



## 2020년 2회 정보처리기사 실기 시험 100% 합격전략집

1  
일차

2  
일차

3  
일차

4  
일차

5  
일차

6  
일차

7  
일차

8  
일차

9  
일차

10  
일차

11  
일차

12  
일차

13  
일차

14  
일차

15  
일차

16  
일차

17  
일차

18  
일차

19  
일차

20  
일차

### 3장 데이터 입 · 출력 구현

핵심 042 데이터 모델

핵심 043 데이터 모델의 종류

핵심 044 데이터 모델에 표시할 요소

핵심 045 이상(Anomaly)

핵심 046 함수적 종속(Functional  
Dependency)

핵심 047 정규화(Normalization)

핵심 048 논리 데이터 모델의 물리  
데이터 모델로 변환

핵심 049 반정규화(Denormalization)

핵심 050 인덱스(Index)

핵심 051 뷰(View)

핵심 052 트랜잭션(Transaction)

핵심 053 클러스터(Cluster)

핵심 054 파티션(Partition)

핵심 055 데이터베이스 용량 설계



## 2020년 2회 정보처리기사 실기 대비용 핵심요약

### 3장 | 데이터 입 · 출력 구현

#### [핵심 042] 데이터 모델

- 현실 세계의 정보들을 컴퓨터에 표현하기 위해서 단순화, 추상화하여 체계적으로 표현한 개념적 모형이다.
- 데이터 모델의 구성 요소
  - 개체(Entity) : 데이터베이스에 표현하려는 것으로, 사람이 생각하는 개념이나 정보 단위 같은 현실 세계의 대상체
  - 속성(Attribute) : 데이터의 가장 작은 논리적 단위로서 파일 구조상의 데이터 항목 또는 데이터 필드에 해당함
  - 관계(Relationship) : 개체 간의 관계 또는 속성 간의 논리적인 연결을 의미함

1. 다음은 데이터 모델의 구성 요소들에 대한 설명이다. 괄호 (①~③)에 들어갈 알맞은 구성 요소를 쓰시오.

- ( ① )은 데이터의 가장 작은 논리적 단위로서 파일 구조상의 데이터 항목 또는 데이터 필드에 해당한다.
- ( ② )는 개체 간의 관계 또는 속성 간의 논리적인 연결을 의미한다.
- ( ③ )는 데이터베이스에 표현하려는 것으로, 사람이 생각하는 개념이나 정보 단위 같은 현실 세계의 대상체이다.

답 :

- ① :  
② :  
③ :

정답 1. ① 속성(Attribute) ② 관계(Relationship) ③ 개체(Entity)

#### [핵심 043] 데이터 모델의 종류

- 개념적 데이터 모델 : 현실 세계에 대한 인간의 이해를 돕기 위해 현실 세계에 대한 인식을 추상적 개념으로 표현하는 과정
- 논리적 데이터 모델 : 개념적 모델링 과정에서 얻은 개념적 구조를 컴퓨터가 이해하고 처리할 수 있는 컴퓨터 세계의 환경에 맞도록 변환하는 과정

- 물리적 데이터 모델 : 실제 컴퓨터에 데이터가 저장되는 방법을 정의하는 물리 데이터베이스 설계 과정

1. 데이터 모델의 종류 중 개념적 모델링 과정에서 얻은 개념적 구조를 컴퓨터가 이해하고 처리할 수 있는 컴퓨터 세계의 환경에 맞도록 변환하는 과정을 의미하는 것은 무엇인지 쓰시오.

답 :

정답 1. 논리적 데이터 모델

#### [핵심 044] 데이터 모델에 표시할 요소

- 구조(Structure) : 논리적으로 표현된 개체 타입들 간의 관계로서 데이터 구조 및 정적 성질을 표현함
- 연산(Operation) : 데이터베이스에 저장된 실제 데이터를 처리하는 작업에 대한 명세로서 데이터베이스를 조작하는 기본 도구
- 제약 조건(Constraint) : 데이터베이스에 저장될 수 있는 실제 데이터의 논리적인 제약 조건

1. 데이터 모델은 현실 세계의 정보들을 컴퓨터에 표현하기 위해서 단순화, 추상화하여 체계적으로 표현한 개념적 모형이다. 데이터 모델에 표시할 요소 세 가지를 쓰시오.

답 :

정답 1. 구조(Structure), 연산(Operation), 제약 조건(Constraint)

#### [핵심 045] 이상(Anomaly)

- 테이블에서 일부 속성들의 중복으로 인해 데이터의 중복(Redundancy)이 발생하고, 이 중복으로 인해 테이블 조작 시 문제가 발생하는 현상을 의미한다.
- 이상의 종류
  - 삽입 이상(Insertion Anomaly) : 테이블에 데이터를 삽입할 때 의도와는 상관없이 원하지 않은 값들로 인해 삽입할 수 없게 되는 현상
  - 삭제 이상(Deletion Anomaly) : 테이블에서 한 튜플을 삭제할 때 의도와는 상관없는 값들도 함께 삭제되는,

즉 연쇄 삭제가 발생하는 현상

- 갱신 이상(Update Anomaly) : 테이블에서 튜플에 있는 속성 값을 갱신할 때 일부 튜플의 정보만 갱신되어 정보에 불일치성(Inconsistency)이 생기는 현상

1. 다음은 <출고> 테이블을 분석한 결과이다. <분석 결과>에 해당하는 이상(Anomaly)의 명칭을 쓰시오.

<출고>

출고번호	제품번호	출고가	출고량
1001	11	15,000	150
1002	22	16,000	200
1003	33	12,000	100
1004	44	18,000	150

<분석 결과>

- 어느 한 제품에 대한 출고를 삭제하면 제품 정보도 함께 삭제된다. 예를 들어 출고번호 1002를 삭제하면 제품번호 22에 대한 제품 정보까지 모두 삭제된다.
- 제품번호 33의 출가를 변경하려면 그 제품번호가 포함된 모든 출고들의 출고가들을 변경해야 한다.
- 제품번호가 55인 새로운 출고 내용을 입력하기 위해서는 반드시 출고번호가 있어야 한다.

답

- 1 :
- 2 :
- 3 :

2. 이상(Anomaly) 현상이 발생하는 원인을 설명하시오.

답 :

**정답** 1. ① 삭제 이상(Deletion Anomaly) ② 갱신 이상(Update Anomaly)  
③ 삽입 이상(Insertion Anomaly)

2. 이상은 테이블에서 일부 속성들의 종속으로 인해 데이터의 중복이 발생하기 때문에 발생한다.

## [핵심 046] 함수적 종속(Functional Dependency)

- 어떤 테이블 R에서 X와 Y를 각각 R의 속성 집합의 부분 집합이라 할 때 속성 X의 값 각각에 대해 시간에 관계없이 항상 속성 Y의 값이 오직 하나만 연관되어 있을 때 Y는 X에 함수적 종속 또는 X가 Y를 함수적으로 결정한다고 하고,  $X \rightarrow Y$ 로 표기한다.
- $X \rightarrow Y$ 의 관계를 갖는 속성 X와 Y에서 X를 결정자(Determinant)라 하고, Y를 종속자(Dependent)라고 한다.
- 완전 함수적 종속 : 어떤 테이블 R에서 속성 Y가 다른 속성 집합 X 전체에 대해 함수적 종속이면서 속성 집합 X의 어떠한 진부분 집합 Z(즉,  $Z \subset X$ )에도 함수적 종속이 아닐 때 속성 Y는 속성 집합 X에 완전 함수적 종속이라고 함
- 부분 함수적 종속 : 어떤 테이블 R에서 속성 Y가 다른 속성 집합 X 전체에 대해 함수적 종속이면서 속성 집합 X의 임의의 진부분 집합에 대해 함수적 종속일 때, 속성 Y는 속성 집합 X에 부분 함수적 종속이라고 함

1. 데이터베이스와 관련된 다음 설명에서 괄호(①, ②)에 들어갈 알맞은 용어를 쓰시오.

어떤 테이블 R에서 X와 Y를 각각 R의 속성 집합의 부분 집합이라 할 때 속성 X의 값 각각에 대해 시간에 관계없이 항상 속성 Y의 값이 오직 하나만 연관되어 있을 때 Y는 X에 ( ① )이라고 하고, ( ② )로 표기한다.

답

- ① :
- ② :

2. 데이터베이스와 관련된 다음 설명에서 괄호(①, ②)에 들어갈 알맞은 함수적 종속의 종류를 쓰시오.

( ① )이라는 말은 어떤 속성이 기본키에 대해 완전히 종속적일 때를 말한다. 예를 들어 <수강> 테이블은 (학번, 과목번호)가 기본키이고, 성적은 학번과 과목번호에 의해서만 결정된다면 성적은 기본키(학번, 과목번호)에 ( ① )이라고 한다. 반면 학년은 과목번호에 관계없이 학번이 같으면 항상 같은 학년이 온다면, 즉 기본키의 일부인 학번에 의해서 학년이 결정된다면 학년은 ( ② )이라고 한다.

답

- ① :  
② :

- 정답** 1. ① 함수적 종속 ②  $X \rightarrow Y$   
2. ① 완전 함수적 종속 ② 부분 함수적 종속



### [핵심 047] 정규화(Normalization)

- 테이블의 속성들이 상호 종속적인 관계를 갖는 특성을 이용하여 테이블을 무손실 분해하는 과정이다.
- 정규화 과정



#### 정규화 단계 암기 요령

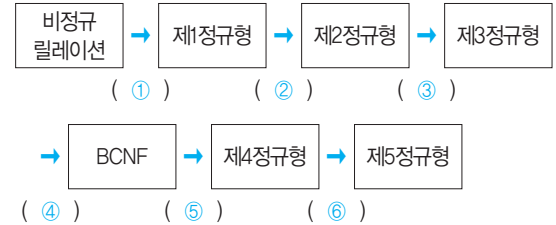
두부를 좋아하는 정규화가 두부가  
게에 가서 가게에 있는 두부를 다  
달라고 말하니 주인이 깜짝 놀라며  
말했다.

두부이걸다줘?  $\approx$  도부이걸다조

도메인이 원자값  
부분적 함수 종속 제거  
이행적 함수 종속 제거  
결정자이면서 후보키가 아닌 것 제거  
다치 종속  
조인 종속성 이용

※ 이행적 함수적 종속 :  $A \rightarrow B$ 이고  $B \rightarrow C$ 일 때  $A \rightarrow C$   
를 만족하는 관계를 이행적 함수적 종속이라고 함

1. 다음은 정규화 과정을 나열한 것이다. 괄호(①~⑥)에 각 정규화 과정별로 필요한 작업을 쓰시오.



답

- ① :  
② :  
③ :  
④ :  
⑤ :  
⑥ :

2020년 1, 2회 기사 필기

2. 다음은 이행적 함수 종속에 대한 설명이다. 괄호에 들어갈 알맞은 기호를 쓰시오.

이행적 함수 종속은  $A \rightarrow B$ 이고  $B \rightarrow C$ 일 때 ( )를 만족하는 관계를 의미한다.

답 :

- 정답** 1. ① 도메인을 원자 값만으로 구성 ② 부분적 함수적 종속 제거  
③ 이행적 함수적 종속 제거 ④ 결정자이면서 후보키가 아닌 것 제거  
⑤ 다치 종속 ⑥ 조인 종속성 이용

2.  $A \rightarrow C$

### [핵심 048] 논리 데이터 모델의 물리 데이터 모델로 변환

- 엔티티(Entity)를 테이블로 변환 : 논리 데이터 모델에서 정의된 엔티티를 물리 데이터 모델의 테이블로 변환
- 슈퍼타입 기준 테이블 변환 : 서브타입을 슈퍼타입에 통합하여 하나의 테이블로 변환
- 서브타입 기준 테이블 변환 : 슈퍼타입 속성들을 각각의 서브타입에 추가하여 서브타입들을 개별적인 테이블로 변환



- 개별타입 기준 테이블 변환 : 슈퍼타입과 서브타입들을 각각의 개별적인 테이블로 변환
- 속성을 컬럼으로 변환 : 논리 데이터 모델에서 정의한 속성을 물리 데이터 모델의 컬럼으로 변환
- 관계를 외래키로 변환 : 논리 데이터 모델에서 정의된 관계는 기본키와 이를 참조하는 외래키로 변환

1. 다음 설명의 괄호(①, ②)에 들어갈 알맞은 용어를 쓰시오.

- 개별타입 기준의 테이블 변환은 ( ① )과 ( ② )들을 각각의 개별적인 테이블로 변환하는 것으로, 두 테이블들 사이에는 각각 1:1 관계가 형성된다.
- ( ① ) 기준의 테이블 변환은 ( ② )을 ( ① )에 통합하여 하나의 테이블로 만드는 것이고, ( ② ) 기준의 테이블 변환은 ( ① ) 속성들을 각각의 ( ② )에 추가하여 ( ② )들을 개별적인 테이블로 만드는 것이다.

답

- ① :  
② :

정답 1. ① 슈퍼타입 ② 서브타입

### [핵심 049] 반정규화(Denormalization)

- 시스템의 성능 향상, 개발 및 운영의 편의성 등을 위해 정규화된 데이터 모델을 통합, 중복, 분리하는 과정으로, 의도적으로 정규화 원칙을 위배하는 행위이다.
- 반정규화 방법

테이블 통합	두 개의 테이블이 조인(Join)되는 경우가 많아 하나의 테이블로 합쳐 사용하는 것이 성능 향상에 도움이 될 경우 수행함
테이블 분할	<ul style="list-style-type: none"> <li>• 수평 분할(Horizontal Partitioning) : 레코드(Record)를 기준으로 테이블을 분할하는 것으로, 레코드별로 사용 빈도의 차이가 큰 경우 사용 빈도에 따라 테이블을 분할함</li> <li>• 수직 분할(Vertical Partitioning) : 하나의 테이블에 속성이 너무 많을 경우 속성을 기준으로 테이블을 분할하는 것</li> </ul>

중복 테이블 추가	<ul style="list-style-type: none"> <li>• 여러 테이블에서 데이터를 추출해서 사용해야 하거나 다른 서버에 저장된 테이블을 이용해야 하는 경우 중복 테이블을 추가하여 작업의 효율성을 향상시킬 수 있음</li> <li>• 추가 방법 : 집계 테이블의 추가, 진행 테이블의 추가, 특정 부분만을 포함하는 테이블의 추가</li> </ul>
중복 속성 추가	조인해서 데이터를 처리할 때 데이터를 조회하는 경로를 단축하기 위해 자주 사용하는 속성을 하나 더 추가하는 것

2020년 1회 기사 실기

1. 비정규화(Denormalization)의 개념을 설명하시오.

답 :

2020년 1, 2회 기사 필기

2. 반정규화(Denormalization) 유형 중 중복 테이블을 추가하는 방법 3가지를 쓰시오.

답 :

정답 1. 비정규화는 시스템의 성능 향상, 개발 및 운영의 편의성 등을 위해 정규화된 데이터 모델을 통합, 중복, 분리하는 과정이다.

2. 집계 테이블의 추가, 진행 테이블의 추가, 특정 부분만을 포함하는 테이블의 추가

### [핵심 050] 인덱스(Index)

- 데이터 레코드를 빠르게 접근하기 위해 <키 값, 포인터> 쌍으로 구성되는 데이터 구조이다.
- 인덱스 키의 순서에 따라 데이터가 정렬되어 저장되는 방식인 클러스터드 인덱스(Clustered Index)와 인덱스의 키 값만 정렬되어 있을 뿐 실제 데이터는 정렬되지 않는 방식인 닌클러스터드 인덱스(Non-Clustered Index)가 있다.
- 인덱스는 인덱스를 구성하는 구조나 특징에 따라 다음과 같이 구분한다.

트리 기반 인덱스	인덱스를 저장하는 블록들이 트리 구조를 이루고 있는 것으로, 상용 DBMS에서는 트리 구조 기반의 B+ 트리 인덱스를 주로 활용함
비트맵 인덱스	인덱스 컬럼의 데이터를 Bit 값인 0 또는 1로 변환하여 인덱스 키로 사용하는 방법



함수 기반 인덱스	컬럼의 값 대신 컬럼에 특정 함수(Function)나 수식(Expression)을 적용하여 산출된 값을 사용하는 것으로, B+ 트리 인덱스 또는 비트맵 인덱스를 생성하여 사용함
비트맵 조인 인덱스	다수의 조인된 객체로 구성된 인덱스로, 단일 객체로 구성된 일반적인 인덱스와 액세스 방법이 다름
도메인 인덱스	개발자가 필요한 인덱스를 직접 만들어 사용하는 것으로, 확장형 인덱스(Extensible Index)라고도 함

1. 인덱스(Index)는 인덱스를 구성하는 특징에 따라 다음과 같이 구분한다. 다음 괄호(①~③)에 들어갈 알맞은 인덱스 방법을 쓰시오.

함수 기반 인덱스	컬럼의 값 대신 컬럼에 특정 Function이나 Expression을 적용하여 산출된 값을 사용하는 것이다.
( ① )	다수의 조인된 객체로 구성된 인덱스로, 단일 객체로 구성된 일반적인 인덱스와 액세스 방법이 다르다.
( ② )	인덱스 컬럼의 데이터를 Bit 값인 0 또는 1로 변환하여 인덱스 키로 사용하는 방법이다.
트리 기반 인덱스	인덱스를 저장하는 블록들이 트리 구조를 이루고 있는 것이다.
( ③ )	개발자가 필요한 인덱스를 직접 만들어 사용하는 것이다.

답

- ① :  
② :  
③ :

2. 다음 괄호에 공통적으로 들어갈 데이터베이스 관련 용어를 쓰시오.

- ( )는 데이터 레코드를 빠르게 접근하기 위해 <키 값, 포인터> 쌍으로 구성되는 데이터 구조이다.
- 클러스터드 ( ) : ( ) 키의 순서에 따라 데이터가 정렬되어 저장되는 방식이다.
- 년클러스터드 ( ) : ( )의 키 값만 정렬되어 있을 뿐 실제 데이터는 정렬되지 않는 방식이다.

답 :

정답 1. ① 비트맵 조인 인덱스 ② 비트맵 인덱스 ③ 도메인 인덱스  
2. 인덱스(Index)

### [핵심 051] 뷰(View)

- 사용자에게 접근이 허용된 자료만을 제한적으로 보여주기 위해 하나 이상의 기본 테이블로부터 유도된, 이름을 가지는 가상 테이블이다.
- 저장장치 내에 물리적으로 존재하지 않지만, 사용자에게는 있는 것처럼 간주된다.
- CREATE문으로 정의하고, DROP문으로 제거한다.
- 데이터의 논리적 독립성을 제공한다.

2020년 1, 2회 기사 필기

1. 다음이 설명하는 데이터베이스 관련 용어를 쓰시오.

- 하나 이상의 기본 테이블로부터 유도된, 이름을 가지는 가상 테이블이다.
- 저장장치 내에 물리적으로 존재하지 않는다.
- CREATE문을 사용하여 정의한다.
- 제거할 때는 DROP문을 사용한다.
- 데이터의 논리적 독립성을 제공한다.

답 :

정답 1. 뷰(View)

### [핵심 052] 트랜잭션(Transaction)

- 데이터베이스의 상태를 변환시키는 하나의 논리적 기능을 수행하기 위한 작업의 단위 또는 한꺼번에 모두 수행되어야 할 일련의 연산들을 의미한다.
- 특징

Atomicity (원자성)	트랜잭션의 연산은 데이터베이스에 모두 반영되도록 완료(Commit)되든지 아니면 전혀 반영되지 않도록 복구(Rollback)되어야 함
Consistency (일관성)	트랜잭션이 그 실행을 성공적으로 완료하면 언제나 일관성 있는 데이터베이스 상태로 변환함



Isolation (독립성)	둘 이상의 트랜잭션이 동시에 병행 실행되는 경우 어느 하나의 트랜잭션 실행중에 다른 트랜잭션의 연산이 끼어들 수 없음
Durability (지속성)	성공적으로 완료된 트랜잭션의 결과는 시스템이 고장나더라도 영구적으로 반영되어야 함

2020년 1회 기사 실기

1. 다음은 트랜잭션의 특징이다. 괄호(①, ②)에 들어갈 알맞은 특징을 쓰시오.

- 일관성 : 트랜잭션이 그 실행을 성공적으로 완료하면 언제나 일관성 있는 데이터베이스 상태로 변환한다.
- ( ① ) : 트랜잭션의 연산은 데이터베이스에 모두 반영되도록 완료(Commit)되든지 아니면 전혀 반영되지 않도록 복구(Rollback)되어야 한다(All or Nothing).
- 지속성 : 성공적으로 완료된 트랜잭션의 결과는 시스템이 고장나더라도 영구적으로 반영되어야 한다.
- ( ② ) : 둘 이상의 트랜잭션이 동시에 병행 실행되는 경우 어느 하나의 트랜잭션 실행중에 다른 트랜잭션의 연산이 끼어들 수 없다.

답 :

- ① :  
② :

정답 1. ① 원자성(Atomicity) ② 독립성(Isolation)

### [핵심 053] 클러스터(Cluster)

- 데이터 저장 시 데이터 액세스 효율을 향상시키기 위해 동일한 성격의 데이터를 동일한 데이터 블록에 저장하는 물리적 저장 방법이다.
- 클러스터링으로 지정된 컬럼 값의 순서대로 저장되고, 여러 개의 테이블이 하나의 클러스터에 저장된다.
- 클러스터링 된 테이블은 데이터 조회 속도는 향상시키지만 데이터 입력, 수정, 삭제에 대한 성능은 저하시킨다.
- 처리 범위가 넓은 경우에는 단일 테이블 클러스터링을, 조인이 많이 발생하는 경우에는 다중 테이블 클러스터링을 사용한다.

1. 데이터 저장 시 데이터 액세스 효율을 향상시키기 위해 동일한 성격의 데이터를 동일한 데이터 블록에 저장하는 물리적 저장 방법이 무엇인지 쓰시오.

답 :

정답 1. 클러스터(Cluster)

### [핵심 054] 파티션(Partition)

- 대용량의 테이블이나 인덱스를 작은 논리적 단위로 나누는 것을 말한다.
- 파티셔닝 방식에 따른 파티션의 종류

범위 분할 (Range Partitioning)	지정한 열의 값을 기준으로 분할
해시 분할 (Hash Partitioning)	해시 함수를 적용한 결과 값에 따라 데이터를 분할
조합 분할 (Composite Partitioning)	범위 분할로 분할한 다음 해시 함수를 적용하여 다시 분할

- 인덱스 파티션 : 파티션된 테이블의 데이터를 관리하기 위해 인덱스를 나누는 것

1. ( )은 데이터베이스에서 테이블이나 인덱스를 작은 논리적 단위로 나누는 것으로, ( ) 별로 데이터가 분산되어 저장되므로 디스크의 성능이 향상된다. 괄호에 공통으로 들어갈 알맞은 용어를 쓰시오.

답 :

2. 데이터베이스에서 파티션은 대용량의 테이블이나 인덱스를 작은 논리적 단위로 나누는 것을 말한다. 다음 괄호(①~③)에 들어갈 알맞은 파티션 종류를 쓰시오.

- ( ① )은 범위 분할로 분할한 다음 해시 함수를 적용하여 다시 분할하는 방식이다.
- ( ② )은 일별, 월별, 분기별 등 지정한 열의 값을 기준으로 분할하는 방식이다.
- ( ③ )은 해시 함수를 적용한 결과 값에 따라 데이터를 분할하는 방식이다.



답

- ① :  
② :  
③ :

정답 1. 파티션(Partition)

2. ① 조합 분할(Composite Partitioning) ② 범위 분할(Range Partitioning) ③ 해시 분할(Hash Partitioning)

## [핵심 055] 데이터베이스 용량 설계

- 데이터가 저장될 공간을 정의하는 것이다.
- 데이터베이스 용량을 설계할 때는 테이블에 저장할 데이터양과 인덱스, 클러스터 등이 차지하는 공간 등을 예측하여 반영해야 한다.
- 데이터베이스의 용량을 정확히 산정하여 디스크의 저장 공간을 효과적으로 사용하고 확장성 및 가용성을 높인다.
- 데이터 접근성을 향상시키는 설계 방법
  - 테이블의 테이블스페이스와 인덱스의 테이블스페이스를 분리하여 구성한다.
  - 테이블스페이스와 임시 테이블스페이스를 분리하여 구성한다.
  - 테이블을 마스터 테이블과 트랜잭션 테이블로 분류한다.

1. 다음은 데이터베이스 용량 설계 시 데이터 접근성을 향상시키기 위한 설계 방법이다. 괄호에 공통으로 들어갈 알맞은 용어를 쓰시오.

- 테이블의 ( )와 인덱스의 ( )를 분리하여 구성한다.
- ( )와 임시 ( )를 분리하여 구성한다.
- 테이블을 마스터 테이블과 트랜잭션 테이블로 분류한다.

답 :

정답 1. 테이블스페이스

