것(도메인) 으로 구분하여 볼 수 있다.

그렇지만 하나의 용어를 놓고 어디까지가 수식어이고 어디까지가 도메인 인지에 관하여 이견이 있을 수 있다. 예를 들어 '평가기준금액' 라는 용어를 ‘평가 + 기준금액’으로 볼 수 있고 ‘평가기준 + 금액’으로 볼 수도 있다.

이 경우 기준금액을 도메인으로 할지 금액을 도메인으로 할지는 위 도메인 사용 목적에 비추어 검토할 수 있다. 우선 용어 표준화 측면에서 도메인을 기준금액으로 하지 않고 금액으로 하더라도 기준금액을 다른 이름으로 부를 가능성(이음동의 가능성)은 없어 보인다.

Data Type / Length 일괄 부여/변경 측면에서 보면 기준금액의 Data Type이나 Length를 일괄해서 부여하거나 변경할 필요성이 있을 것으로 보인다.

그러므로 기준금액을 별도의 도메인으로 하는 것이 금액을 도메인으로 하여 다른 번호와 같은 도메인을 쓰는 것보다 합리적일 것으로 보인다.

속성 값 유효성 체크 측면을 보면 기준금액에 특별한 규칙이 있다면 별도 도메인 적용 하는 것이, 그렇지 않으면 도메인 적용 하지 않는 것이 합리적일 것이다.

## 도메인 구성 요소

### 도메인그룹

도메인 그룹은 도메인의 성격이 유사한 도메인들을 그룹화 해서 관리하는 단위로 예를 들면 금액, 일자, 내용, 율 등의 도메인 그룹이 존재한다.

### 도메인 유형

관리되는 데이터의 성격이 동일하여 동일한 인포타입을 부여할 수 있는 도메인을 그룹핑하여 유형을 관리한다. 즉, 인포타입을 부여하는 단위가 되며 이는 동일한 성격의 도메인에 대하여 별도로 인포타입이 부여되는 것을 방지하기 위함이다.

### 도메인명

도메인은 각 속성의 일반적인 특징을 나타내는 데이터의 성격을 그룹화해서 명명한 것으로 실제 속성명에 적용되는 명칭이다.

### 인포타입(Information Type)

인포타입은 Information Type의 약어로 해당 도메인에서 사용할 수 있는 데이터 타입이 결합된 형태이다. 영문으로는 Information Type 한글로는 인포타입으로 명명한다. 인포 타입은 향후 Data Type-Length를 부여 받고 Data Type-Length변경의 기초 단위가 되며 부여된 Data Type-Length는 향후 Database Generation시 이용된다.

[표 5‑2] 도메인 구성 요소 예

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 도메인 그룹 | 도메인 유형명 | 도메인명 | 인포타입 | 데이터 타입(길이) |
| 명 | 고객명 | 고객명 | 고객명V40 | VARCHAR(40) |
| 업종명 | 업종명 | 업종명V50 | VARCHAR(50) |
| 금액 | 금액 | 금액 | 금액N30 | NUMBER |
| 금 | 가지급금 | 금N30 | NUMBER |
| 미수금 |
| 세 | 소득세 | 세N30 | NUMBER |
| 주민세 |

# 도메인 분류

도메인은 크게 번호 도메인, 제한형 도메인, 코드 도메인의 3가지로 분류할 수 있으며 각 도메인 분류별로 세부 도메인 그룹을 가진다.

[표 6‑1] 도메인 분류

|  |  |  |
| --- | --- | --- |
| 도메인 분류 | 도메인 그룹 | 설명 |
| **번호 도메인** | 식별번호 | 엔티티에서 독립적으로 식별자 역할을 하는 속성을 정의하기 위한 도메인으로 자기자신만의 도메인을 갖는 도메인 (단, 코드도메인 제외)  예) 고객번호, 법인등록번호, 주문번호, 거래번호 |
| 일반번호 | 식별번호 도메인을 제외한 각 자리 별 특정 의미를 가지거나 체계를 가지고 관리되어야 하는 속성을 정의하기 위한 도메인  예) 전화번호, 일련번호 |
| **코드 도메인** | 코드 | 코드화 하여 데이터를 분류, 구분하기 위한 속성을 정의하기 위한 도메인  예) 거래구분코드, 표준산업분류코드, 사용여부 |
| **제한형 도메인** | 데이터 값을 제한하는 도메인 | |
| **금액** | 화폐의 가치를 수로 표현하기 위한 도메인  예) 금액, 세금, 잔액, 총액 |
| **건수** | 객체의 개수나 량을 수로써 표현하기 위한 도메인. 일반적인 측량 단위도 포함됨  예) 건수, 개수, 매수 |
| **율** | 비율을 수로 표현하기 위한 도메인  예) 금리, 이율, 비율 |
| **명** | 문자 형식으로 객체에 대한 식별을 표현하기 위한 도메인  예) 명, 주소, ID, 장소, 기관명 |
| **일자** | 특정 사건이 일어난 시점 또는 시점과 시점 간의 시간을 표현하기 위한 도메인  예) 일자, 일, 년, 월, 일, 일시, 반기, 분기 |
| **내용** | 서술 형식으로 상세 내용을 표현한 것으로 자유 형식의 텍스트를 표현하기 위한 도메인  예) 내용, 요약, 비고, 설명 |

# 제한형 도메인 표준

본 문서에서는 도메인 그룹별로 공통적인 사항을 기술하며 각 세부 도메인은 ‘표준 도메인 사전’을 참고 한다.

## 명

* ‘명’ 도메인 그룹의 대표 도메인은 ‘명’ 이며 관용적으로 자주 사용되는 도메인 및 특별히 관리를 필요로 하는 인명(人名)이나 기관명은 세부 도메인으로 도출해서 관리한다. 예) 업체명, 기관명
* ‘명’ 과 관련되어서 수식어(한글, 영문, 한자, 약어)가 없을 경우 기본적으로 한글전체명을 의미한다.

### 물리속성명(프로그램명 VS 프로그램영문명 VS 프로그램한글명 등)

‘프로그램명’,‘테이블명’,‘컬럼명’,‘데이터베이스명’등 시스템오브젝트 및 물리적 속성에 대한 명칭은 영문명이 기본이므로 의미를 명확하게 하기 위해 ‘한글명’,‘영문명’ 도메인을 반드시 적용하여 구분한다.

### ID

* ‘ID’는 Identifier로 식별번호 도메인으로 분류하여 사용한다.
* ID 와 번호 에 대한 개념이 모호하지만 기본적인 ID의 경우 영숫자가 포함된 것을 나타내며 식별번호의 의미를 포함할 경우 식별번호 도메인으로 정의 한다.
* 단, 시스템 오브젝트에 대한 용어 정의시 ID를 사용할 수 있으나 그룹 도메인으로 사용한다.

예) 트리거ID, 작업ID, 타스크ID

### 주소

[표 7‑1] 우편번호 주소 체계

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 주소 | 우편주소 | 발송용주소 | 시도명 |  | |
| 시구군명 |  | |
| 배달용주소 | 읍면동명 |  | |
| 배달처명 | 리명 | |
| 다량  배달처명 | 아파트명 |
| 건물명 |
| 사서함명 |
| 부속주소 | 번지명 | 본번지명 |  | |
| 부번지명 |  | |
| 마을명 |  |  | |
| 아파트명 |  |  | |
| 동명 |  |  | |
| 호명 |  |  | |
| 동호명 |  |  | |

※도메인으로 선정된 항목은 밑줄로 표시

## 내용

내용은 특정한 데이터의 부가적 설명이나 의견, 사유 등의 설명 정보를 저장할 때 사용되는 도메인 그룹이다.

### 전문내용

‘전문내용’ 은 외부기관에서 정해진 길이로 데이터가 관리 되어야 하므로 별도 도메인으로 관리한다.

## 금액

### 금액 VS 금

* 모든 금액은 ‘금액’ 도메인을 기준으로 한다.
* ‘금’은 단독 도메인으로 허용하지 않으며 관용적으로 사용하는 용어나 ‘금’과 ‘금액’의 의미가 다른 용어일 경우 수식어와 결합된 형태로 ‘금’ 도메인을 정의하고‘금’으로 사용되는 도메인을 별도로 관리한다.

예) 하자보수보증금,환수금, 할증요금

* ‘액’도메인은 사용하지 않으며 모두 ‘금액’으로 사용하나 관용적으로 사용되는 도메인만 별도 관리한다.

예) 잔액, 차액

### 가격 VS 가

‘가격’은 도메인으로 허용하지 않으며 ‘가’ 도메인유형하에 수식어와 결합된 ‘가’로 사용되는 도메인을 별도로 관리한다.

예) 프리미엄단가, 판매단가, 취득원가

### 세금 VS 세

세금은 국가 및 지방에서 이미 정해진 각 종류별 세금은 수식어가 결합된 형태로 ‘세’ 도메인 유형으로 각 종류별 세금 도메인을 관리하며 그 이외의 세금 금액을 나타내는 경우는 ‘세금’ 도메인을 사용한다.

예) 소득세, 주민세, 법인세, 부가세

### 이익 VS 손실

‘~익’, ‘~손’ 은 ‘이익’, ‘손실’을 기본으로 하나 ‘~익’, ‘~손’으로 관리되어야 할 용어는 각 도메인 유형으로 별도 도메인으로 관리한다.

예) 거래손익, 영업이익, 기타정산수익

### 원화금액 VS 외화금액

* 금액을 명시할 경우 통화에 대한 수식어가 없는 경우 ‘원화’를 의미하며 일반적으로 외화가 들어가는 경우나 원화와 외화 에 대한 금액을 같이 가지고 있을 경우에는 모호성을 없애기 위해서 ‘외화금액’을 사용한다.
* 특히 외환관련 업무상의 금액이 외화와 구분되어야 할 시 반드시 ‘원화+금액’, ‘외화+금액’을 구분해서 사용하며 일반적으로는 ‘금액’을 사용한다.

## 수량

### 건수 VS 수

기본적으로 ‘건수’, ‘매수’, ‘개수’ 도메인을 기본으로 하나 관용적으로 ‘~수’로 사용하는 용어에 대해서는 수식어와 결합된 형태로 ‘수’ 도메인 유형으로 ‘수’로 사용되는 도메인을 별도로 관리한다.

예) 업체수, 회원수

### 시간 VS 시각

* 시간은 숫자 형태의 어떠한 작업이나 시간이 경과한 수를 의미한다. 이는 시간, 분, 초를 포함한 경과 시간을 나타낼 때 사용하며 시간, 분, 초 각각의 경과 시간을 나타낼 경우 ‘시간수’, ‘분수’, ‘초수’를 사용한다.
* HH24MISS형태로 표시되는 정확한 시각을 나타낼 경우 ‘날짜’ 도메인의 ‘시각’을 사용한다.

## 날짜

### 일 VS 일자

* ‘일자’는 특정 년월일을 나타내는 도메인으로 사용한다. (회신일자: 2018/03/16)
* ‘일’ 특정일을 나타내는 도메인으로 사용한다. (결제기준일: 21일, 25일)

## 율

### 금리 VS 이율

* ‘금리’는 자금시장에서 구체적으로 거래되고 있는 자금의 사용료로 대외적으로 공시되는 기준이 되는 의미로 사용한다.
* ‘이율’은 ‘원금에 대한 이자의 비율, 즉 이자 산출에 기초가 되는 비율’의 의미인 경우 사용하며 ‘이자율’은 사용하지 않는다.

### 율 VS률

* ‘율/률’은 문법적으로 구분되어 사용되어 지나 사용자가 명확히 구분하여 사용하기 어려워 표준용어의 혼용 사용이 우려되므로‘율’로 통일하여 사용한다. 단, 확률은 예외로 한다.
* 문법적으로 구분 가능한 ‘율/률’의 차이

‘~율’은 모음으로 끝나거나 ‘ㄴ’받침을 가진 일부 명사 뒤에 붙임

‘~률’은 ‘ㄴ’받침을 제외한 받침 있는 일부 명사 뒤에 붙임

# 번호도메인표준

번호도메인의 경우 당사 공통적으로 사용되거나 외부기관에서 정의된 공통 도메인과 각 업무에서 개별적으로 정의하여 일부 업무에만 적용되는 도메인으로 나 눌 수 있다.

[표 8‑1] 도메인 예1

|  |  |  |  |
| --- | --- | --- | --- |
| 도메인 명 | 인포타입 | 데이터타입 | 번호 체계 |
| 계좌번호 | 계좌번호V14 | VARCHAR(14) | 외부기관에서 정의 |
| 사업자등록번호 | 사업자등록번호V10 | VARCHAR(10) | 외부기관에서 정의 |
| 법인등록번호 | 법인등록번호V13 | VARCHAR(13) | 외부기관에서 정의 |

[표 8‑2] 도메인 예2

|  |  |  |  |
| --- | --- | --- | --- |
| 도메인 명 | 인포타입 | 데이터타입 | 비고 |
| 거래요청번호 | 거래요청번호V22 | VARCHAR(22) |  |
| 관리번호 | 관리번호V25 | VARCHAR(25) |  |

### 번호 VS 일련번호

* ‘번호’ 도메인은 일정한 체계를 가지거나 특정 자리의 의미가 존재하는 속성을 정의할 때 사용하는 것을 원칙으로 한다.
* ‘번호’만으로는 도메인으로 사용할 수 없으며 번호에 대한 의미를 명확히 하는 수식어를 부여한다.
* ‘일련번호’ 도메인은 숫자 값이 무의미하게 연속으로 부여되는 경우에 사용한다. 그 자체로는 의미가 없으며 단순하게 데이터를 구분하기 위해서 사용하는 경우에도 도메인 자체를 독립적으로 사용할 수 없고 반드시 수식어를 포함하여 사용한다.

### 실명번호

* ‘실명번호’의 경우 주민등록번호, 사업자등록번호, 여권번호등 종류별 실명번호를 동시에 관리하는 경우 사용한다.
* 2가지 이상의 도메인을 동시에 내포할 경우 합성된 형태로 사용한다.

예) 주민사업자등록번호

# 코드 도메인 표준

코드는 사물을 표현하는 기호 또는 부호화된 정보로서, 표준화의 대상 중 데이터 속성을 이해하기 쉽고 기준에 의해 분류할 수 있는 형태로 기술하는 것을 말한다

### 코드 유형명

코드를 유형별로 분류하여 관리함으로써 코드 유형명을 기준으로 코드도메인을 정의하도록 한다. 이는 코드 유형이 동일함에도 불구하고 각각의 코드도메인이 부여되는 것을 방지함으로써 코드도메인 및 인포타입의 등록을 최소화하여 업무효율성을 향상시키기 위함이다.

[표 9‑1] 코드 유형명

|  |  |  |
| --- | --- | --- |
| **코드 유형명** | **코드 유형 설명** | **코드 용어 예시** |
| 여부 | [與否] 그러함과 그러하지 아니함 | 확인^여부 |
| 유무 | [有無] 있음과 없음 | 사고^유무 |
| 구분 | [區分] 따로따로 갈라서 나눔 | 계약^구분^코드 |
| 상태 | [狀態] 사물·현상이 놓여 있는 모양이나 형편 | 계좌^상태^코드 |
| 종류 | [種類] 사물의 부문을 나누는 갈래 | 카드^종류^코드 |
| 유형 | [類型] 어떤 비슷한 것들의 본질을 개체로서 나타낸 것, 또는 그것들의 공통되는 성질이나 모양 | 평가^유형^코드 |
| 분류 | [分類] 사물을 공통되는 성질에 따라 종류별로 나눔 | 업종^분류^코드 |
| 사유 | [事由] 일의 까닭. 연고. 연유(緣由) | 해지^사유^코드 |
| 방법 | [方法] 어떤 일을 해 나가거나 목적을 이루기 위하여 취하는 수단이나 방식 | 결제^방법^코드 |
| 방식 | [方式] 일정한 방법이나 형식 | 구조^방식^코드 |
| 형태 | [形態] 사물의 생김새나 모양 | 선적^형태^코드 |
| 거래 | [去來] 자산·부채·자본의 증감 변화를 일으키거나 수익 또는 비용을 발생케 하는 모든 사항 또는 사상 | 거래^구분^코드 |
| 관계 | [關係] 둘 이상의 사람,사물,현상 따위가 서로 관련을 맺거나 관련이 있음 | 업체^관계^코드 |
| 용도 | [用途] 쓰이는 곳이나 쓰는 법 | 사용^용도^코드 |
| 자격 | [資格] 일정한 신분이나 지위를 가지거나 일정한 일을 하는 데 필요한 조건이나 능력 | 참여^자격^코드 |
| 단위 | [單位] 길이,넓이,무게,수효,시간 따위의 수량을 수치로 나타낼 때 기초가 되는 일정한 기준 | 금액^단위^코드 |
| 등급 | [等級] 높고 낮음이나 좋고 나쁨 따위의 차이를 여러 층으로 구분한 단계 | 인증^등급^코드 |
| 규모 | [規模] 사물이나 현상의 크기나 범위 | 기업^규모^코드 |
| 결과 | [結果] 어떤 원인으로 결말이 생김. 또는 그런 결말의 상태 | 응답^결과^코드 |
| 그룹 | 함께 행동하거나 공통점이 있어 한데 묶일 수 있는 사람들의 무리 | 업종^그룹^코드 |
| 주기 | [週期] 현상이나 특징이 한 번 나타나고 다음 번 되풀이되기까지의 기간 | 리스^지급^주기^코드 |
| 지역 | [地域] 일정한 땅의 구역이나 땅의 경계, 또는 그 안의 땅 | 관할^지역^코드 |
| 조건 | [條件] 어떠한 일을 진행되게 하거나 성립되게 하기 위하여 갖추어야만 할 요소 | 가격^조건^코드 |
| 목적 | [目的] 어떤 일을 통해 이루고자 하는 바. 또는, 어떤 일을 하는 동기나 이유 | 거래^목적^코드 |
| 기준 | [基準] 어떤 일을 판단하거나 무엇을 구별하기 위해 본보기로 삼아야 할 일정한 원칙이나 수준이나 대상 | 정산^기준^코드 |
| 기타 | 정의된 유형에 속하지 않고 개별적으로 관리되는 경우 | 상품^코드 |

### 코드 도메인 명명규칙

코드도메인은 {수식어} + 코드유형명 + ‘코드’ 형태로 명명한다. 수식어는 코드유형에 따라 붙이는 경우와 그렇지 않은 경우가 있다.

단, ‘여부’와 ‘유무’는 코드로 분류하지만 예외적으로 ‘코드’를 붙이지 않는다.

또한 코드 자체의 의미가 명확하고 관용적으로 사용될 경우 예외적으로 `코드유형명`을 붙이지 않는다. 예) 은행구분코드 -> 은행코드

[표 9‑2] 코드 유형별 코드 도메인 명명규칙

|  |  |  |  |
| --- | --- | --- | --- |
| **코드**  **유형명** | **코드 도메인**  **형식** | **수식어**  **(단어기준)** | **코드도메인**  **예시** |
| 여부 | 코드유형명 | 붙이지 않음 | '여부' |
| 유무 | 코드유형명 | 붙이지 않음 | '유무' |
| 구분  상태 | {수식어} + 코드유형명 + '코드' | 하나만 붙이거나  붙이지 않음 | '처리^구분^코드', '구분^코드'  '원장^상태^코드', '상태^코드' |
| 종류 외  20개 유형 | 코드유형명 + '코드'  ☞ 유형,분류,사유,방법,방식,형태,  거래,관계,용도,자격,단위,등급,규모,  결과,그룹,주기,지역,조건,목적,기준 | 붙이지 않음 | '종류^코드', '사유^코드' |
| 기타  (유형명 없음) | {수식어} + '코드' | 하나만 붙이거나  붙이지 않음 | '상품^코드', '계정과목^코드'  '코드' |

### 여부VS 유무

‘여부’와 ‘유무’는 분류와 관련된 도메인으로 코드도메인에 포함시켜 사용한다.

`여부`와`유무`의 의미는 명확히 다르나 단어의 관리 특성상 동일 물리명을 허용하지 않음으로 유무도 여부로 표기한다.

[표 9‑3] 여부/유무 비교

|  |  |  |
| --- | --- | --- |
| 코드 유형명 | 정의 및 적용방법 | 코드 용어 예시 |
| 여부 | ‘이다/아니다’, ‘하다/하지않다’ 의 의미로 사용하며, 값은 ‘Y’ 와 ‘N’ 으로만 표현한다. True/False | 결혼유무 -> 결혼여부 |
| 유무 | ‘있음/없음’의 의미로 명사 뒤에 붙여서 사용하며 값은 ‘Y’와 ‘N’ 으로만 표현한다. Exist/Not Exist | 가족유무 -> 가족여부  사고유무 -> 사고여부 |

### 구분 VS 유형 VS 분류 VS 종류

[표 9‑4] 구분/유형/분류/종류 비교

|  |  |  |
| --- | --- | --- |
| 코드 유형명 | 정의 및 적용방법 | 코드 용어 예시 |
| 구분 | - 모든 코드에 대한 기본 유형명으로 사용  - 코드 자체의 의미가 명확하고 관용적으로 사용될 경우 생략 가능함. | 거래구분코드, 관리구분코드 |
| 유형 | - ‘구분’을 사용하여 명명하는 경우와 명확하게 구분하기 어려움.  - ‘유형’을 반드시 사용해야만 하는 경우에만 제한적으로 사용하고 기본적으로 ‘구분’을 사용하도록 함. | 고객유형코드, 거래유형코드 |
| 분류 | - 코드값을 체계화 하여 관리하는 경우 사용  - 대·중·소·세분류 체계를 갖는 코드에 대해서만 ‘분류’를 사용 | 대분류코드, 중분류코드,  소분류코드, 세분류코드 |
| 종류 | - 사물을 나누어 구분하는 경우 사용 | 카드종류코드, 상품종류코드 |

### 외부기관 정의 코드

외부기관에서 정의하고 당사에서 사용하는 코드는 정의한 외부기관의 용어명을 따르도록 한다.

# 인포타입(Information Type)

## 인포타입 명명 규칙

인포타입의 명명 규칙은 도메인유형명과 데이터 유형과 자리수가 결합된 형태로 명명한다

**인포타입명** = **도메인유형명** + **데이터유형** + **자릿수**

[표 10‑1] 인포 타입 명명 예

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 도메인 그룹 | 도메인 유형 | 도메인 | 인포타입 | 데이터 길이 |
| 명 | 담당자명 | 담당자명 | 담당자명V35 | VARCHAR2(35) |
| 명 | 명 | 명V100 | VARCHAR2(100) |
|  |  |  |  |  |
| 금액 | 금액 | 금액 | 금액N15 | NUMBER(15) |
| 금 | 환수금 | 금N15 | NUMBER(15) |
| 세 | 소득세 | 세N15 | NUMBER(15) |
| 법인세 |

## 인포타입 부여 규칙

* 제한형 도메인은 하나의 도메인유형에 대하여 하나 이상의 인포타입을 가질 수 있다. 즉, 하나의 도메인이 여러 데이터 타입/ 길이를 가질 수 있다.
* 하나의 도메인이 하나 이상의 인포타입을 갖는 경우 대표 인포타입을 지정하여 관리함으로써 도메인 내에서의 인포타입 불일치 가능성을 최소화한다.
* 인포타입을 정할 때 데이터 타입과 길이가 상이한 것을 하나의 인포타입으로 해서는 안 된다. 예를 들면 금액 도메인을 쓰는 모든 속성을 하나의 인포타입으로 할 경우 물리 속성 생성시 변경 요건이 있어도 모두 변경해야 하는 폐단이 있다. 따라서 이 경우에는 예상되는 속성 길이에 따라 금액18, 금액18,3, 금액23,3 등 복수 개의 인포타입으로 하는 것이 합리적이다.

## 인포타입 데이터유형

인포타입 명명시 필요한 인포타입 유형은 대표적인 데이터타입을 나타낸다.

[표 10‑2] 대표적 데이터 타입

|  |  |  |  |
| --- | --- | --- | --- |
| 데이터 유형 | 데이터 타입 | 예제 | |
| **V** | VARCHAR | 명V30 | VARCHAR(30) |
| **C** | CHAR | 코드C10 | CHAR(10) |
| **TX** | TEXT | 내용TX | TEXT |
| **LR** | LONG RAW | 이미지LR | LONG RAW |
| **N** | NUMBER | 번호N10 | NUMBER(10) |
| **DT** | DATE | 일자DT | DATE |
| **TS** | TIMESTAMP | 시각TS | TIMESTAMP |
| **I** | INT | 금액I | INT(10) |

## 표준 인포타입

* 표준 인포타입은 도메인 그룹에 대한 인포타입 중에서 표준으로 지정된 것이다. 표준 인포타입 중에서 대표 인포타입을 선정하여 도메인 그룹 특성에 맞는 데이터 타입과 유형을 일관성 있게 부여할 수 있다.
* 아래의 표준 인포타입은 DBMS에 종속적이지 않으며 실제 데이터베이스의 테이블 생성시 해당 DBMS에 맞게 변환해서 사용한다. 따라서 표준 인포타입에서 선정한 데이터 유형과 실제 데이터베이스상의 데이터 타입은 서로 다를 수 있다.

[표 10-3] 기본 데이터 타입 대 대표 인포타입 매핑

| **도**  **메**  **인**  **그**  **룹** | **대표**  **도메인** | **기본** | | | | | **표준** | | **비고** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **데이터**  **타입** | **유형** | **DBMS** | **표준 여부** | **인포**  **타입** | **표준**  **인포**  **타입** | **대표**  **인포**  **타입** |
| **명** | 명 | VARCHAR2 | V | Oracle | Y | 명V | 명V | 명V |  |
| 명 | VARCHAR | V | Mysql | Y | 명V |
| 명 | CHAR | C | Oracle | N | 명C |
| 명 | CHAR | C | Mysql | N | 명C |
| **내용** | 내용 | VARCHAR2 | V | Oracle | Y | 내용V | 내용V | 내용V |  |
| 내용 | VARCHAR | V | Mysql | Y | 내용V |
| 내용 | CHAR | C | Oracle | N | 내용C |
| 내용 | CHAR | C | Mysql | N | 내용C |
| 내용 | BLOB | BL | Oracle | Y | 내용 BL |
| 내용 | CLOB | CL | Oracle | Y | 내용 CL |
| 내용 | TEXT | TX | Mysql | Y | 내용 TX |
| **번호** | 번호 | VARCHAR2 | V | Oracle | Y | 번호V | 번호V | 번호V |  |
| 번호 | VARCHAR | V | Mysql | Y | 번호V |
| 번호 | CHAR | C | Oracle | N | 번호C |
| 번호 | CHAR | C | Mysql | N | 번호C |
| 번호 | NUMBER | N | Oracle | Y | 번호N | 번호N | 번호N |  |
| 번호 | INT | I | Mysql | Y | 번호I | 번호I |
| 번호 | DECIMAL | D | Mysql | Y | 번호D | 번호D |
| 일련번호 | NUMBER | N | Oracle | Y | 일련번호N | 일련번호N | 일련번호N |  |
| 일련번호 | DECIMAL | D | Mysql | Y | 일련번호D | 일련번호D |
| **금액** | 금액 | NUMBER | N | Oracle | Y | 금액N | 금액N  금액BI | 금액N |  |
| 금액 | BIGINT | BI | Mysql | Y | 금액BI |
| 금액 | DECIMAL | D | Mysql | Y | 금액D | 금액D |
| **수량** | 수량 | NUMBER | N | Oracle | Y | 수량N | 수량N | 수량N |  |
| 수량 | DECIMAL | D | Mysql | Y | 수량D | 수량D |
| **율** | 율 | NUMBER | N | Oracle | Y | 율N | 율N | 율N |  |
| 율 | DECIMAL | D | Mysql | Y | 율D | 율D |
| **코드** | 코드 | VARCHAR2 | V | Oracle | N | 코드V | 코드C | 코드C |  |
| 코드 | VARCHAR | V | Mysql | N | 코드V |
| 코드 | CHAR | C | Oracle | Y | 코드C |
| 코드 | CHAR | C | Mysql | Y | 코드C |
| **날짜** | 날짜 | VARCHAR2 | V | Oracle | Y | 날짜V | 날짜V | 날짜V |  |
| 날짜 | VARCHAR | V | Mysql | Y | 날짜V |
| 날짜 | CHAR | C | Oracle | N | 날짜C |
| 날짜 | CHAR | C | Mysql | N | 날짜C |
| 날짜 | DATE | D | Oracle | Y | 날짜D | 날짜D | 날짜D |  |
| 날짜 | DATE | D | Mysql | Y | 날짜D |
| 날짜 | DATETIME | D | Mysql | Y | 날짜D |
| 날짜 | TIMESTAMP | T | Oracle | Y | 날짜T | 날짜T | 날짜T |  |
| **기타** | 비표준 | LONG |  | Oracle | N |  | 표준으로  대체 | 표준으로  대체 | 'VARCHAR2' 또는 CLOB, BLOB로 대체 |
| 비표준 | RAW |  | Oracle | N |  |

# 코드 표준화

## 코드 표준화 개요

* 코드 표준화는 코드ID명 표준화와 코드 인스턴스 표준화가 있다.
* 코드ID명 표준화는 코드의 의미가 명확히 표현될 수 있도록 명명 규칙을 정의하고 이에 따라 코드ID명 이 정의될 수 있도록 표준화 작업을 수행한다.
* 코드 인스턴스 표준화는 코드 종류 및 코드 부여 방식에 따라 코드 인스턴스 부여 원칙을 정의하고 이 원칙에 위배되거나 중복 관리되는 코드를 표준화하는 작업을 수행한다.

### 코드 표준화 목적

현행 코드는 각 개발자 별로 필요에 따라 추가하여 사용함에 따라 이미 존재하는 코드를 중복하여 사용하거나 동일한 의미를 가진 코드에 대해 다양한 코드 명칭이 존재하고 있어 이에 대한 표준화를 수행함으로써 코드 관리의 일관성 및 효율성을 향상시키기 위함이다.

### 코드 표준화 원칙

* 원칙적으로 당사에서 사용하는 모든 코드는 통합 관리한다.
* 업무적으로 동일한 의미의 코드나 유사한 코드를 통합한 후 표준화된 코드ID명과 코드를 부여한다.
* 목록성 코드의 경우 참조정보(DB명, 테이블명, 컬럼명)만 관리하며 별도의 코드정보(코드, 코드명)를 관리하지 않는다.

## 코드의 종류

코드의 구성방법에 따라 코드 종류를 분류하고, 이에 따라 코드 관리방법을 정의한다.

### 코드 종류

코드의 구성벙법에 따라 단일코드, 목록성코드, 여부코드로 분류한다.

[표 11‑1 코드 종류]

|  |  |
| --- | --- |
| 코드종류 | 설명 |
| 단일코드 | 하나의 코드로 Key 가 구성되며 코드와 코드명만 관리한다.  예) 원장구분코드 → 1: 출금, 2: 입금, 3: 가수금 |
| 목록성코드 | 하나 이상의 코드로 Key 가 구성되며 코드와 코드명 이외의 부가적인 속성도 관리한다.  예) 우편주소코드 → 우편번호, 시도, 구군, 동, 리, 번지 등 |
| 여부코드 | ‘Y’ 와 ‘N’ 또는 ‘0’과 ‘1’의 두가지 인스턴스만 관리한다.  여부코드: ‘이다/아니다’, ‘하다/하지않다’ 의 의미로 사용 (Yes/No, True/False)  유무코드: ‘있음/없음’의 의미로 사용 (Exist/Not Exist)  예) 지급여부, OTP사용유무 |

### 코드종류별 관리방법

[표 11‑2 코드종류별 관리 방법]

|  |  |
| --- | --- |
| 코드 종류 | 코드 관리 방법 |
| 단일코드 | 가장 일반적인 형태의 코드로써 단순히 코드와 코드 정보만 등록하여 관리한다. |
| 목록성코드 | 코드와 코드명 이외의 부가 정보가 추가적으로 관리되어야 함으로 코드를 직접 관리하지 않고 코드 목록을 코드테이블에 관리하고 해당 테이블에 대한 연결 정보만 관리한다. 따라서 별도의 코드를 등록하지 않는다. |
| 여부코드 | 여부코드와 유무코드의 2가지 유형만 관리한다. |

### 목록성 코드

목록성 코드란 코드, 코드명 이외의 추가적인 정보가 존재함으로써 테이블 형태로 관리해야 할 코드를 의미한다.

일반적으로 분류체계를 가지는 코드는 대부분이 목록성 코드이며, 외부에서 정의되는 외부기관코드나 거래를 통해서 입력되는 코드의 경우 목록성 코드로 관리한다.

[표 11‑3 목록성코드 종류]

|  |  |
| --- | --- |
| 목록성코드 종류 | 설명 |
| 분류체계 | 표준산업분류코드(1차분류+2차분류+3차분류)  계정과목코드(BS구분 + 계정코드+일련번호) |
| 외부기관코드 | 타행환응답코드, 국가코드, 통화코드, 지로코드 |
| 기타 | 상품코드, 자금용도코드, 담보코드 |

### 코드 부여방식

코드를 부여하는 방식은 순차채번형, 계층분류형, 코드미등록대상이 있다.

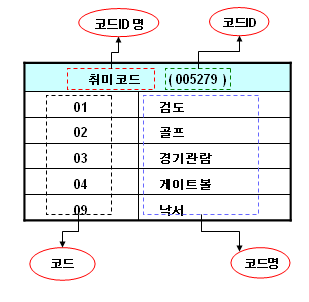
[표 11‑5 코드부여방식]

|  |  |  |
| --- | --- | --- |
| 부여방식 | 부여방법 | 적용예 |
| 순차채번형 | - 일련번호와 같이 의미없이 순차적으로 코드를 부여함.  - 숫자인 경우 가능한 결번이 없도록 정의하며 코드 길이만큼 ‘0’을 앞에 채워서 번호를 부여함.  - 문자인 경우 가능한 알파벹순으로 정의함. | 가족관계코드 → 01: 부, 02: 모, 03: 배우자, 99: 기타 |
| 계층분류형  (복합코드 포함) | - 모든 분야를 1~9의 10진분류 체계로 정의하거나 대중소 분류에 의한 구분이 필요한 경우 적용  - 복합코드는 순차채번형과 계층분류형이 결합된 형태의 코드를 의미함. | 기업규모코드 → 00: 미분류, 10~12: 대기업, 20~26: 중기업, 30~32: 영세·소기업, 40: 개인 사업자, 99: 공공 및 기타  담보구분코드 → A00001: 건물, A00002: 토지, B00001: 예금 등 |
| 코드미등록 대상 | - 목록성 코드(코드를 별도의 테이블로 관리)이거나 코드를 별도로 부여하여 관리하지 않고 정보로 서만 관리하는 경우에는 코드를 등록하지 않음. | 목록성코드 → 계정과목코드(테이블)  정보로서만 관리하는 경우 → HS코드 |

## 코드 명명규칙

### 코드관련 용어 정의

코드표 관련 용어는 아래와 같이 정의한다.



* 코드ID: 코드그룹들을 구분하는 기본키. 일련번호
* 코드ID명: 코드ID에 대응되는 코드그룹의 명칭
* 코드: 값을 추상화 해서 부여하는 기호 또는 숫자
* 코드명: 추상화된 코드에 대한 실제 값

### 코드ID 명명규칙

코드ID명은 코드의 의미가 명확히 표현될 수 있도록 정의한다.

* 코드ID명은 일반적으로 코드 용어와 동일하게 정의하여 사용하나, 코드 용어와 동일한 코드ID명이 다른 코드 인스턴스로 이미 등록되어 사용하는 경우에는 별도의 코드ID명을 정의하여 사용한다. 또한 범용적으로 정의된 통합된 코드ID명의 코드 인스턴스와 동일한 코드를 사용하는 경우에는 그 통합된 코드ID명을 사용할 수 있다.

# 코드 부여 표준화

## 코드 부여원칙

* 코드의 부여는 원칙적으로 문자형 일련번호를 부여한다.
* 자료를 발생 순으로 나열해 가는 경우에 최적이며, 추가도 용이하다.
* 자료가 비교적 많지 않고, 장기간에 걸쳐 변하지 않으리라고 예상되는 경우에 적합하다.
* 코드 Length는 향후 확장성을 고려해서 부여한다.

### 코드 부여원칙

표준코드에 대한 코드 길이는 코드 개수 및 향후 확장성을 고려하여 부여한다.

* 숫자로만 이루어진 코드는 원칙적으로 허용하지 않으며 코드 길이만큼 문자형 숫자를 이용해서‘0’을 채워서 코드를 부여한다.

[표 12‑1 코드부여원칙 예시]

|  |  |  |
| --- | --- | --- |
| 취미코드(2자리) | AS-IS | TO-BE |
| 운동 | 1 | 01 |
| 테니스 | 2 | 02 |
| ~ | ~ | ~ |
| 영화감상 | 10 | 04 |
| 수영 | 11 | 05 |

* 특별한 경우에 ‘기타’ 의 의미로 사용하며 해당 자리의 최대값으로 부여한다. 원칙적으로 모든 코드값은 의미를 반드시 부여하여 관리한다.
* 코드는 ‘1’,‘01’ 로 시작하며 ‘0’,‘00’은 As-Is의 경우 기존의 ‘0’ 값을 허용하며 To-Be의 경우 As-Is의 ‘0’ 값에 대한 To-Be의 매핑이 존재하지 않을 경우(매핑을 할 수 없을 경우) ‘0’ 값을 허용한다.
* 코드 부여는 가능한 연속적으로 부여한다. 단, 보수관계(0: 신규 9: 해지)의 경우를 제외하고 비연속적 인 부여는 원칙적으로 허용하지 않는다.
* 분류코드의 경우도 일련번호로 부여하며 중, 소코드 앞자리에 상위 코드을 포함하지 않도록 한다. 하위코드에 상위코드을 포함하는 경우 코드만으로 상위 분류가 식별되는 장점이 있으나, 분류가 바뀔 경우 하위코드의 변경을 적용하기 어렵다.

### 코드 부여원칙 예외

* 기존 현행코드에서 특별한 의미를 가지는 코드로 사용되었을 경우 그대로 채택한다.

[표 12‑2 코드부여원칙 예외]

|  |  |
| --- | --- |
| 코드 | 코드명 |
| KRW | 원화 |
| USD | 미국 달러 |
| HKD | 홍콩달러 |
| BTC | 비트코인 |

* 목록성 코드의 경우 표준화 대상에서 제외(상품코드, 계정과목코드)
* 외부기관에서 정의된 코드의 경우 표준화 대상에서 제외(은행코드, 지로코드)
* 현재 사용중인 코드 중 향후 당사적 차원에서 표준화가 논의 되어야 할 코드의 경우 표준화를 하지 않는다. 예) 거래코드, 업무코드, 담보코드 등

## 코드표준 세부작업 방안

표준 코드의 도출을 위해서는 현행코드를 분석하여 다음과 같은 원칙을 반영하여 작업한다.

* 코드도메인 분류
* 중복 코드도메인의 제거
* 동일 의미 코드의 통합
* 코드 관계설정
* 코드 표준화
* As-Is To-Be 코드 매핑

### 코드 분류

코드를 유형별로 분류한다.

* ‘여부’, ‘유무’ 도메인 분류
* ‘목록성코드’, ‘복합코드’의 분류
* 코드 유형별 분류(‘구분코드’, ‘종류코드’, ‘분류코드’…… )

### 중복 코드의 제거

동일한 코드 내의 동일한 코드의 중복 제거와 동일한 코드의 코드ID명이 상이한 경우 중복을 제거해서 표준화 한다.

* 코드유사성 코드목록 추출 및 중복여부 Check
* 유사코드목록 작성 및 표준화
* 중복코드 제거

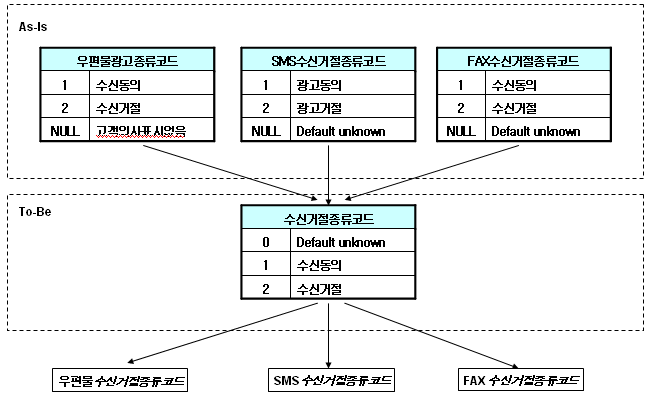
[표 12‑3 중복코드 제거]

|  |  |  |
| --- | --- | --- |
| 코드 | 코드명 | 비고 |
| 00 | 해당사항없음 | ‘00’ 과 ‘99’를 하나로 통합 |
| 02 | 주소불명 |
| ~ | ~ |
| 99 | 해당사항없음 |

### 동일 의미 코드의 통합

다수 시스템 또는 다수개의 코드에서 유사한 도메인이 반복으로 정의되어 사용되고 있을 경우 이를 포괄하여 칭할 수 있는 표준코드명을 정한 후 통합한다. 이를 참조하는 컬럼은 표준코드명을 포함하는 참조 도메인으로 재정의해서 사용한다.

* 코드에 의한 유사코드 목록 작성
* 유사코드에 대한 분할 및 통합



### 코드 표준화

코드 부여원칙에 어긋나는 코드에 대한 표준화를 한다.

* 코드 부여원칙에 따른 코드 표준화
* 코드, 코드명 표준화
* 코드에 의한 동일의미, 중복도메인 분할 및 통합 반복작업

[표 12‑4 코드 표준화 예시]

|  |  |  |  |
| --- | --- | --- | --- |
| 직급코드 | AS-IS | TO-BE | 비고 |
| 사원 | 1 | 01 | 숫자와 문자를 혼용해서 사용할 수 없으며 따라서 2자리 문자형 숫자로 변경해서 표준화 함. |
| 대리 | 2 | 02 |
| 과장 | 3 | 03 |
| ~ | ~ | ~ |
| 상무 | A | 11 |
| 전무 | B | 12 |