

## 핵심정리

### 발생시점(發生時點)에 따른 엔티 분류

- 기본/키엔티(Fundamental Entity, Key Entity)
- 중심엔티(Main Entity)
- 행위엔티(Active Entity)

### 데이터모델링이란

- 정보시스템을 구축하기 위한 데이터 관점의 업무 분석 기법
- 현실세계의 데이터(what)에 대해 약속된 표기법에 의해 표현하는 과정
- 데이터베이스를 구축하기 위한 분석/설계의 과정

### 데이터 모델링 유의점

- 중복(Duplication)
- 비유연성(Inflexibility)
- 비일관성(Inconsistency)

1

모델링은 현실세계에 대해서 표현하는 것으로 이해할 수 있다. 다음 중 모델링의 특징으로 가장 부적절한 것은?

- ① 현실세계를 일정한 형식에 맞추어 표현하는 추상화의 의미를 가질 수 있음
- ② 시스템 구현만을 위해 진행되는 사전단계의 작업으로서 데이터베이스 구축을 위한 사전작업의 의미가 있음
- ③ 복잡한 현실을 제한된 언어나 표기법을 통해 이해하기 쉽게 하는 단순화의 의미를 가지고 있음
- ④ 애매모호함을 배제하고 누구나 이해가 가능하도록 정확하게 현상을 기술하는 정확화의 의미를 가짐

구현 및 업무분석, 업무현상화

2

다음 설명 중 데이터 모델링이 필요한 주요 이유로 가장 부적절한 것은?

- ① 업무정보를 구성하는 기초가 되는 정보들에 대해 일정한 표기법에 의해 표현한다.
- ② 분석된 모델을 가지고 데이터베이스를 생성하여 개발 및 데이터관리에 사용하기 위한 것이다.
- ③ 데이터베이스를 구축하기 위한 용도를 위해 데이터모델링을 수행하고 업무에 대한 설명은 별도의 표기법을 이용한다.
- ④ 데이터모델링 자체로서 업무의 흐름을 설명하고 분석하는 부분에 의미를 가지고 있다.

3

다음 중 데이터모델링을 할 때 유의해야 할 사항으로 가장 부적절한 것은?

- ① 여러 장소의 데이터베이스에 같은 정보를 저장하지 않도록 하여 중복성을 최소화 한다.
- ② 데이터의 정의를 데이터의 사용 프로세스와 분리하여 유연성을 높인다.
- ③ 사용자가 처리하는 프로세스나 장표 등에 따라 매핑이 될 수 있도록 프로그램과 테이블간의 연계성을 높인다.
- ④ 데이터간의 상호 연관관계를 명확하게 정의하여 일관성 있게 데이터가 유지되도록 한다.

연계성이 높아지면 데이터 모델이 업무 변경에 대해 취약해진다.

## 핵심정리

## 개념적 데이터 모델링

추상화 수준이 높고  
업무중심적이고 포괄적인  
수준의 모델링 진행.  
전사적 데이터 모델링.  
EA수립시 많이 이용

## 논리적 데이터 모델링

시스템으로 구축하고자  
하는 업무에 대해 Key,  
속성, 관계 등을 정확하게  
표현, 재사용성이 높음

## 물리적 데이터 모델링

실제로 데이터베이스에  
이식할 수 있도록 성능,  
저장 등 물리적인 성격을  
고려하여 설계

4

다음 중 아래 설명이 의미하는 데이터모델링의 유의점에 해당하는 특성은 무엇인가?

아 래

데이터 모델을 어떻게 설계했느냐에 따라 사소한 업무변화에도 데이터 모델이 수시로 변경됨으로써 유지보수의 어려움을 가중시킬 수 있다. 데이터의 정의를 데이터의 사용 프로세스와 분리함으로써 데이터 모델링은 데이터 혹은 프로세스의 작은 변화가 애플리케이션과 데이터 베이스에 중대한 변화를 일으킬 수 있는 가능성을 줄인다.

- ☒ ㉠ 중복
- ☒ ㉡ 비유연성
- ☒ ㉢ 비일관성
- ☒ ㉣ 일관성

5

다음 중 아래 데이터모델링 개념에 대한 설명에서 ㉠, ㉡에 들어갈 단어로 가장 적절한 것은?

아 래

전사적 데이터 모델링을 수행할 때 많이 하며, 추상화 수준이 높고 업무 중심적이고 포괄적인 수준의 모델링을 진행하는 것을 ㉠ 데이터 모델링 이라고 한다. 이와 달리 실제로 데이터베이스에 이식할 수 있도록 성능, 저장 등의 물리적인 성격을 고려한 데이터 모델링은 ㉡ 데이터모델링 이라고 한다.

- ☒ ㉠ ㉠-개념적, ㉡-물리적
- ☒ ㉡ ㉠-논리적, ㉡-개념적
- ☒ ㉢ ㉠-논리적, ㉡-물리적
- ☒ ㉣ ㉠-개념적, ㉡-논리적

## 핵심정리

### 데이터베이스 스키마

#### 구조 3단계

- 외부스키마 (External Schema)
- 개념스키마 (Conceptual Schema)
- 내부스키마 (Internal Schema)

6

다음 중 ANSI-SPARC에서 정의한 3단계구조(three-level architecture)에서 아래 내용이 설명하는 스키마구조로 가장 적절한 것은?

아래

- 모든 사용자 관점을 통합한 조직 전체 관점의 통합적 표현
- 모든 응용시스템들이나 사용자들이 필요로 하는 데이터를 통합한 조직 전체의 DB를 기술한 것으로 DB에 저장되는 데이터와 그들 간의 관계를 표현하는 스키마

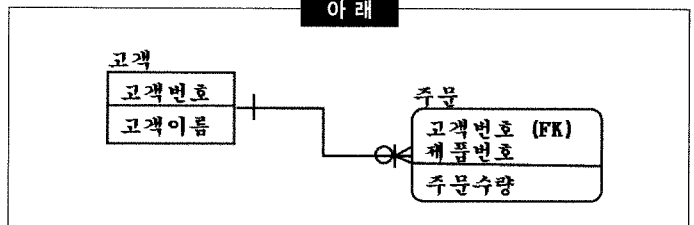
- ① 외부스키마 (External Schema)
- ② 개념스키마 (Conceptual Schema)
- ③ 내부스키마 (Internal Schema)
- ④ 논리스키마 (Logical Schema)

통합관점 - 개념스키마

7

다음 중 고객과 주문의 ERD에 대한 설명으로 가장 부적절한 것은?

아래



- ① 한 명의 고객은 여러 개의 제품을 주문 할 수 있다. 주문은 할 수도 있고 안 할 수도 있다.
- ② 하나의 주문은 반드시 한 명의 고객에 의해 주문된다.
- ③ 주문에 데이터를 입력할 때는 반드시 고객데이터가 존재해야 한다.
- ④ 고객에 데이터를 입력할 때는 주문데이터가 존재하는 고객만을 입력할 수 있다.

## 핵심정리

## ERD 작성 순서

- ① 엔티티를 그린다.
- ② 엔티티를 적절하게 배치한다.
- ③ 엔티티간 관계를 설정한다.
- ④ 관계명을 기술한다.
- ⑤ 관계의 참여도를 기술한다.
- ⑥ 관계의 필수여부를 기술한다.

## 엔티티의 특징

- 반드시 해당 업무에서 필요하고 관리하고자 하는 정보이어야 한다.  
(예. 환자, 토익의 응시횟수, ...)
- 유일한 식별자에 의해 식별이 가능해야 한다.
- 영속적으로 존재하는 인스턴스의 집합이어야 한다.  
(한 개가 아니라 '두 개 이상')
- 엔티티는 업무 프로세스에 의해 이용되어야 한다.
- 엔티티는 반드시 속성이 있어야 한다.
- 엔티티는 다른 엔티티와 최소 한 개 이상의 관계가 있어야 한다.

8

다음 중 ERD에 대한 설명으로 가장 부적절한 것은?

- ① 1976년 피터첸(Peter Chen)에 의해 Entity-Relationship Model(E-R Model)이라는 표기법이 만들어졌다.
- ② 일반적으로 ERD를 작성하는 방법은 엔티티 도출 → 엔티티 배치 → 관계 설정 → 관계명 기술의 흐름으로 작업을 진행한다.
- ③ 관계의 명칭은 관계 표현에 있어서 매우 중요한 부분에 해당한다.
- ④ 가장 중요한 엔티티를 오른쪽 상단에 배치하고 추가 발생하는 엔티티들을 왼쪽 편과 하단에 배치하는 것이 원칙이다.



9

다음 중 아래 시나리오에서 엔티티로 가장 적절한 것은?

아래

S병원원 여러 명의 환자가 존재하고 각 환자에 대한 이름, 주소 등을 관리해야 한다.  
(단, 업무범위와 데이터의 특성은 상기 시나리오에 기술되어 있는 사항만을 근거하여 판단해야 함)

- ① 병원
- ② 환자
- ③ 이름
- ④ 주소

10

다음 중 엔티티의 특징으로 가장 부적절한 것은?

- ① 속성이 없는 엔티티는 있을 수 없다. 엔티티는 반드시 속성을 가져야 한다.
- ② 엔티티는 다른 엔티티와 관계가 있을 수 밖에 없다. 단, 통계성 엔티티나, 코드성 엔티티의 경우 관계를 생략할 수 있다.
- ③ 객체지향의 디자인패턴에는 싱글턴패턴이 있어 하나의 인스턴스를 가지는 클래스가 존재한다. 이와 유사하게 엔티티는 한 개의 인스턴스를 가지는 것만으로도 충분한 의미를 부여할 수 있다.
- ④ 데이터로서 존재하지만 업무에서 필요로 하지 않으면 해당 업무의 엔티티로 성립될 수 없다.

## 엔터티, 인스턴스, 속성, 속성값의 관계

- 한 개의 엔터티는 두 개 이상의 인스턴스의 집합이어야 한다.
- 한 개의 엔터티는 두 개 이상의 속성을 갖는다.
- 한 개의 속성은 한 개의 속성값을 갖는다.

11

다음 중 엔터티의 일반적인 특징으로 가장 부적절한 것은?

- ☒ ㉠ 다른 엔티티와의 관계를 가지지 않는다.
- ☒ ㉡ 유일한 식별자에 의해 식별이 가능해야 한다.
- ☒ ㉢ 엔티티는 업무 프로세스에 의해 이용되어야 한다.
- ☒ ㉣ 엔티티는 반드시 속성을 포함해야 한다.

12

다음 중 다른 엔터티로부터 주식별지를 상속받지 않고 자신의 고유한 주식별  
자를 가지며 사원, 부서, 고객, 상품, 자재 등이 예가 될 수 있는 엔터티로  
가장 적절한 것은?

- ☒ ① 기본 엔터티(키엔터티)  
☒ ② 중심 엔터티(메인엔터티)  
☒ ③ 행위 엔터티  
☒ ④ 개념 엔터티

13

다음 중 엔터티의 이름을 부여하는 방법으로서 가장 부적절한 것은?

- ① 가능하면 약어를 사용하여 엔터티의 이름을 간결하고 명확하게 한다.
- ② 현업의 업무 용어를 사용하여 업무상의 의미를 분명하게 한다.
- ③ 모든 엔터티에서 유일한 이름이 부여되어야 한다.
- ④ 엔터티가 생성되는 의미대로 자연스럽게 부여하도록 한다.

14

업무에서 필요로 하는 인스턴스에서 관리하고자 하는 의미상 더 이상 분리되지 않는 최소의 데이터 단위를 무엇이라 하는가?

[ ]

15

다음 중 속성에 대한 설명으로 가장 부적절한 것은?

- ① 엔터티에 대한 자세하고 구체적인 정보를 나타낸다.
- ② 하나의 엔터티는 두 개 이상의 속성을 갖는다.
- ③ 하나의 인스턴스에서 각각의 속성은 하나 이상의 속성값을 가질 수 있다.
- ④ 속성도 집합이다.

## 핵심정리

## 속성의 특성에 따른 분류

- 기본속성
- 설계속성
- 파생속성

각 속성은 가질 수 있는 값의 범위가 있는데 이를 그 속성의 도메인(Domain)이라하며, 엔터티 내에서 속성에 대한 데이터타입과 크기 그리고 제약사항을 지정하는 것이다.

16

다음 중 아래와 같은 사례에서 속성에 대한 설명으로 가장 부적절한 것은?

## 아 래

우리은행은 예금분류(일반예금, 특별예금 등)의 원금, 예치기간, 이자율을 관리할 필요가 있다. 또한 원금에 대한 이자율을 적용하여 계산된 이자에 대해서도 속성으로 관리하고자 한다. 예를 들어 원금이 1000원이고 예치기간이 5개월이며 이자율이 5.0%라는 속성을 관리하고 계산된 이자도 관리한다. 일반예금이나 특별예금 등에 대해서는 코드를 부여(예, 01-일반예금, 02-특별예금 등)하여 관리한다.

- ① 일반예금은 코드 엔터티를 별도로 구분하고 값에는 코드값만 포함한다.  
 ② 원금, 예치기간은 기본(BASIC)속성이다.  
 ③ 이자와 이자율은 파생(DERIVED)속성이다.  
 ④ 예금분류는 설계(DESIGNED)속성이다.

이자는 계산된 값으로 파생속성  
 이자율은 현재 가지고 있는 기본속성

17

다음 중 데이터를 조회할 때 빠른 성능을 낼 수 있도록 하기 위해 원래 속성의 값을 계산하여 저장할 수 있도록 만든 속성으로 가장 적절한 것은?

- ① 파생속성(Derived Attribute)  
 ② 기본속성(Basic Attribute)  
 ③ 설계속성(Designed Attribute)  
 ④ PK속성(Derived Attribute)

18

다음 중 아래 설명이 나타내는 데이터모델의 개념으로 가장 적절한 것은?

## 아 래

주문이라는 엔터티가 있을 때 단가라는 속성 값의 범위는 100에서 10,000 사이의 실수 값이며 제품명이라는 속성은 길이가 20자리 이내의 문자열로 정의할 수 있다.

- ① 시스템카탈로그(System Catalog)  
 ② 용어사전(Word Dictionary)  
 ③ 속성사전(Attribute Dictionary)  
 ④ 도메인(Domain)

## 핵심정리

### 속성의 명칭 부여

- 해당업무에서 사용하는 이름을 부여 한다.
- 서술식 속성명은 사용하지 않는다.
- 약어사용은 가급적 제한한다.
- 전체 데이터모델에서 유일성 확보하는 것이 좋다.

ERD에서는 존재적 관계와 행위에 의한 관계를 구분하지 않지만 클래스 다이어그램에서는 이것을 구분하여 연관관계와 의존관계로 표현한다.

19

다음 중 데이터모델링을 할 때 속성의 명칭을 부여하는 방법으로 가장 부적절한 것은?

- ① 속성의 이름에 약어를 사용할 경우 그 의미를 명확하게 이해할 수 없고 혼돈을 초래하여 커뮤니케이션의 혼란을 야기할 수 있으므로 지나친 약어 사용은 가급적 제한하도록 한다.
- ② 속성의 이름에는 서술식 용어는 사용하지 않도록 한다.
- ③ 직원 엔터티의 이름, 고객 엔터티의 이름과 같이 각 엔터티별로 동일한 속성명을 사용하여 데이터모델의 일관성을 가져가는 것이 좋다.
- ④ 데이터모델링 대상에서 사용하는 용어도 있고 외부에서 사용하는 용어도 있어 중복이 있을 때, 가급적 해당 업무에서 자주 사용하는 이름을 이용하도록 한다.

20

다음 중 데이터모델링의 관계에 대한 설명으로 가장 부적절한 것을 2개 고르시오.

- ① 관계는 존재에 의한 관계와 행위에 의한 관계로 구분될 수 있으나 ERD에서는 관계를 연결할 때, 존재와 행위를 구분하지 않고 단일화된 표기법을 사용한다.
- ② UML(Unified Modeling Language)에는 클래스 다이어그램의 관계 중 연관관계(Association)와 의존관계(Dependency)가 있고 이것은 실선과 점선의 표기법으로 다르게 표현이 된다.
- ③ 관계는 존재에 의한 관계와 행위에 의한 관계로 구분될 수 있고 ERD에서는 관계를 연결할 때, 존재와 행위를 구분하여 실선과 점선의 표기법으로 다르게 표현한다.
- ④ UML(Unified Modeling Language)에는 클래스 다이어그램의 관계 중 연관관계(Association)와 의존관계(Dependency)가 있고 있으나 구분하지 않고 단일화된 표기법을 사용한다.

21

다음 중 관계에 대한 설명으로 가장 부적절한 것은?

- ① 관계는 존재적 관계와 행위에 의한 관계로 나누어볼 수 있다.
- ② 관계의 표기법은 관계명, 관계차수, 식별성의 3가지 개념을 사용한다.
- ③ 부서와 사원 엔터티 간의 '소속' 관계는 존재적 관계의 사례이다.
- ④ 주문과 배송 엔터티 간의 '배송근거' 관계는 행위에 의한 관계의 사례이다.

## 핵심정리

## 관계의 표기법

- 관계명(Membership) :  
관계의 이름
- 관계차수(Cardinality) :  
1:1, 1:M, M:N
- 관계선택사양  
(Optionality)  
: 필수관계, 선택관계

## 관계 읽기

- 기준(Source) 엔터티를  
한 개(One) 또는  
각(Each)으로 읽는다.
- 대상(Target)  
엔터티의 관계참여도  
즉 개수(하나, 하나  
이상)를 읽는다.
- 관계선택사양과  
관계명을 읽는다.

22

다음 중 엔터티간의 관계에서 1:1, 1:M과 같이 관계의 기수성을 나타내는 것으로 가장 적절한 것은?

- ☒ ① 관계명(Relationship Membership)
- ☒ ② 관계차수(Relationship Degree/Cardinality)
- ☒ ③ 관계선택사양(Relationship Optionality)
- ☒ ④ 관계정의(Relationship Definition)

23

다음 중 두 개의 엔터티 사이에 정의한 관계를 체크하는 사항으로 가장 부적절한 것은?

- ☒ ① 두 개의 엔터티 사이에 관심 있는 연관규칙이 존재하는가?
- ☒ ② 두 개의 엔터티 사이에 정보의 조합이 발생하는가? **왜**
- ☒ ③ 업무기술서, 장표에 관계연결을 가능하게 하는 명사(Noun)가 있는가?
- ☒ ④ 업무기술서, 장표에 관계연결에 대한 규칙이 서술되어 있는가?

24

다음 중 두 개의 엔터티 사이에서 관계를 도출 할 때 체크 할 사항을 모두 고른 것은?

## 아 래

가. 두 개의 엔터티 사이에 관심있는 연관규칙이 존재하는가?  
 나. 두 개의 엔터티 사이에 정보의 조합이 발생하는가?  
 다. 업무기술서, 장표에 관계연결에 대한 규칙이 서술되어 있는가?  
 라. 업무기술서, 장표에 관계연결을 가능하게 하는 동사(Verb)가 있는가?

- ☒ ① 가, 다, 라
- ☒ ② 가, 다, 라
- ☒ ③ 가, 나, 다
- ☒ ④ 가, 나, 다, 라



## 핵심정리

### 식별자의 종류

- 엔터티 내에서 대표성을 가지는가에 따라 주식별자 (Primary Identifier)와 보조식별자(Alternate Identifier)로 구분
- 엔터티 내에서 스스로 생성되었는지 여부에 따라 내부식별자와 외부식별자(Foreign Identifier)로 구분
- 단일 속성으로 식별이 되는가에 따라 단일식별자 (Single Identifier)와 복합식별자(Composit Identifier)로 구분
- 원래 업무적으로 의미가 있던 식별자 속성을 대체하여 일련번호와 같이 새롭게 만든 식별자를 구분하기 위해 본질식별자와 인조식별자로 구분

25

다음 중 아래에서 주식별자를 지정할 때 고려해야 할 사항을 묶은 것으로 가장 적절한 것은?

아 래

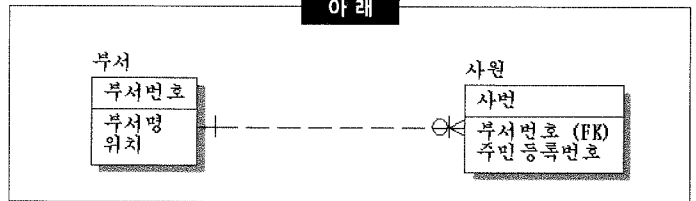
- ㉠ 주식별자에 의해 엔터티 내의 모든 인스턴스들이 유일하게 구분되어야 한다.
- ㉡ 주식별자를 구성하는 속성의 수는 유일성을 만족하는 최소의 수가 되어야 한다.
- ㉢ 지정된 주식별자의 값은 자주 변하지 않는 것이어야 한다.
- ㉣ 주식별자가 지정이 되면 반드시 값이 들어와야 한다.

- ㉠ 가, 나, 다
- ㉡ 가, 나, 라
- ㉢ 나, 다, 라
- ㉣ 가, 나, 다, 라

26

다음 중 사원엔터티에서 식별자의 특성에 해당하지 않는 것은 무엇인가?

아 래



- ㉠ 주식별자
- ㉡ 단일식별자
- ㉢ 내부식별자
- ㉣ 인조식별자

27

다음 중 식별자로 가장 부적절한 것은?

- ㉠ 사원  

사원번호: NUMBER(10)
주민등록번호: NUMBER(13)
- ㉡ 사원  

이름: VARCHAR2(20)
사원번호: NUMBER(10)
- ㉢ 사원  

주민번호: NUMBER(13)
사원번호: NUMBER(10)
- ㉣ 사원  

일련번호: VARCHAR2(10)
주민등록번호: CHAR(18)
사원번호: NUMBER(10)

## 핵심정리

## 주식별자의 특징

- 유일성 : 주식별자에 의해 엔터티내에 모든 인스턴스들을 유일하게 구분함
- 최소성 : 주식별자를 구성하는 속성의 수는 유일성을 만족하는 최소의 수가 되어야 함
- 불변성 : 주식별자가 한 번 특정 엔터티에 지정되면 그 식별자의 값은 변하지 않아야 함
- 존재성 : 주식별자가 지정되면 반드시 데이터 값이 존재 (Null 안됨)

28

다음 중 아래에서 엔터티 내에 주식별자를 도출하는 기준을 묶은 것으로 가장 적절한 것은?

아 래

- 가. 해당 업무에서 자주 이용되는 속성을 주식별자로 지정한다.  
 나. 명칭, 내역 등과 같이 이름으로 기술되는 것들을 주식별자로 지정한다.  
 다. 복합으로 주식별자를 구성할 경우 너무 많은 속성을 포함하지 않도록 한다.  
 라. 자주 수정되는 속성을 주식별자로 지정한다.

- ① 가, 나  
 ② 가, 다  
 ③ 다, 라  
 ④ 나, 라

29

프로젝트를 전개할 때는 식별자관계와 비식별자관계를 선택하여 연결해야 하는 높은 수준의 데이터모델링 기술이 필요하다. 다음 중 비식별자관계를 선택하는 기준으로 가장 부적절한 것은?

- ① 관계의 강약을 분석하여 상호간에 연관성이 약할 경우 비식별자관계를 고려한다.  
 ② 자식테이블에서 독립적인 Primary Key의 구조를 가지기 원할 때 비식별자관계를 고려한다.  
 ③ 모든 관계가 식별자 관계로 연결되면 SQL Where절에서 비교하는 항목이 증가되어 조인에 참여하는 테이블에 따라 SQL문장이 길어져 SQL문의 복잡성이 증가되는 것을 방지하기 위해 비식별자관계를 고려한다.  
 ④ 부모엔터티의 주식별자를 자식엔터티에서 받아 손자엔터티까지 계속 흘러보내기 위해 비식별자관계를 고려한다.

## 식별자와 비식별자관계 비교

항목	식별자관계	비식별자관계
목적	강한 연결관계 표현	약한 연결관계 표현
자식 주식별자 영향	자식 주식별자의 구성에 포함됨	자식 일반 속성에 포함됨
표기법	실선 표현	점선 표현
연결 고려사항	<ul style="list-style-type: none"> <li>- 반드시 부모엔터티 종속</li> <li>- 자식 주식별자구성에 부모 주식별자포함 필요</li> <li>- 상속받은 주식별자속성을 타 엔터티에 이전 필요</li> </ul>	<ul style="list-style-type: none"> <li>- 약한 종속관계</li> <li>- 자식 주식별자구성을 독립적으로 구성</li> <li>- 자식 주식별자구성에 부모 주식별자 부분 필요</li> <li>- 상속받은 주식별자속성을 타 엔터티에 차단 필요</li> <li>- 부모쪽의 관계참여가 선택관계</li> </ul>

30

다음 중 비식별자 관계로 연결하는 것을 고려해야 하는 경우로 가장 부적절한 것은?

- ① 부모엔터티에 참조값이 없어도 자식엔터티의 인스턴스가 생성될 수 있는 경우
- ② 부모엔터티의 인스턴스가 자식 엔터티와 같이 소멸되는 경우
- ③ 여러 개의 엔터티를 하나로 통합하면서 각각의 엔터티가 갖고 있던 여러 개의 개별 관계가 통합되는 경우
- ④ 자식쪽 엔터티의 주식별자를 부모엔터티와는 별도로 생성하는 것이 더 유리하다고 판단하는 경우

#### 식별자의 분류체계

분류	식별자	설명
대표성 여부	주식별자	엔터티 내에서 각 어커런스를 구분할 수 있는 구분자이며, 타 엔터티와 참조관계를 연결할 수 있는 식별자
	보조식별자	엔터티 내에서 각 어커런스를 구분할 수 있는 구분자이나 대표성을 가지지 못해 참조관계 연결을 못함
스스로 생성여부	내부식별자	엔터티 내부에서 스스로 만들어지는 식별자
	외부식별자	타 엔터티와의 관계를 통해 타 엔터티로부터 받아오는 식별자
속성의 수	단일식별자	하나의 속성으로 구성된 식별자
	복합식별자	둘 이상의 속성으로 구성된 식별자
대체 여부	본질식별자	업무에 의해 만들어지는 식별자
	인조식별자	업무적으로 만들어지지는 않지만 원조식별자가 복잡한 구성을 가지고 있기 때문에 인위적으로 만든 식별자