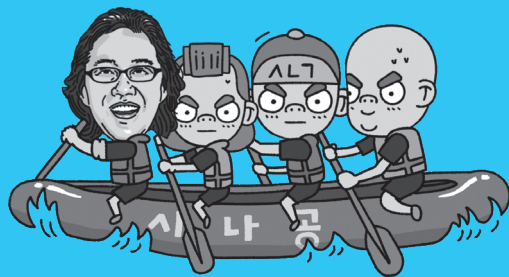


2 과목

정답 및 해설

| 데이터베이스 실무 일반 |





Section 060

[문제 1]

- 통합 데이터(Integrated Data) : 검색의 효율성을 위해 중복이 최소화된 데이터의 모임이다.
- 저장 데이터(Stored Data) : 컴퓨터가 접근 가능한 저장 매체에 저장된 데이터이다.
- 운영 데이터(Operational Data) : 조직의 목적을 위해 존재 가치가 확실하고 반드시 필요한 데이터이다.
- 공유 데이터(Shared Data) : 여러 응용 프로그램들이 공동으로 사용하는 데이터이다.

[문제 2]

① - b ② - c ③ - f ④ - a ⑤ - e ⑥ - d

[문제 3]

데이터 설계자(Data Architect)

[문제 4]

데이터 관리자(Data Administrator)

[문제 5]

데이터베이스 관리자(DBA; DataBase Administrator)

[문제 6]

① DDL 컴파일러 ② 트랜잭션 관리자

Section 061

[문제 1]

a, d, f, g, i

- ⑥ DBMS는 데이터베이스의 전문가가 부족한 단점이 있다.
- ⑦ DBMS는 시스템이 복잡해지는 단점이 있다.
- ⑧ DBMS는 전산화 비용이 증가하는 단점이 있다.
- ⑨ DBMS는 데이터의 백업과 회복이 어려운 단점이 있다.

[문제 2]

데이터 스트림 관리 시스템(DSMS)

[문제 3]

조작

Section 062

[문제 1]

MARC는 목록 레코드를 식별하여 축적 · 유통할 수 있도록 코드화한 메타 데이터이다.

[문제 2]

상호운용성

[문제 3]

MDR은 메타 데이터의 등록과 인증을 통하여 메타 데이터를 유지·관리하며, 메타 데이터의 명세를 공유하는 레지스트리이다.

[문제 4]

- ① 메타 데이터(Meta Data) ② MDR(Meta Data Registry, 메타 데이터 레지스트리)

[문제 5]

- ① 외부 스키마(External Schema) ② 개념 스키마(Conceptual Schema) ③ 내부 스키마(Internal Schema)

Section 063

[문제 1]

- ① 요구 조건 분석 ② 개념적 설계 ③ 논리적 설계 ④ 물리적 설계 ⑤ 데이터베이스 구현

Section 064

[문제 1]

- ① 이름 ② 강좌번호 ③ 학생 ④ 1 ⑤ N

④, ⑤ 하나의 학과에는 여러 명의 교수가 소속될 수 있으나, 교수는 하나의 학과에만 소속되므로 학과와 교수 간에는 일 대 다(1 : N)의 관계가 존재한다. 그러므로 학과 측 대응 카디널리티에는 1을 표시하며, 교수 측 대응 카디널리티에는 N을 표시한다.

[문제 2]

- 1:1 관계 : 관계에 참여하고 있는 두 개체 타입이 모두 하나씩의 개체 어커런스를 갖는 관계이다.
- 1:N 관계 : 관계에 참여하고 있는 개체 타입 중 한 개체 타입은 여러 개의 개체 어커런스를, 다른 한 개체 타입은 하나의 개체 어커런스를 갖는 관계이다.
- N:M 관계 : 관계에 참여하고 있는 두 개체 타입 모두 여러 개의 개체 어커런스를 가질 수 있는 관계이다.

[문제 3]

개체-관계 모델 또는 E-R 모델(E-R Diagram)

예상문제은행

1장

정답

데이터베이스 기본



[문제 1]

- 외부 스키마 : 사용자나 응용 프로그래머가 각 개인의 입장에서 필요로 하는 데이터베이스의 논리적 구조를 정의한 것이다.
- 내부 스키마 : 데이터베이스의 물리적 구조를 정의한 것이다.
- 개념 스키마 : 개체 간의 관계와 제약 조건을 나타내고 데이터베이스의 접근 권한, 보안 정책 및 무결성 규정에 관한 명세를 정의한 것이다.

[문제 2]

- DDL(데이터 정의어) : 데이터베이스 구조, 데이터 형식, 접근 방식 등 데이터베이스를 구축하거나 변경할 목적으로 사용하는 언어로, 데이터베이스의 논리적, 물리적 구조를 정의 및 변경하며, 스키마(Schema)에 사용되는 제약 조건을 정의한다.
- DCL(데이터 제어어) : 보안 및 권한 제어, 무결성, 회복, 병행 제어를 위한 언어이다.
- DML(데이터 조작어) : 데이터 처리를 위해서 응용 프로그램과 데이터베이스 관리 시스템 간의 인터페이스를 위한 언어이며, 데이터 처리를 위한 연산의 집합으로 데이터의 검색, 삽입, 삭제, 갱신 연산 등이 있다.

[문제 3]

- ① - ② ② - ③ ③ - ①

[문제 4]

사용자와 데이터베이스 사이에서 사용자의 요구에 따라 정보를 생성해 주고 데이터베이스를 관리해 주는 소프트웨어이다.

[문제 5]

- ① - ㉠ ② - ㉡ ③ - ㉢ ④ - ㉣ ⑤ - ㉤

[문제 6]

- ① 외부 스키마(External Schema) ② 개념 스키마(Conceptual Schema) ③ 내부 스키마(Internal Schema)

[문제 7]

- ① 주소 ② 취미 ③ 경력 ④ 사원번호

[문제 8]

- ① 개체(Entity) 타입 ② 관계(Relationship) 타입 ③ 속성(Attribute) ④ 기본키 속성 ⑤ 개체 타입 간의 연관성
⑥ 개체 타입과 속성을 연결





Section 065

[문제 1]

- ① 스키마(schema) ② 인스턴스(instance) ③ 차수(degree) ④ 카디널리티(cardinality) ⑤ 원자

[문제 2]

1 : 0 또는 1 : 1

관계의 표현

관계	표현	의미
1:1		양쪽에 반드시 1개씩 존재
1:0 또는 1:1		왼쪽에는 반드시 1개, 오른쪽에는 없거나 1개 존재
1:N		왼쪽에는 반드시 1개, 오른쪽에는 반드시 여러 개 존재
1:1 또는 1:N		왼쪽에는 반드시 1개, 오른쪽에는 1개 또는 여러 개 존재
1:0 또는 1:1 또는 1:N		왼쪽에는 반드시 1개, 오른쪽에는 0개 또는 1개 또는 여러 개 존재

[문제 3]

교차 엔티티(Intersection Entity)

[문제 4]

식별 관계(Identifying)

[문제 5]

비식별 관계(Non-identifying)

[문제 6]

- ① 1:1(일대일) ② 1:N(일대다)

[문제 7]

릴레이션(Relation)

[문제 8]

- ① 5 ② 3

[문제 9]

- ① 릴레이션(Relation) ② 도메인(Domain)

Section 066

[문제 1]

- 기본키 : 후보키 중에서 특별히 선정된 키로 중복된 값을 가질 수 없다.
- 후보키 : 릴레이션을 구성하는 속성들 중에서 튜플을 유일하게 식별하기 위해 사용되는 속성들의 부분집합이다. 유일성과 최소성을 만족시킨다.
- 슈퍼키 : 한 릴레이션 내에 있는 속성들의 집합으로 구성된 키이다. 유일성은 만족시키지만 최소성은 만족시키지 못한다.
- 외래키 : 다른 릴레이션의 기본키를 참조하는 속성 또는 속성들의 집합이다.
- 대체키 : 후보키 중에서 선정된 기본키를 제외한 나머지 후보키이다.

[문제 2]

복합키(Composite Key)

Section 067

[문제 1]

무결성(Integrity)

[문제 2]

- 참조(Referential) 무결성 : 외래키 값은 NULL이거나 참조 릴레이션의 기본키 값과 동일해야 한다는 규정이다.
- 개체(Entity) 무결성 : 기본 릴레이션의 기본키를 구성하는 어떤 속성도 NULL일 수 없다는 규정이다.
- 도메인(Domain) 무결성 : 특정 속성의 값이 그 속성이 정의된 도메인에 속한 값이어야 한다는 규정이다.

[문제 3]

- 참조 무결성 위배 사유 : <직원> 릴레이션의 '지점번호' 중 103, 108과 같이 <지점> 릴레이션의 '지점번호' 속성에 없는 '지점번호'가 있기 때문에 참조 무결성 제약조건에 위배된다.
- 개체 무결성 위배 사유 : <지점> 릴레이션의 '지점번호' 속성에 NULL 값이 존재하므로 개체 무결성이 위배된다.

Section 068

[문제 1]

관계해석(Relational Calculus)

[문제 2]

관계대수(Relational Algebra)

[문제 3]

$\sigma_{\text{학년}=4}(\text{학생})$

<학생> 릴레이션에서 학년이 4인 조건을 만족하는 튜플들의 행 전체를 모두 표시해야 하므로 SELECT 연산자를 사용하여 " $\sigma_{\text{학년}=4}(\text{학생})$ "으로 쓸 수 있다.

[문제 4]

$\pi_{\text{학년, 학과}}(\text{학생})$

<학생> 릴레이션에서 일부 속성인 학년과 학과 속성의 값을 모두 추출하는 연산이므로 PROJECT 연산자를 이용하여 " $\pi_{\text{학년, 학과}}(\text{학생})$ "으로 쓸 수 있다.

[문제 5]

JOIN

[문제 6]

세타(Theta)

[문제 7]

자연(NATURAL)

[문제 8]

- ① σ ② π ③ \bowtie ④ \div ⑤ \cup ⑥ \cap ⑦ $-$ ⑧ \times



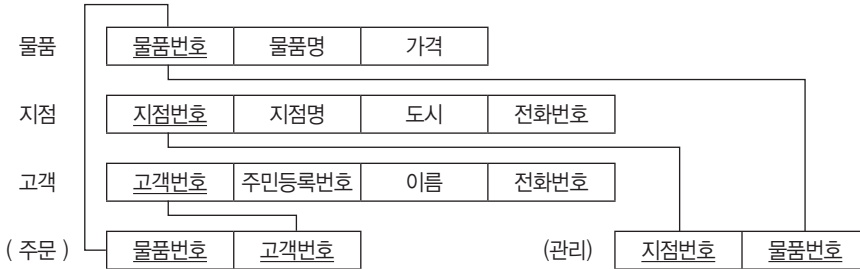
[문제 1]

- ① 속성(Attribute) ② 튜플(Tuple) ③ 스키마(Schema) ④ 인스턴스(Instance)

[문제 2]

- ① - d ② - c ③ - a ④ - b ⑤ - e

[문제 3]



[문제 4]

PROJECT

[문제 5]

SELECT

[문제 6]

- ① 후보키(Candidate Key) ② 대체키(Alternate Key) ③ 외래키(Foreign Key)

[문제 7]

매핑 룰(Mapping Rule)

[문제 8]

- ① 도메인(Domain) 무결성 ② 개체(Entity) 무결성 ③ 관계(Relationship) 무결성 ④ NULL 무결성 ⑤ 참조(Referential) 무결성
⑥ 키(Key) 무결성 ⑦ 고유(Unique) 무결성

[문제 9]

기본키는 중복된 값을 가질 수 없는데, '학과'나 '학년'에는 중복된 값이 존재하므로 기본키로 사용할 수 없다.

[문제 10]

참조 무결성 위배 : <예금계좌> 릴레이션의 '지점명' 중 "화양점"과 "명동점"이 <지점> 릴레이션의 '지점명' 속성에 존재하지 않기 때문에 참조 무결성 제약조건에 위배된다.

[문제 11]

$\sigma_{\text{학과}=\text{'컴퓨터'}}(\text{학생})$

<학생> 릴레이션에서 '학과'가 "컴퓨터"인 조건을 만족하는 튜플들의 행 전체를 모두 표시해야 하므로 SELECT 연산자를 사용하여 " $\sigma_{\text{학과}=\text{'컴퓨터'}}(\text{학생})$ "으로 표기해야 한다.

[문제 12]

$\pi_{\text{학번,이름}}(\text{학생})$

〈학생〉 릴레이션에서 일부 속성인 '학번'과 '이름' 속성만을 추출하는 연산이므로 PROJECT 연산자를 이용하여 " $\pi_{\text{학번,이름}}(\text{학생})$ "으로 쓸 수 있다.

[문제 13]

$\pi_{\text{학번,이름}}(\sigma_{\text{학과}=\text{"컴퓨터"} \wedge \text{학년}=4}(\text{학생}))$

SELECT 연산을 적용하여 〈학생〉 릴레이션에서 '학과'가 "컴퓨터"이고 '학년'이 4인 조건을 만족하는 튜플들의 부분집합을 추출한 다음, PROJECT 연산을 적용하여 '학번'과 '이름' 속성을 추출하면 되므로 관계대수식은 " $\pi_{\text{학번,이름}}(\sigma_{\text{학과}=\text{"컴퓨터"} \wedge \text{학년}=4}(\text{학생}))$ "으로 표기할 수 있다.

[문제 14]

- ① Cartesian Product, 카티션 프로덕트, 교차곱 ② select





Section 069

[문제 1]

```
CREATE TABLE patient
( id CHAR(5) PRIMARY KEY,
  name CHAR(10),
  sex CHAR(1),
  phone CHAR(20),
  CONSTRAINT sex_ck CHECK (sex='f' or sex='m'),

  CONSTRAINT id_fk FOREIGN KEY(id) REFERENCES doctor(doc_id) );
```

생성한 테이블 이름은 <patient>이다.

'id' 속성은 문자 5자리이며, 기본키이다.

'name' 속성은 문자 10자리이다.

'sex' 속성은 문자 1자리이다.

'phone' 속성은 문자 20자리이다.

'sex' 속성은 'f' 또는 'm'만 입력되어야 하며, 이 제약 조건의 이름은 'sex_ck'이다.

'id' 속성은 <doctor> 테이블의 기본키인 'doc_id' 속성을 참조하는 외래키이며, 이 제약 조건의 이름은 'id_fk'이다.

- ※ 기본키를 'phone CHAR(20),' 다음 줄에 'PRIMARY KEY(id),'로 지정해도 되고 속성을 정의하면서 'id CHAR(5) PRIMARY KEY,'와 같이 추가로 기본키를 지정해도 됩니다.
- ※ 외래키를 'phone CHAR(20),' 다음 줄에 'FOREIGN KEY(id) REFERENCES doctor(doc_id)'와 같이 입력해도 되지만 제약조건명이 있으므로 CONSTRAINT절을 사용해야 합니다.

[문제 2]

```
ALTER TABLE patient
ADD job CHAR(20);
```

수정할 테이블 이름은 <patient>이다.

문자 20자리인 'job' 속성을 추가한다.

[문제 3]

```
CREATE TABLE Instructor
( id CHAR(5) NOT NULL,
  name CHAR(15) NOT NULL,
  dept CHAR(15),
  PRIMARY KEY(id),
  FOREIGN KEY(dept) REFERENCES Department(name)

  ON DELETE SET NULL

  ON UPDATE CASCADE
);
```

생성한 테이블 이름은 <Instructor>이다.

'id' 속성은 문자 5자리이며, NULL 값을 갖지 않는다.

'name' 속성은 문자 15자리이며, NULL 값을 갖지 않는다.

'dept' 속성은 문자 15자리이다.

'id' 속성은 기본키이다.

'dept' 속성은 <Department> 테이블의 기본키인 'name' 속성을 참조하는 외래키이다.

<Department> 테이블에서 튜플이 삭제되면 관련된 모든 튜플의 'dept' 속성의 값을 NULL로 변경한다.

<Department> 테이블에서 튜플이 변경되면 관련된 모든 튜플의 'dept' 속성의 값도 같은 값으로 변경한다.

[문제 4]

```
CREATE VIEW CC(ccid, ccname, instname) AS

SELECT Course.id, Course.name, Instructor.name

FROM Course, Instructor

WHERE Course.instructor = Instructor.id;
```

생성한 뷰의 '이름'은 <CC>이며, 'ccid', 'ccname', 'instname' 속성을 갖는다.

<Course> 테이블에서 'id'와 'name' 속성을 가져오고, <instructor> 테이블에서 'name' 속성을 가져온다.

<Course>와 <instructor> 테이블에서 속성을 가져와 뷰를 생성한다.

<course> 테이블의 'instructor' 속성과 <instructor> 테이블의 'id' 속성의 값이 같은 자료만을 뷰로 생성한다.

[문제 5]

```
CREATE UNIQUE INDEX Stud_idx

On Student(ssn ASC);
```

생성한 인덱스의 이름은 <Stud_idx>이며, 중복값이 없는 속성으로 인덱스를 생성한다.

<Student> 테이블의 'ssn' 속성을 오름차순 정렬한다.

※ ASC는 생략할 수 있습니다.

[문제 6]

① FOREIGN KEY REFERENCES ② CASCADE

```
CREATE TABLE 사원
( 사원번호 NUMBER(4) PRIMARY KEY,
  사원명 VARCHAR2(10),
  근무지번호 NUMBER(2) FOREIGN KEY REFERENCES 근무지
ON DELETE CASCADE );
```

생성한 테이블 이름은 <사원>이다.

'사원번호' 속성은 숫자 4자리이며, 기본키이다.

'사원명' 속성은 문자 10자리이다.

'근무지번호'는 숫자 2자리이며, <근무지> 테이블의 '근무지번호'를 참조하는 외래키이다.

<근무지> 테이블에서 '근무지번호'가 삭제되면, <사원> 테이블의 '근무지번호'도 삭제된다.

※ 'FOREIGN KEY (속성) REFERENCES 테이블(속성)' 절을 생략하고 외래키로 지정할 속성 옆에 직접 'FOREIGN KEY REFERENCES 테이블명'을 추가할 수 있습니다.

[문제 7]

① DEFAULT '사원' ② CONSTRAINT ③ CHECK (VALUE IN('사원', '대리', '과장', '부장', '이사', '사장'))

```
CREATE DOMAIN 직위 VARCHAR2(10)
DEFAULT '사원'
CONSTRAINT VALID-직위 CHECK (
  VALUE IN('사원', '대리', '과장', '부장', '이사', '사장') );
```

'직위' 속성에 대한 도메인을 생성한다. 크기는 문자 10자리이다.

'직위' 도메인의 기본 값은 "사원"이다.

'직위' 속성에는 "사원", "대리", "과장", "부장", "이사", "사장" 중 하나의 값만을 저장할 수 있다. 'VALID-직위'는 제약 조건의 이름이다.

[문제 8]

```
CREATE INDEX 직원__name ON 직원(이름);
```

<직원> 테이블의 '이름' 속성으로 '직원__name'이라는 이름의 인덱스를 생성한다.

[문제 9]

TRIGGER

```
CREATE TRIGGER 월급_trg AFTER UPDATE ON 급여
REFERENCING OLD TABLE AS old_tbl NEW TABLE AS new_tbl
FOR EACH ROW
WHEN old_tbl.직위=new_tbl.직위
BEGIN
    UPDATE 사원 SET 월급 = new_tbl.기본급+new_tbl.성과급;
END;
```

테이블에 레코드를 갱신하면 동작하는 '월급_trg'를 생성한다.

'월급_trg' 트리거는 추가될 튜플들에 적용되며 그 튜플들의 집합을 <new_tbl>로 명명한다.

튜플들의 수만큼 반복실행한다.

'old_tbl'의 '직위' 속성과 'new_tbl'의 '직위' 속성이 같은 레코드에 대해서만 '월급_trg'가 적용된다.

트리거가 동작시킬 SQL문장의 시작이다.

<사원> 테이블의 '월급' 속성을 <new_tbl>의 '기본급' 속성과 '성과급' 속성의 값을 더한 값으로 치환한다.

트리거가 동작시킬 SQL문장의 끝이다.

[문제 10]

CASCADE

DROP TABLE 학생 CASCADE;

<학생> 테이블을 삭제하되, <학생> 테이블을 참조하는 다른 모든 개체를 함께 제거한다.

[문제 11]

① CHECK ② IN

```
CREATE TABLE 사원
(직원코드 NUMBER NOT NULL,
성명 CHAR(10) UNIQUE,
직책 CHAR(10) CHECK (직책 IN('사원', '대리', '과장', '팀장')),
연봉 NUMBER);
```

<사원> 테이블을 생성한다.

'직원코드' 속성은 숫자형으로, NULL을 갖지 않는다.

'성명' 속성은 최대 문자 10자로, 중복된 값을 가질 수 없다.

'직책' 속성은 최대 문자 10자로, '사원', '대리', '과장', '팀장' 중에 하나의 값을 가져야 한다.

'연봉' 속성은 숫자형이다.

[문제 12]

Create, Alter, Drop

Section 070

[문제 1]

```
SELECT 학번, 이름
FROM 학생
WHERE 학년 IN (3, 4);
```

'학번', '이름'을 표시한다.

<학생> 테이블에서 검색한다.

'학년'의 값이 3 또는 4인 자료만을 대상으로 한다.

[문제 2]

```
SELECT DISTINCT name
FROM Shop
WHERE id IN (

    SELECT shopid
    FROM Staff
    WHERE id = 10);
```

'name'을 표시하되, 같은 'name' 속성의 값은 한 번만 표시한다.
<Shop> 테이블에서 검색한다.
<Shop> 테이블의 'id'가 IN 다음에 쓰인 하위 질의의 검색 결과와 같은 자료만을 대상으로 한다.
'shopid'를 표시한다.
<Staff> 테이블에서 검색한다.
'id'가 10인 자료만을 대상으로 한다. 즉 'id'가 10인 자료의 'shopid' 속성의 값이 표시된다.

※ 조건절의 설명을 위해 4줄에 걸쳐 썼지만 보통 'WHERE id IN (SELECT shopid FROM Staff WHERE id = 10);'와 같이 한 줄로 입력합니다.

[문제 3]

```
SELECT SUM(psale)
FROM Sale
WHERE pid IN (

    SELECT id
    FROM Product
    WHERE name LIKE 'USB%');
```

'psale'의 합계를 표시한다.
<Sale> 테이블을 대상으로 검색한다.
<Sale> 테이블의 'pid'가 IN 다음에 쓰인 하위 질의의 결과와 같은 자료만을 대상으로 한다.
'id'를 표시한다.
<Product> 테이블에서 검색한다.
'name'이 "USB"로 시작하는 자료만을 대상으로 한다. 즉 'name'이 "USB"로 시작하는 자료의 'id' 속성의 값이 표시된다.

[문제 4]

```
SELECT name
FROM Product
WHERE price IS NULL

ORDER BY name;
```

'name'을 표시한다.
<Product> 테이블을 대상으로 검색한다.
'price' 속성의 값이 NULL인, 즉 내용이 입력되지 않은 튜플만을 대상으로 검색한다.
'name' 속성의 값을 기준으로 오름차순 정렬한다.

[문제 5]

```
SELECT pid
FROM Sale
WHERE psale BETWEEN 10 AND 20;
```

'pid'를 표시한다.
<Sale> 테이블을 대상으로 검색한다.
'psale' 속성의 값이 10~20 사이인 자료만을 대상으로 한다.

※ 'psale BETWEEN 10 AND 20' 대신 'psale >= 10 and psale <= 20'으로 입력해도 결과는 같습니다.

[문제 6]

```
SELECT 학생정보.학번, 이름, 결제여부

FROM 학생정보, 신청정보, 결제
WHERE 학생정보.학번 = 신청정보.학번
    AND 신청정보.신청번호 = 결제.신청번호
    AND 신청과목 = 'OpenGL';
```

'학번', '이름', '결제여부'를 표시한다. '학번' 속성은 여러 테이블에 있으므로 테이블명을 지정해야 한다. '신청정보.학번' 또는 '결제.학번'을 입력해도 결과는 같다.
<학생정보>, <신청정보>, <결제> 테이블을 대상으로 검색한다.
<학생정보> 테이블의 '학번'과 <신청정보> 테이블의 '학번'이 같고
<신청정보> 테이블의 '신청번호'와 <결제> 테이블의 '신청번호'가 같으며,
'신청과목'이 "OpenGL"인 학생만을 대상으로 한다.

[문제 7]

```
SELECT 이름, 전공, 신청과목
FROM 학생정보, 신청정보
WHERE 학생정보.학번 = 신청정보.학번
      AND 신청과목 = 'JAVA'
GROUP BY 이름, 전공, 신청과목
HAVING 전공 = '컴퓨터공학';
```

'이름', '전공', '신청과목'을 표시한다.
 <학생정보>와 <신청정보> 테이블을 대상으로 검색한다.
 <학생정보> 테이블의 '학번'과 <신청정보> 테이블의 '학번'이 같고
 '신청과목'이 "JAVA"인 학생만을 대상으로 한다.
 '이름', '전공', '신청과목'을 기준으로 그룹을 지정한다.
 그룹 중 '전공'이 "컴퓨터공학"인 학생만을 표시한다.

[문제 8]

```
SELECT 결제여부, COUNT(*) AS 학생수
FROM 결제
GROUP BY 결제여부;
```

'결제여부'와 개수를 표시하되, 개수는 '학생수'로 표시한다.
 <결제> 테이블을 대상으로 검색한다.
 '결제여부'를 기준으로 그룹을 지정한다.

[문제 9]

```
SELECT DISTINCT 과목
FROM 학생
WHERE 학년 >= 3 AND 점수 >= 80;
```

'과목'을 표시하되, 같은 '과목'은 한 번만 표시한다.
 <학생> 테이블을 대상으로 한다.
 '학년' 속성의 값이 3 이상이고 '점수' 속성의 값이 80 이상인 자료만을
 대상으로 한다.

[문제 10]

- ① SELECT ID, NAME FROM CUSTOMER;
- ② SELECT DISTINCT GRADE FROM CUSTOMER;
- ③ SELECT * FROM CUSTOMER ORDER BY ID DESC;
- ④ SELECT NAME FROM CUSTOMER WHERE AGE IS NULL;
- ⑤ SELECT NAME FROM CUSTOMER WHERE AGE IS NOT NULL;

• ①번 수행 결과

ID	NAME
G001	KIM
G002	LEE
G003	PARK
G004	JUNG
G005	JO
G006	CHOI

• ②번 수행 결과

GRADE
GOLD
VIP
SILVER

• ③번 수행 결과

ID	NAME	AGE	GRADE	JOB	POINT
G006	CHOI	50	VIP	자영업	3000
G005	JO	NULL	SILVER	주부	650
G004	JUNG	28	SILVER	학생	800
G003	PARK	30	GOLD	프리랜서	1500
G002	LEE	42	VIP	회사원	2500
G001	KIM	35	GOLD	회사원	1800

• ④번 수행 결과

NAME
JO

• ⑤번 수행 결과

NAME
KIM
LEE
PARK
JUNG
CHOI

Section 071

[문제 1]

① = ② NATURAL JOIN ③ USING

방법1	SELECT 이름, 학과명 AS 학과 FROM 학생, 학과 WHERE 학생.학과코드 = 학과.학과코드;	'이름'과 '학과명'을 표시하되, '학과명'은 '학과'로 표시한다. 〈학생〉테이블과 〈학과〉테이블을 대상으로 한다. 〈학생〉테이블의 '학과코드'와 〈학과〉테이블의 '학과코드'가 같은 튜플을 JOIN한다.
방법2	SELECT 이름, 학과명 AS 학과 FROM 학생 NATURAL JOIN 학과;	'이름'과 '학과명'을 표시하되, '학과명'은 '학과'로 표시한다. 〈학생〉테이블과 〈학과〉테이블에서 같은 속성을 기준으로 JOIN한다.
방법3	SELECT 이름, 학과명 AS 학과 FROM 학생 JOIN 학과 USING(학과코드);	'이름'과 '학과명'을 표시하되, '학과명'은 '학과'로 표시한다. 〈학생〉테이블과 〈학과〉테이블에서 '학과코드'를 기준으로 JOIN한다.

※ 〈학생〉테이블과 〈학과〉테이블에는 같은 이름의 속성과 범위가 같은 도메인을 갖는 '학과코드'가 있기 때문에 NATURAL JOIN이 가능합니다.

[문제 2]

① 학생 LEFT OUTER JOIN 학과 ② 학생.학과코드 = 학과.학과코드(+)

방법1	SELECT 학번, 학생.학과코드, 학과명 FROM 학생 LEFT OUTER JOIN 학과 ON 학생.학과코드 = 학과.학과코드;	'학번', '학과코드', '학과명'을 표시한다. 〈학생〉테이블의 '학과코드'와 〈학과〉테이블의 '학과코드'로 JOIN하되, 〈학생〉테이블의 튜플들은 〈학과〉테이블의 연관성과 관계없이 모두 표시하는 LEFT OUTER JOIN이다.
방법2	SELECT 학번, 학생.학과코드, 학과명 FROM 학생, 학과 WHERE 학생.학과코드 = 학과.학과코드(+);	'학번', '학과코드', '학과명'을 표시한다. 〈학생〉테이블과 〈학과〉테이블을 '학과코드'로 JOIN하되, 〈학생〉테이블의 튜플들은 〈학과〉테이블의 연관성과 관계없이 모두 표시하는 LEFT OUTER JOIN이다.

Section 072

[문제 1]

```
INSERT INTO 학생  
VALUES(98170823, '한국산', 3, '경영학개론', '?-1234-1234');
```

〈학생〉 테이블에 삽입하라.
첫 번째 필드부터 순서대로 98170823, '한국산', 3, '경영학개론',
'?-1234-1234'를 삽입하라.

[문제 2]

```
DELETE  
FROM 학생  
WHERE 이름 = 'Scott';
```

삭제하라.
〈학생〉 테이블을 대상으로 하라.
'이름'이 'Scott'인 학생만을 대상으로 하라.

[문제 3]

① SET ② WHERE

```
UPDATE 사원  
SET 연봉 = 연봉 + 100000  
WHERE 직급 = '차장';
```

〈사원〉 테이블을 갱신하라.
'연봉'을 '연봉'에 100000을 더한 값으로 갱신하라.
'직급'이 '차장'인 사원만을 대상으로 하라.

[문제 4]

① SET ② 학과번호 ③ LIKE

```
UPDATE 학부생  
SET 학과번호 = 999  
WHERE 담당관 LIKE '이%';
```

〈학부생〉 테이블을 갱신하라.
'학과번호'를 999로 갱신하라.
'담당관'의 이름이 '이' 시작하는 학부생을 대상으로 하라.

[문제 5]

```
INSERT INTO 기획부(성명, 경력, 주소, 기본급)  
SELECT 성명, 경력, 주소, 기본급  
FROM 사원  
WHERE 부서 = '기획';
```

〈기획부〉 테이블의 '성명', '경력', '주소', '기본급'에 삽입하라.
삽입할 자료 '성명', '경력', '주소', '기본급'을 검색하라.
〈사원〉 테이블을 대상으로 하라.
'부서'가 '기획'인 튜플만을 대상으로 하라.

[문제 6]

```
UPDATE 사원  
SET 부서 = '편집'  
WHERE 성명 IN ('장발장', '황진이');
```

〈사원〉 테이블을 갱신하라.
'부서'를 '편집'으로 갱신하라.
'성명'이 '장발장' 또는 '황진이'인 사원만을 대상으로 하라.

[문제 7]

```
INSERT  
INTO 사원  
  
VALUES ('황진이', '기획', 10, '서교동', 2000000);
```

삽입하라.
〈사원〉 테이블에 삽입하라.
'황진이', '기획', 10, '서교동', 2000000을 〈사원〉 테이블의 각 필드에
삽입하라. 〈사원〉 테이블의 '성명'에 '황진이', '부서'에 '기획', '경력'에
10, '주소'에 '서교동', '기본급'에 2000000이 삽입된다.

[문제 8]

DELETE
FROM 사원;

삭제하라.
〈사원〉 테이블을 대상으로 하라. WHERE 조건이 없으므로 〈사원〉
테이블의 모든 데이터가 삭제된다..

[문제 9]

Insert, Update, Select, Delete

Section 073

[문제 1]

GRANT ALL ON 학생 TO 김하늘;

[문제 2]

GRANT DELETE ON 강좌 TO 김하늘 WITH GRANT OPTION;

[문제 3]

REVOKE SELECT, INSERT, DELETE ON 교수 FROM 임격정;

[문제 4]

REVOKE UPDATE ON 수강 FROM 임격정 CASCADE;

※ 'GRANT OPTION FOR'를 생략한 이유는 임격정에게 부여된 〈수강〉 테이블에 대한 UPDATE 권한을 취소하면 다른 사람에게 UPDATE 권한을 부여할 수 있는 권한도 함께 취소되기 때문이다. 자신에게 권한이 없어지면 해당 권한을 다른 사람에게 부여할 수 없다.

[문제 5]

COMMIT 연산은 트랜잭션 처리가 정상적으로 종료되어 트랜잭션이 수행한 변경 내용을 데이터베이스에 반영하는 연산이다.

Section 074

[문제 1]

CREATE VIEW ST_목록
AS SELECT 제품.제품코드, 제품.제품명, 판매.수량

FROM 제품, 판매
WHERE 제품.제품코드 = 판매.제품ID;

생성한 뷰의 이름은 〈ST_목록〉이다.
〈제품〉 테이블에서 '제품코드', '제품명', 〈판매〉 테이블의 '수량'으로
작성한다.
〈제품〉과 〈판매〉 테이블에서 속성을 가져와 뷰를 생성한다.
〈제품〉 테이블의 '제품코드' 속성과 〈판매〉 테이블의 '제품ID' 속성으로
JOIN한다.

[문제 2]

DROP VIEW ST_목록 RESTRICT;

[문제 3]

DROP VIEW ST_목록 CASCADE;

Section 075

[문제 1]

① DECLARE SECTION ② DECLARE ③ 직원 ④ OPEN ⑤ FETCH ⑥ UPDATE ⑦ CURRENT OF ⑧ CLOSE

내장 SQL과 호스트 언어의 실행문 구별

- 내장 SQL문을 선행처리가 식별할 수 있도록 하기 위해 다음과 같은 형식으로 사용한다.

```
EXEC SQL <내장 SQL문> EXEC SQL END
```

- 내장 SQL문의 사용은 SQL이 내장될 호스트 언어에 따라 다르다.
 - C, C++ 언어에서는 'EXEC SQL END' 대신 세미콜론(;)을 사용한다.
- 변수의 구분
 - 내장 SQL 내에서 사용하는 호스트 변수는 변수 앞에 콜론(:) 문자를 붙인다.
 - 호스트 언어 내에서 호스트 변수는 콜론(:) 없이 그대로 사용한다.

커서 관련 명령어

- DECLARE : 커서를 정의하는 등 커서에 관련된 선언을 하는 명령어
- OPEN : 커서가 질의 결과의 첫 번째 튜플을 가리키도록 설정하는 명령어
- FETCH : 질의 결과의 튜플들 중 현재의 다음 튜플로 커서를 이동시키는 명령어
- CLOSE : 질의 실행 결과에 대한 처리 종료 시 커서를 닫기 위해 사용하는 명령어

```
salaryIncrease( )
{
    EXEC SQL BEGIN DECLARE SECTION;
        char SQLSTATE[6];
    EXEC SQL END DECLARE SECTION;

    EXEC SQL DECLARE employeeCur CURSOR FOR 직원;

    EXEC SQL OPEN employeeCur;
        while( 1 ){
            EXEC SQL FETCH employeeCur;

            if (!(strcmp(SQLSTATE, "02000"))) break;

            if (2017 - 입사년도 >= 15)
                EXEC SQL UPDATE 직원
                    SET 연봉 = 연봉 + (연봉 * 0.15)
                    WHERE CURRENT OF employeeCur;
            EXEC SQL CLOSE employeeCur;
        }
}
```

타입이 문자형인 변수 SQLSTATE를 선언한다.

〈직원〉 테이블 전체를 나타내는 커서 employeeCur를 선언한다.

커서 employeeCur가 〈직원〉 테이블의 첫 번째 튜플을 가리키도록 한다.

커서 employeeCur가 가리키는 〈직원〉 테이블의 튜플을 차례대로 가져온다.

변수 SQLSTATE의 값이 "02000"이면 커서 employeeCur가 더 이상 가리키는 튜플이 없는 것이므로 작업을 종료한다.

재직년수가 15년 이상이면 월급을 15% 증액하는 갹신 연산을 수행한다.

커서 employeeCur를 닫는다.

※ WHERE CURRENT OF employeeCur은 조건절로서 커서 employeeCur이 가리키는 현재의 튜플을 말한다.

Section 076

[문제 1]

① IN ② OUT ③ CURSOR FOR ④ deptName ⑤ BEGIN ⑥ SET ⑦ END

스토어드 프로시저 생성

스토어드 프로시저 생성을 위해서는 CREATE PROCEDURE 명령어를 사용한다.

```
CREATE PROCEDURE 프로시저명(파라미터)
```

```
지역변수선언
```

```
프로시저 BODY;
```

- **프로시저명** : 생성하려는 스토어드 프로시저의 이름이다.
- **파라미터** : 스토어드 프로시저 파라미터로는 다음과 같은 것들이 올 수 있다.
 - IN : 호출되는 스토어드 프로시저에 값을 전달하는 것을 지정한다.
 - OUT : 스토어드 프로시저가 그 호출 프로그램에게 값을 반환한다는 것을 지정한다.
 - INOUT : 스토어드 프로시저에 값을 전달하고, 스토어드 프로시저는 실행 후 호출 프로그램에 값을 반환해야 한다는 것을 지정한다.
- **프로시저 BODY** : 스토어드 프로시저의 본문 코드를 기록하는 부분이다. 적어도 하나의 SQL문이 있어야 한다. 그렇지 않으면 오류가 발생한다.

```
① CREATE PROCEDURE findEmployee(  
②   IN deptName CHAR(15),  
   OUT avgSalary REAL)  
③ DECLARE Not_Found CONDITION FOR SQLSTATE '02000';  
④ DECLARE salaryCur CURSOR FOR  
   SELECT 연봉 FROM 직원 WHERE 부서명 = deptName;  
⑤ DECLARE salaryVar INTEGER;  
   DECLARE employeeCount INTEGER;  
   DECLARE salarySum INTEGER;  
⑥ BEGIN  
⑦   SET salarySum = 0;  
   SET avgSalary = 0.0;  
   SET employeeCount = 0;  
⑧   OPEN salaryCur;  
⑨   salaryLoop: LOOP  
⑩     FETCH salaryCur INTO salaryVar;  
⑪     IF Not_Found THEN  
       LEAVE salaryLoop;  
     END IF;  
⑫     SET employeeCount = employeeCount + 1;  
⑬     SET salarySum = salarySum + salaryVar;  
⑭   END LOOP;  
⑮   SET avgSalary = salarySum / employeeCount;  
⑯   CLOSE salaryCur;  
⑰ END;
```

- ① 생성할 스토어드 프로시저명은 findEmployee이다.
- ② 입력 파라미터로 문자형인 deptName과 출력 파라미터로 실수형인 avgSalary를 가진다.
- ③ Not_Found를 SQLSTATE 값 '02000'을 나타내는 변수로 선언한다.
- ④ <직원> 테이블에서 부서명이 deptName과 같은 직원의 연봉을 검색한 질의 결과를 나타내는 커서 salaryCur를 선언한다.
- ⑤ 타입이 정수형인 변수 salaryVar, employeeCount, salarySum을 선언한다.
- ⑥ 스토어드 프로시저의 본문 코드가 시작된다.
- ⑦ 변수 salarySum, avgSalary를 0.0, employeeCount를 0으로 초기화한다.
- ⑧ 커서 salaryCur가 질의 결과의 첫 번째 튜플을 가리키도록 한다.
- ⑨ 루프 salaryLoop를 시작한다.
- ⑩ 커서 salaryCur가 가리키는 튜플의 연봉을 변수 salaryVar에 저장한다.
- ⑪ 커서 salaryCur가 가리키는 튜플이 없으면 루프를 빠져나온다.
- ⑫ 변수 employeeCount 값을 1 증가시킨다.
- ⑬ 변수 salarySum에 변수 salaryVar 값을 더한다.
- ⑭ 루프 salaryLoop의 끝을 나타낸다.
- ⑮ salarySum을 employeeCount로 나눈 값을 avgSalary에 저장한다.
- ⑯ 커서 salaryCur를 닫는다.
- ⑰ 스토어드 프로시저의 본문 코드가 종료된다.

예상문제는 행

3장

정답

SQL



[문제 1]

DROP TABLE 직원;

<직원> 테이블을 제거한다.

[문제 2]

```
CREATE TABLE 직원
(사번 CHAR(15),
이름 CHAR(4) NOT NULL,
전화번호 CHAR(20),
부서번호 CHAR(10),
경력 INT,
기본급 INT,
PRIMARY KEY(사번),
UNIQUE(전화번호),
FOREIGN KEY(부서번호) REFERENCES 부서(부서번호),
CHECK (기본급 >= 1000000));
```

생성한 테이블 이름은 <직원>이다.
 '사번' 속성은 문자 15자리이다.
 '이름' 속성은 문자 4자리인데, NULL 값을 가질 수 없다.
 '전화번호' 속성은 문자 20자리이다.
 '부서번호' 속성은 문자 10자리이다.
 '경력' 속성은 숫자이다.
 '기본급' 속성은 숫자이다.
 '사번' 속성은 기본키이다.
 '전화번호' 속성은 중복된 값을 가질 수 없다.
 '부서번호' 속성은 <부서> 테이블의 '부서번호'를 참조하는 외래키이다.
 '기본급' 속성은 1000000보다 크거나 같은 값을 가져야 한다.

[문제 3]

SELECT *
FROM 사원;

모든 속성을 표시한다.
 <사원> 테이블을 대상으로 검색한다.

[문제 4]

```
SELECT DISTINCT 이름
FROM 자격증
WHERE 경력 >= 3;
```

'이름'을 표시하되, 같은 '이름'은 한 번만 표시한다.
<자격증> 테이블을 대상으로 검색한다.
'경력'이 3 이상인 자료만을 대상으로 검색한다.

[문제 5]

```
SELECT 이름, 재직년도, 기본급
FROM 사원
WHERE 이름 NOT IN (
    SELECT 이름
    FROM 자격증);
```

'이름', '재직년도', '기본급'을 표시한다.
<사원> 테이블을 대상으로 검색한다.
<자격증> 테이블에 '이름'이 NOT IN 다음에 쓰인 하위 질의의 결과에 없는 자료만을 대상으로 한다.
'이름'을 표시한다.
<자격증> 테이블에서 검색한다. 즉, <자격증> 테이블의 '이름'을 표시한다.

[문제 6]

```
SELECT 이름
FROM 자격증
GROUP BY 이름
HAVING COUNT(*) >= 2;
```

'이름'을 표시한다.
<자격증> 테이블을 대상으로 검색한다.
'이름'을 기준으로 그룹을 지정한다.
그룹 중 개수가 2개 이상인 자료만을 표시한다. 즉 자격증이 2개 이상인 사원의 '이름'을 검색한다.

[문제 7]

```
CREATE VIEW 3학년학생
AS SELECT *
FROM 학생
WHERE 학년 = 3
WITH CHECK OPTION;
```

생성한 뷰의 이름은 <3학년학생>이다.
모든 속성을 가져온다.
<학생> 테이블에서 속성을 가져온다.
'학년'이 3인 학생만을 대상으로 한다.
생성된 <3학년학생> 뷰에 갱신이나 삽입 연산 수행 시 뷰의 정의 조건(학년이 3학년)을 위반하면 실행을 거절한다.

[문제 8]

```
SELECT 학교, 학교명, 지역
FROM 학교 LEFT OUTER JOIN 학과 ON 학교.학교명 = 학과.학교명
WHERE 학과명 Is Null;
```

'학교명'과 '지역'을 표시한다. '학교명' 속성은 두 테이블에 모두 있으므로 테이블명을 지정해야 한다. '학과.학교명'을 입력해도 결과는 같다.
<학교> 테이블의 '학교명'과 <학과> 테이블의 '학교명'으로 JOIN하되, <학교> 테이블(왼쪽 테이블)의 튜플들은 <학과> 테이블의 연관성과 관계없이 모두 표시하는 LEFT JOIN이다.
'학과명'이 NULL인, 즉 내용이 입력되지 않은 튜플만을 대상으로 검색한다. '학과명' 대신 <학과> 테이블에 있는 다른 필드로 지정해도 결과는 같다

[문제 9]

① Left([주소],2) ② Month([주문날짜])=5

```
SELECT Left([주소],2) AS 지역, Sum(수량) AS 수량합계
FROM 주문현황, 회원관리
WHERE 주문현황.고객ID = 회원관리.고객ID
AND Month([주문날짜])=5
GROUP BY Left([주소],2);
```

'주소'의 왼쪽 두 글자를 '지역'으로, '수량'의 합계를 '수량합계'로 표시한다.

〈주문현황〉과 〈회원관리〉 테이블을 대상으로 검색한다.

〈주문현황〉 테이블의 '고객ID'와 〈회원관리〉 테이블의 '고객ID'가 같고 '주문날짜'의 월이 '5'인 주문만을 대상으로 검색한다.

'주소'의 왼쪽 두 글자를 기준으로 그룹을 지정한다.

[문제 10]

```
CREATE VIEW 강좌교수(강좌명, 강의실, 수강제한인원, 교수이름)
AS SELECT 강좌명, 강의실, 수강인원, 이름
FROM 강좌, 교수
WHERE 강좌.교수번호 = 교수.교수번호;
```

생성한 뷰의 이름은 〈강좌교수〉로, '강좌명', '강의실', '수강제한인원', '교수이름'으로 속성을 표시한다.

'강좌명', '강의실', '수강인원', '이름' 속성을 가져온다.

〈강좌〉 테이블과 〈교수〉 테이블에서 속성을 가져온다.

〈강좌〉 테이블의 '교수번호'와 〈교수〉 테이블의 '교수번호'가 같은 자료만을 대상으로 한다.

※ 테이블에서 불러온 속성명을 뷰에서 그대로 사용하지 않을 경우 사용하고 싶은 속성명을 뷰 이름 옆에 입력합니다.

[문제 11]

```
SELECT 예금번호, 도시
FROM 지점, 예금계좌
WHERE 지점.지점명 = 예금계좌.지점명
ORDER BY 예금번호 DESC;
```

〈검색 결과〉에 '예금번호'와 '도시'가 표시되어 있으므로, SELECT 다음에 '예금번호, 도시'를 지정한다.

문제에서 〈지점〉 테이블과 〈예금계좌〉 테이블을 '지점명'으로 조인하라고 했으므로 FROM 다음에 '지점, 예금계좌'를, WHERE 다음에 '지점.지점명 = 예금계좌.지점명'을 지정한다.

〈검색 결과〉가 '예금번호'를 기준으로 내림차순 정렬되어 있으므로, ORDER BY 다음에 '예금번호 DESC'를 지정한다.

[문제 12]

① commit ② rollback ③ grant ④ revoke ⑤ cascade

[문제 13]

GRANT SELECT ON 강좌 TO 홍길동;

[문제 14]

GRANT ALL ON 학생 TO 홍길동 WITH GRANT OPTION;

[문제 15]

REVOKE INSERT ON 교수 FROM 박문수;

[문제 16]

REVOKE SELECT ON 수강 FROM 박문수 CASCADE;

- ※ 박문수에게 부여된 <수강> 테이블에 대한 SELECT 권한을 취소하면 박문수가 다른 사람에게 SELECT 권한을 부여할 수 있는 권한도 함께 취소된다. 자신에게 권한이 없다면 해당 권한을 다른 사람에게 부여할 수 없기 때문이다.
- ※ 박문수에게 부여된 <수강> 테이블에 대해 SELECT 권한은 유지하면서 다른 사람에게 부여할 수 있는 권한만 제거하려면 다음과 같이 작성한다.
REVOKE GRANT OPTION FOR SELECT ON 수강 FROM 박문수;

[문제 17]

SELECT 상호, 총액	'상호'와 '총액'을 표시한다.
FROM 거래내역	<거래내역> 테이블을 대상으로 검색한다.
WHERE 총액 IN ('총액'과 IN 다음에 쓰인 하위 질의의 결과와 같은 거래처를 대상으로 한다.
SELECT MAX(총액)	'총액' 중 가장 큰 값을 표시한다.
FROM 거래내역);	<거래내역> 테이블에서 검색한다. 즉, <거래내역> 테이블에서 '총액' 중 가장 큰 값을 표시한다.

※ WHERE절에는 MAX, SUM, AVG 등과 같은 집계 함수를 사용할 수 없으므로 'WHERE 총액 = MAX(총액)'과 같이 작성할 수 없습니다.

[문제 18]

1. 삭제

DELETE	삭제하라.
FROM 상품	<상품> 테이블을 대상으로 하라.
WHERE 제품코드 = 'P-20';	'제품코드'가 "P-20"인 상품만을 대상으로 하라.

2. 삽입

INSERT	삽입하라.
INTO 상품	<상품> 테이블에 삽입하라.
VALUES ('P-20', 'PLAYER', 8800, 6600);	"P-20", "PLAYER", 8800, 6600을 <상품> 테이블의 각 필드에 삽입하라.





Section 077

[문제 1]

- ① 삽입 이상(Insertion Anomaly) ② 삭제 이상(Deletion Anomaly) ③ 갱신 이상(Update Anomaly)

- 삽입 이상(Insertion Anomaly)이란 테이블에 데이터를 삽입할 때, 의도와는 상관없이 원하지 않는 값들로 인해 삽입할 수 없는 현상을 의미한다. <주문> 테이블의 기본키가 주문번호+부품번호이기 때문에 부품번호를 삽입할 때도 반드시 '주문번호'가 있어야 가능하다.
- 삭제 이상(Deletion Anomaly)이란 테이블에서 한 튜플을 삭제할 때, 의도와는 상관없는 값들도 함께 삭제되는 연쇄 삭제가 일어나는 현상을 의미한다.
- 갱신 이상(Update Anomaly)이란 테이블에서 튜플에 있는 속성값을 갱신할 때 일부 튜플의 정보만 갱신되어 정보에 불일치성이 생기는 현상이다.

[문제 2]

- ① 원인 : 테이블에서 일부 속성들의 종속으로 인해 데이터의 중복이 발생하기 때문에 이상이 발생함
 ② 종류 : 삽입 이상(Insertion Anomaly), 삭제 이상(Deletion Anomaly), 갱신 이상(Update Anomaly)

Section 078

[문제 1]

- ① 함수적 종속 ② 학번 → 학과

[문제 2]

- ① 부품번호 ② 거래처지역

- 주문번호, 부품번호는 식별자이다 : 거래처코드, 거래처지역, 부품가격, 주문수량, 주문날짜는 기본키인 (주문번호, 부품번호)에 의해 함수적으로 종속된다.
- 주문번호는 주문날짜를 결정한다 : 주문번호 '1518'은 항상 주문날짜가 '16/10/8', 주문번호 '1607'은 항상 주문날짜가 '16/10/22', 주문번호 '1729'는 항상 주문날짜가 '16/11/03'이다.
- 부품번호는 부품가격을 결정한다 : 부품번호 '100'은 항상 부품가격이 '1000'이고, 부품번호 '200'은 항상 부품가격이 '500'이다.
- 거래처코드는 거래처지역을 결정한다 : 거래처코드 'A01'은 항상 거래처지역이 '서울'이고, 거래처코드 'A04'는 거래처지역이 항상 '부산'이다.

[문제 3]

- ① 결정자(Determinant) ② 종속자(Dependent)
 ③ 어떤 애트리뷰트(속성)가 기본키에 대해 완전히 종속적인 경우 완전 함수 종속이라고 한다.

Section 079

[문제 1]

- ① 정규화(Normalization) ② 비정규화(Denormalization, 역정규화)

[문제 2]

- ① 기본키 ② 부분 함수적 종속(성) ③ 완전 함수적 종속(성) ④ 이행적 함수적 종속(이행적 종속성) ⑤ 후보키

[문제 3]

- ① 제 3정규화 ② 이행적 함수적 종속 제거

<주문목록> 테이블에는 다음과 같은 함수적 종속이 존재한다.

- 주문번호 → 고객아이디, 주소
- 고객아이디 → 주소

고객아이디와 주소가 기본키인 주문번호에 대해 완전 함수적 종속이므로 제 2정규형이다. 그러나 <주문목록> 테이블에서 고객아이디가 주문번호에 함수적 종속이고, 주소가 고객아이디에 함수적 종속이므로 주소는 기본키인 주문번호에 대해 이행적 함수적 종속을 만족한다. 즉 주문번호 \rightarrow 고객아이디이고, 고객아이디 \rightarrow 주소이므로 주문번호 \rightarrow 주소는 이행적 함수적 종속이 된다. 따라서 <주문목록> 테이블은 제 3정규형이 아니다. 문제의 <정규화 결과>는 <주문목록> 테이블에서 이행적 함수적 종속(즉 주문번호 \rightarrow 주소)을 제거하여 <주문> 테이블과 <고객> 테이블로 무손실 분해한 것이다.

[문제 4]

• 정규화 사유

<제품납품> 테이블에는 다음과 같은 함수적 종속이 존재한다.

- 납품번호, 제품번호 \rightarrow 업체번호, 업체명, 납품수량
- 납품번호 \rightarrow 업체번호, 업체명

<제품납품> 테이블에는 기본키인 (납품번호, 제품번호)에 완전 함수적 종속이 되지 않는 속성이 존재한다. 업체번호와 업체명은 납품번호에 의해서도 결정될 수 있으므로 기본키에 대해 완전 함수적 종속이 아닌 부분 함수적 종속이다. 따라서 <제품납품> 테이블은 제 2정규형이 아니다.

• 정규화 결과

<제품납품> 테이블에서 기본키의 일부인 납품번호에 함수적 종속되는 부분 함수적 종속을 제거하기 위해 속성들을 분리하여 제 2정규형을 만들면 다음과 같다.

<납품목록>

납품번호	제품번호	납품수량
K102	P7720	100
J116	P7620	150
K217	P0098	300
A548	P0098	250
K102	P0892	100
K217	P7720	250

<주문>

납품번호	업체번호	업체명
K102	214	대한기계
J116	712	모연전자
K217	879	삼일제조
A548	523	효진항공

[문제 5]

제 4정규화

(수강과목, 담임교수)에 대응하는 교재 속성의 다치 종속성

(수강과목, 담임교수)에 대응하는 교재 속성 중 'D/B 일반'은 수강과목이 모두 '데이터베이스'로 종속되지만 담임교수는 '홍길동', '박찬호'와 같이 여러 개에 대응되므로 담임교수 속성에는 무관하다.

(수강과목, 교재)에 대응하는 담임교수 속성의 다치 종속성

(수강과목, 교재)에 대응하는 담임교수 속성 중 '홍길동'은 수강과목이 모두 '데이터베이스'로 종속되지만 교재는 'D/B 일반', 'SQL 개념'과 같이 여러 개에 대응되므로 교재 속성에는 무관하다.

※ 다중 값 종속(다치 종속성) : A, B, C 3개의 속성을 가진 테이블 R에서 어떤 복합 속성(A, C)에 대응하는 B 값의 집합이 A 값에만 종속되고 C 값에는 무관하며, B는 A에 다중 값 종속이라 하고 $A \twoheadrightarrow B$ 로 표기함

예상문제은행

4장

정답

정규화



[문제 1]

- ① : 도메인을 원자 값만으로 구성
- ② : 부분적 함수적 종속 제거

- ③ : 이행적 함수적 종속 제거
- ④ : 결정자이면서 후보키가 아닌 것 제거
- ⑤ : 다치 종속성
- ⑥ : 조인 종속성 이용

[문제 2]

제 3정규형

모든 도메인이 원자 값이므로 제 1정규형을 만족하고, 키가 아닌 모든 속성들이 기본키에 대해 완전 함수 종속이므로 제 2정규형도 만족한다. 그리고 이행적 함수적 종속 관계도 제거 되었으므로 제 3정규형까지 만족한다.

[문제 3]

$A \rightarrow B$

[문제 4]

㉠, ㉡, ㉢, ㉤

BCNF를 만족하려면 '도, 부, 이, 결'의 조건을 모두 만족해야 한다. 즉 도메인이 원자 값이어야 하고, 부분적 함수적 종속 제거, 이행적 함수적 종속 제거, 결정자이면서 후보키가 아닌 것도 제거되어야 한다.

[문제 5]

- ① 제 1정규화 ② 도메인이 원자 값만을 갖도록 다중 값 속성을 제거한다.

하나의 학번이 여러 개의 수강과목을 가지고 있다. 이러한 다중 값 속성을 제거하여 릴레이션의 모든 속성 값이 원자 값만을 갖도록 하는 것이 제 1정규화이다.

[문제 6]

- ① 삽입 이상(Insertion Anomaly)
 ② <수강> 테이블에 수강 정보를 입력하기 위해서는 기본키인 학번과 과목번호를 모두 입력해야 하는데 수강 신청서에 학번이 기입되지 않았기 때문에 삽입 이상이 발생한다.

삽입 이상(Insertion Anomaly)이란 테이블에 데이터를 삽입할 때, 의도와는 상관없이 원하지 않는 값들로 인해 삽입할 수 없는 현상을 의미한다.

[문제 7]

- ① 생산자 ② 상품명 ③ 상품가격

- 주문번호, 생산번호는 식별자이다 : 상품명, 상품가격, 주문수량, 생산자는 기본키인 (주문번호, 생산번호)에 의해 함수적으로 종속된다.
- 생산번호는 생산자, 상품명, 상품가격을 결정한다 : 생산번호 'P01'은 항상 생산자가 '홍길동'이고 항상 상품명 '사과'이며 항상 상품가격이 '500'이다.
- 상품명은 상품가격을 결정한다 : 생산번호에 관계없이 상품명 '배'의 상품가격은 항상 '1,000'이고 상품명 '오이'의 상품가격은 항상 '200'이다.

[문제 8]

• 정규화 사유

<부품구입> 릴레이션에는 다음과 같은 함수적 종속이 존재한다.

- | |
|--|
| <ul style="list-style-type: none"> • 부품번호 → 구입처, 구입연락처 • 구입처 → 구입연락처 |
|--|

<부품구입> 릴레이션에서 구입처가 부품번호에 함수적 종속이고, 구입연락처가 구입처에 함수적 종속이므로 구입연락처는 기본키인 부품번호에

대해 이행적 함수적 종속을 만족한다. 즉 부품번호 → 구입처이고, 구입처 → 구입연락처이므로 부품번호 → 구입연락처는 이행적 함수적 종속이 된다. 따라서 〈부품구입〉 릴레이션은 이행적 함수적 종속을 제거하여 제 3정규형으로 만들어야 한다.

• 정규화 결과

〈부품구입〉 릴레이션에서 이행적 함수적 종속(즉 부품번호 → 구입연락처)을 제거하여 〈부품구입처〉 릴레이션과 〈구입처정보〉 릴레이션으로 무손실 분해하면 다음과 같다.

〈부품구입처〉		〈구입처정보〉	
부품번호	구입처	구입처	구입연락처
BO-4185	영풍기계	영풍기계	031-478-5231
BO-3345	제일공구	제일공구	02-345-6789
BD-4785	한삼전자	한삼전자	051-733-4747
BC-6231	제일공구		
BC-2377	한삼전자		

[문제 9]

- ① BCNF ② 결정자이지만 후보키가 아닌 속성(관리자 → 참고명) 제거

• 〈참고정보〉 릴레이션에는 다음과 같은 함수적 종속이 존재한다.

- (물류번호, 참고명) → 관리자
- 관리자 → 참고명

〈참고정보〉 릴레이션에는 결정자이지만 후보키가 아닌 속성이 존재한다. 즉 함수적 종속 '관리자 → 참고명'에서 관리자는 결정자이지만 〈참고정보〉 릴레이션에서 후보키가 아니다. 왜냐하면 관리자 '김원중'은 항상 참고명이 '파주'이고 관리자 '김예중'은 항상 참고명이 '대구'로 관리자가 참고명을 종속하지만 관리자 속성에는 중복된 값이 있어 관리자 속성만으로는 레코드를 유일하게 식별할 수 없기 때문이다. 그러므로 〈참고정보〉 릴레이션은 BCNF가 아니다. BCNF는 릴레이션의 결정자가 모두 후보키가 되는 정규형이다.

• 문제에 제시된 내용은 〈참고정보〉 릴레이션에서 BCNF를 만족하지 못하게 하는 속성(즉 관리자 → 참고명)을 분리하여 〈물류관리〉 릴레이션과 〈참고관리〉 릴레이션으로 무손실 분해하는 정규화 과정이다.

[문제 10]

- ① 제 2정규화 ② 부분 함수적 종속 제거 ③ 제 3정규화 ④ 이행적 함수적 종속 제거

• 〈상품주문〉 릴레이션에는 다음과 같은 함수적 종속이 존재한다.

- (주문번호, 생산번호) → 상품명, 상품가격, 주문수량, 생산자
- 생산번호 → 생산자, 상품명, 상품가격
- 상품명 → 상품가격

〈상품주문〉 릴레이션에는 기본키인 (주문번호, 생산번호)에 완전 함수적 종속이 되지 않는 속성이 존재한다. 즉 주문수량은 기본키에 대해 완전 함수적 종속이지만 생산자, 상품명, 상품가격은 생산번호에 의해서도 결정될 수 있으므로 기본키에 대해 완전 함수적 종속이 아니다. 따라서 부분 함수적 종속을 제거하는 제 2정규화 과정이 필요하다.

• 〈상품주문〉 릴레이션에서 생산번호에 함수적 종속이 되는 생산자, 상품명, 상품가격 속성을 분리(즉 부분 함수적 종속을 제거)하여 〈주문〉 릴레이션과 〈생산〉 릴레이션으로 무손실 분해한 것이다.

• 〈생산〉 릴레이션에는 다음과 같은 함수적 종속이 존재한다.

- 생산번호 → 상품명, 상품가격, 생산자
- 상품명 → 상품가격

〈생산〉 릴레이션에서 상품명이 생산번호에 함수적 종속이고, 상품가격이 상품명에 함수적 종속이므로 상품가격은 기본키인 생산번호에 대해 이행적 함수적 종속을 만족한다. 즉 생산번호 → 상품명이고, 상품명 → 상품가격이므로 생산번호 → 상품가격이다. 따라서 〈생산〉 릴레이션은 이행적 함수적 종속을 제거하는 제 3정규화 과정이 필요하다.

• 〈생산〉 릴레이션에서 이행적 함수적 종속(즉 생산번호 → 상품가격)을 제거하여 〈생산자〉 릴레이션과 〈상품〉 릴레이션으로 무손실 분해한 것이다.

[문제 11]

① 제 4정규화 ② 다치 종속성 제거

- 〈납품〉 릴레이션에서 〈납품처, 담당사원〉에 대응하는 제품명 속성의 집합이 납품처에만 종속되고 담당사원 속성에는 무관하므로 제품명 속성은 납품처에 다치 종속성이 존재한다. 또한 〈납품처, 제품명〉에 대응하는 담당사원 속성의 집합이 납품처에만 종속되고 제품명 속성에는 무관하므로 담당사원 속성은 납품처에 다치 종속성이 존재한다.
 - 문제에 제시된 내용은 〈납품〉 릴레이션에서 다치 종속성을 제거하여 〈납품담당〉과 〈납품제품〉 릴레이션으로 무손실 분해한 정규화 과정이다.
 - 〈납품처, 담당사원〉에 대응하는 제품명 속성의 다치 종속성
(납품처, 담당사원)에 대응하는 제품명 속성 중 '베어링'은 납품처가 모두 '대명금속'으로 종속되지만 담당사원은 '김봉연', '김정영'과 같이 여러 개에 대응되므로 담당사원 속성에는 무관하다.
 - 〈납품처, 제품명〉에 대응하는 담당사원 속성의 다치 종속성
(납품처, 제품명)에 대응하는 담당사원 속성 중 '김봉연'은 납품처가 모두 '대명금속'으로 종속되지만 제품명은 '베어링', '나사'와 같이 여러 개에 대응되므로 제품명 속성에는 무관하다.
- ※ 다중 값 종속(다치 종속성) : A, B, C 3개의 속성을 가진 테이블 R에서 어떤 복합 속성(A, C)에 대응하는 B 값의 집합이 A 값에만 종속되고 C 값에는 무관하면, B는 A에 다중 값 종속이라 하고 $A \twoheadrightarrow B$ 로 표기함





Section 080

[문제 1]

시스템 카탈로그(System Catalog)

[문제 2]

① - ⑥ ② - ⑤ ③ - ① ④ - ③ ⑤ - ④ ⑥ - ②

Section 081

[문제 1]

인덱스(Index)

[문제 2]

① 컬럼(속성) ② 행 이주(Row Migration)

Section 082

[문제 1]

- 원자성(Atomicity) : 트랜잭션의 연산은 데이터베이스에 모두 반영되든지 아니면 전혀 반영되지 않아야 한다.
- 일관성(Consistency) : 트랜잭션 수행이 성공적으로 완료되면 언제나 일관성 있는 데이터베이스 상태로 변환한다.
- 독립성(Isolation) : 한 트랜잭션이 데이터를 갱신하는 동안 이 트랜잭션이 완료되기 전에는 갱신중인 데이터를 다른 트랜잭션들이 접근하지 못하도록 해야 한다.
- 영속성(Durability) : 트랜잭션의 실행이 성공적으로 실행 완료된 후에는 시스템에 오류가 발생하더라도 트랜잭션에 의해 변경된 내용은 계속 보존되어야 한다.

[문제 2]

① 실패(Failed) ② 철회(Aborted)

- 실패(Failed) : 트랜잭션 실행중에 오류가 발생하여 중단된 상태
- 철회(Aborted) : 트랜잭션이 비정상적으로 종료되어 ROLLBACK 연산을 수행한 상태

Section 083

[문제 1]

① 시스템(System) ② 덤프(Dump) ③ 검사점(Check Point)

Section 084

[문제 1]

갱신 분실(Lost Update)

갱신 분실(Lost Update)은 2개 이상의 트랜잭션이 같은 데이터를 공유하여 갱신할 때 생기는 문제이다. 트랜잭션 T₁이 ③, ④ 단계에서 X의 값을 읽어 1000 증가시킨 후 트랜잭션 T₂가 ③, ④ 단계에서 X의 값을 다시 읽어 다시 0.9배로 증가시켰다. 그리고 트랜잭션 T₁은 ⑤ 단계에서 변경시킨 X의 값을 기록하였고, 트랜잭션 T₂도 ① 단계에서 변경시킨 X의 값을 기록하였다. 그 결과 트랜잭션 T₁의 갱신 결과를 트랜잭션 T₂가 겹쳐 기록하였기 때문에 트랜잭션 T₁은 데이터베이스에 아무런 영향을 주지 못했다. 즉 두 개 이상의 트랜잭션이 같은 데이터를 공유하여 갱신할 때 갱신 결과의 일부가 없어지는 갱신 분실이 발생하여 X의 최종 값은 9000이 된다.

T ₁ 결과	T ₁	T ₂	T ₂ 결과
X			X
10000 11000	Ⓐ read(X); Ⓑ X = X + 1000;	Ⓒ read(X); Ⓓ X = X * 0.9;	10000 9000
11000	Ⓔ write(X);	Ⓙ write(X);	9000

[문제 2]

타임 스탬프(Time Stamp)

[문제 3]

- Growing(확장) 단계 : 트랜잭션이 잠금만 수행할 수 있고 잠금 해제는 수행할 수 없는 단계이다.
- Shrinking(축소) 단계 : 트랜잭션이 잠금 해제만 수행할 수 있고 잠금은 수행할 수 없는 단계이다.

[문제 4]

- ① 병행 제어 : 다중 프로그램의 이점을 활용하여 동시에 여러 개의 트랜잭션을 병행 수행할 때 실행되는 트랜잭션들이 데이터베이스의 일관성을 파괴하지 않도록 트랜잭션 간의 상호 작용을 제어하는 기술이다.
- ② 로킹 기법 : 주요 데이터의 액세스를 상호 배타적으로 하는 것으로, 트랜잭션들이 어떤 로킹 단위를 액세스하기 전에 Lock(잠금)을 요청해서 Lock이 허락되어야만 그 로킹 단위를 액세스할 수 있도록 하는 기법이다.

Section 085

[문제 1]

개인키 암호화(Private Key Encryption) 기법

[문제 2]

공개키 암호화(Public Key Encryption) 기법

Section 086

[문제 1]

튜닝(Tuning)

[문제 2]

선택성(Selectivity) 또는 분포도

[문제 3]

- ① 클러스터드(Clustered)
- ② 년클러스터드(Non-Clustered)

Section 087

[문제 1]

- ① 메소드(Method)
- ② 클래스(Class)

[문제 2]

- 오버로딩(Overloading) : 하나의 메소드 이름으로 다른 파라미터를 지정하여 사용하는 기법으로, 메소드 사용의 효율성을 높인 기법이다.
- 오버라이딩(Overriding) : 슈퍼 클래스에 있는 메소드를 서브 클래스에서 상속받아 다른 기능을 수행하는 메소드로 재정의하는 기법이다.

Section 088

[문제 1]

ODBC Driver

[문제 2]

- ① Drill-down ② Pivoting ③ Slicing ④ Dicing

[문제 3]

OLAP(Online Analytical Processing)

[문제 4]

- ① 데이터 마이닝 : 데이터 웨어하우스에 저장된 데이터 집합에서 사용자의 요구에 따라 유용하고 가능성 있는 정보를 발견하기 위한 기법이다.
② 군집화 : 상호 간에 유사한 특성을 갖는 데이터들을 집단화(Clustering)하는 방법이다.

예상문제는 행

5장

정답

데이터베이스 기타 실무 응용



[문제 1]

506

- 차수가 3인 트리의 노드는 기본적으로 다음과 같이 구성된다.

차수	포인터	키 값	주소	포인터	키 값	주소	포인터
0	*	403	50	*	()	70	*

- 그러므로 <학생> 릴레이션에서 주소가 70인 레코드의 키 값인 학번을 찾으면 된다.

[문제 2]

- 오류1 : <주문> 릴레이션의 주문번호 '027'에 대한 주문 내역이 <판매> 릴레이션에 추가되어 있지 않으므로 원자성의 특성에 위배된다.
• 오류2 : <주문> 릴레이션에서 주문번호 '034'인 상품의 수량이 '1'이므로 <판매> 릴레이션에서 주문번호 '034'인 상품의 수량도 '1'이 되어야 하지만 '1'이므로 일관성의 특성에 위배된다.

- 원자성(Atomicity) : 트랜잭션 연산이 데이터베이스에 모두 반영되든지 아니면 전혀 반영되지 않아야 함
• 일관성(Consistency) : 트랜잭션 수행이 성공적으로 완료되면 언제나 일관성 있는 데이터베이스 상태로 변환되어야 함

[문제 3]

- ① 173 ② 김유신 ③ 26 ④ 452

- ROLLBACK은 하나의 트랜잭션 처리가 비정상적으로 종료되어 데이터베이스의 일관성이 깨졌을 때 이 트랜잭션의 일부가 정상적으로 처리되었더라도 트랜잭션의 원자성을 실현하기 위해 이 트랜잭션이 수행한 모든 연산을 취소하는 연산이다.
• ROLLBACK 연산의 실행으로 인해 <2차 합격자> 데이터를 <신입사원> 릴레이션에 추가하는 트랜잭션이 모두 취소되었으므로 <신입사원> 릴레이션에 남은 레코드는 사원번호가 173, 이름이 '김유신', 나이가 26, 자원번호가 452인 레코드이다.

[문제 4]

- ① 63,000
② 83,000

T ₁ 결과		T ₁	T ₂	T ₂ 결과	
X	Y			X	Y
60,000		lock-S(X); read(X); X = X + 1000; lock-X(X); write(X); unlock(X);	lock-S(Y); read(Y); Y = Y + 2000; lock-X(Y); write(Y); unlock(Y);		
61,000					
61,000					
	82,000	lock-S(Y); read(Y); Y = Y + 1000; lock-X(Y); write(Y); unlock(Y);	lock-S(X); read(X); X = X + 2000; lock-X(X); write(X); unlock(X);		80,000
	83,000				82,000
	83,000				82,000
				61,000	
				63,000	
				63,000	

[문제 5]

클래스 계층 구조를 형성하는 서브 클래스와 슈퍼 클래스 간에 존재하는 계승의 개념이다.

[문제 6]

- 위치 투명성(Location Transparency) : 접근하려는 데이터베이스의 실제 위치를 알 필요 없이 단지 데이터베이스의 논리적인 명칭만으로 접근할 수 있다.
- 중복 투명성(Replication Transparency) : 동일한 데이터가 여러 곳에 중복되어 있더라도 사용자는 마치 하나의 데이터만 존재하는 것처럼 사용할 수 있고, 시스템은 자동으로 여러 데이터에 대한 작업을 수행한다.
- 병행 투명성(Concurrency Transparency) : 분산 데이터베이스와 관련된 다수의 트랜잭션들이 동시에 실행되더라도 그 트랜잭션들의 수행 결과는 서로 영향을 받지 않는다.
- 장애 투명성(Failure Transparency) : 트랜잭션, DBMS, 네트워크, 컴퓨터 장애에도 불구하고 트랜잭션은 정확하게 수행된다.

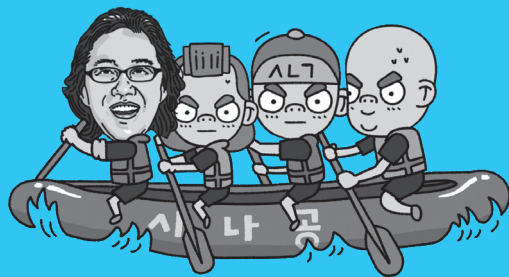
m · e · m · o



3 과목

정답 및 해설

| 업무 프로세스 실무 일반 |





Section 089

[문제 1]

CSF(Critical Success Factor)

[문제 2]

5-FORCE

[문제 3]

① 7S ② Shared Value

[문제 4]

SWOT 분석

[문제 5]

측정 → 분석 → 향상 → 관리

[문제 6]

특정 분야에서 우수한 상대를 모델로 하여 자기 기업과의 성과 차이를 비교 분석하고, 단점을 극복하기 위해 상대의 앞선 운영 프로세스를 배우면서 꾸준히 노력하여 자기 혁신을 하는 경영 기법이다.

Section 090

[문제 1]

As-Is

[문제 2]

BPM(Business Process Management)

[문제 3]

KPI(Key Performance Indicator)

[문제 4]

RTE(Real-Time Enterprise)

[문제 5]

업무 프로세스 재설계(BPR; Business Process Reengineering)

Section 091

[문제 1]

ERP(Enterprise Resource Planning)

[문제 2]

MRP(Material Requirements Planning)

[문제 3]

① SEM(Strategic Enterprise Management) ② ABM(Activity Based Management) ③ BSC(Balanced Score Card)

[문제 4]

MOT(Management Of Technology, 기술 경영)

Section 092

[문제 1]

① SCM(Supply Chain Management) ② SCP(Supply Chain Planning) ③ SCE(Supply Chain Execution)

[문제 2]

CRM(Customer Relationship Management)

[문제 3]

가치 사슬(Value Chain)

[문제 4]

VOC(Voice Of Customer)

[문제 5]

기업 성과 관리(CPM; Corporate Performance Management)

[문제 6]

PLM(Product Lifecycle Managemnet)

Section 093

[문제 1]

EAI(Enterprise Application Integration)

[문제 2]

KMS(Knowledge Management System)

[문제 3]

BRMS(Business Rule Management System)

[문제 4]

EIP(Enterprise Information Portal)

[문제 5]

① 의사 결정 지원 시스템(DSS; Decision Support System) ② 전략 정보 시스템(SIS; Strategic Information System)

[문제 6]

EA(Enterprise Architecture, 전사적 아키텍처)

[문제 7]

① 지식 경영 최고 책임자(CKO; Chief Knowledge Officer) ② 전자 문서 관리 시스템(EDMS; Electronic Document Management System)
③ 지식 공동체(COP; Communities of Practice) ④ 업무 분업 구조(WBS; Work-Breakdown Structure)

Section 094

[문제 1]

ITIL(Information Technology Infrastructure Library)

[문제2]

BPO(Business Process Outsourcing)

[문제3]

SLA(Service Level Agreement)

Section 095

[문제 1]

CALS(Commerce At Light Speed)

[문제 2]

- ① B2B ② C2C ③ B2G

Section 096

- ① 사무국 ② 서면평가

Section 097

- ① 기술참조모델(TRM) ② 상위5% ③ 하위5% ④ 연봉인상 제한대상 ⑤ 업적자료표

Section 098

- ① ITA(정보기술 아키텍처) ② SWOT ③ 손망실변상심의요청 ④ 자산관리위원회 ⑤ 심의결과 확정

Section 099

- ① 대외보고서 재고조사 ② 대외보고서 배포 ③ 대외보고서 이관

예상문제은행

1장

정답

업무 프로세스



[문제 1]

WO

[문제 2]

고객(Customer), 경쟁사(Competitor), 자사(Company)

[문제 3]

BPR(Business Process Reengineering)

[문제 4]

PI(Process Innovation)

[문제 5]

- ① 인적사항 ② 담당부서 ③ 심사위원회 ④ 포상심사 ⑤ 선정결과서

[문제 6]

- ① 자기고과실시 ② 자기고과표 ③ 담당부서 ④ 고과 요청 ⑤ 고과 실시

[문제 7]

- ① 해외출장 타당성 협의 ② 여비산정 ③ 해외출장 내부결재 ④ 여비지출 결의 ⑤ 인사발령(직책자)

[문제 8]

B2C

[문제 9]

Escrow

[문제 10]

MDM(Master Data Management)

[문제 11]

- ① 대차대조표(Balance Sheet) ② 이익잉여금처분계산서(Statement of Retainer Earnings) ③ 현금흐름표(Statement of Cash Flows)
④ 안정성 ⑤ 매출채권 회전을

[문제 12]

- ① 종합소득 ② 분류과세소득 ③ 근로소득 ④ 교육비공제 ⑤ 종합과세

[문제 13]

SLM(Service Level Management)

[문제 14]

- ① 상품배송 ② 거래승인 ③ 승인요청 ④ 매입요청 ⑤ 대금입금

[문제 15]

SRM(Supplier Relationship Management)

[문제 16]

VRM(Vendor Relationship Management)

[문제 17]

- ① 수주 ② 출하지시 ③ 입고 ④ 입하 ⑤ 물류센터

[문제 18]

- ① 구축 ② 분석 ③ 설계 ④ GAP ⑤ 적용

[문제 19]

- ① 생산계획 ② 작업지시 ③ 생산실적 등록 ④ 생산입고 ⑤ 전사적 자원관리(ERP)

[문제 20]

프로세스(Process), 인력(People), 조직(Organization), 기술(Technology)

[문제 21]

ISO/IEC 20000

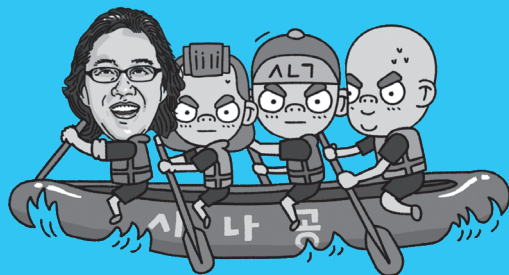
m · e · m · o



4 과목

정답 및 해설

| IT 신기술 동향 및 시스템 관리, 전산 영어 |





Section 100

[문제 1]

DES(Data Encryption Standard)

[문제 2]

PKI(Public Key Infrastructure)

[문제 3]

- ① 기밀성 : 시스템 내의 정보와 자원은 인가된 사용자에게만 접근이 허용되는 것을 의미한다.
- ② 무결성 : 시스템 내의 정보는 인가된 사용자만 수정이 가능한 것을 의미한다.
- ③ 가용성 : 인가받은 사용자는 언제라도 사용 가능한 것을 의미한다.

[문제 4]

양자 암호 키 분배(QKD; Quantum Key Distribution)

[문제 5]

해시(Hash)

[문제 6]

크로스 사이트 스크립팅(XSS; Cross Site Scripting)

Section 101

[문제 1]

RFID(Radio Frequency IDentification)

[문제 2]

DRM(Digital Rights Management)

[문제 3]

디지털 포렌식(Digital Forensics)

[문제 4]

WEP, Wired Equivalent Privacy

[문제 5]

OTP, One-Time Password, 일회용 패스워드

[문제 6]

CCL(Creative Commons License)

[문제 7]

PET(Privacy Enhancing Technology)

[문제 8]

- ① copyright(저작권) ② copyleft

[문제 9]

허니팟(honeypot)

[문제 10]

브로드 데이터(Broad Data)

[문제 11]

목표 복구 시간(RTO; Recovery Time Objective)

[문제 12]

RPO(Recovery Point Objective, 목표 복구 시점)

[문제 13]

IPsec(Internet Protocol security, IP 보안 프로토콜)

[문제 14]

업무 연속성 계획(BCP; Business Continuity Planning)

[문제 15]

트러스트존(TrustZone) 기술

[문제 16]

킬 스위치(Kill Switch)

Section **102**

[문제 1]

스니핑, Sniffing

[문제 2]

VoIP 보안 위협, VoIP Security Threat

[문제 3]

Watering Hole

[문제 4]

네트워크의 중간에서 남의 패킷 정보를 도청하는 해킹 유형의 하나로 수동적 공격에 해당한다.

[문제 5]

Back Door 또는 Trap Door

[문제 6]

Zero Day Attack(제로 데이 공격)

[문제 7]

Social Engineering(사회 공학)

[문제 8]

DDoS

[문제 9]

OWASP(The Open Web Application Security Project, 오픈 웹 애플리케이션 보안 프로젝트)

[문제 10]

스턱스넷(Stuxnet)

[문제 11]

살라미(salami)

[문제 12]

SQL injection(SQL 삽입) 공격

[문제 13]

Brute Force Attack(무작위 공격)

[문제 14]

Qshing(큐싱)

[문제 15]

다크 데이터(Dark Data)

[문제 16]

그레이웨어(grayware)

[문제 17]

블록체인(Blockchain)

[문제 18]

타이포스쿼팅(typosquatting)

[문제 19]

딥페이크(Deepfake)

예상문제는행

1장

정답

시스템 관리 및 보안



[문제 1]

- ① 인증(Authentication) ② 가용성(Availability) ③ 무결성(Integrity)
④ 기밀성(Confidentiality, 비밀성) ⑤ 접근 통제(Access Control) ⑥ 부인 방지(Non Repudiation)

[문제 2]

순차적 암호화(OPE, Order Preserving Encryption)

[문제 3]

- 가로막기 : 데이터의 정상적인 전달을 가로막아서 흐름을 방해하는 행위
- 가로채기 : 송신된 데이터가 수신지까지 가는 도중에 몰래 보거나 도청하여 정보를 유출하는 행위
- 수정 : 전송된 데이터를 원래의 데이터가 아닌 다른 내용으로 바꾸는 행위
- 위조 : 마치 다른 송신자로부터 데이터가 송신된 것처럼 꾸미는 행위

[문제 4]

백업 대상 데이터 중 변경되거나 증가된 데이터만을 대상으로 백업을 수행하는 방식

[문제 5]

SSL(Secure Sockets Layer)

[문제 6]

① 방화벽(Firewall) ② 캐시(Cache)

[문제 7]

Typosquatting

[문제 8]

Botnet

[문제 9]

APT(Advanced Persistent Threats)

[문제 10]

SMURFING

[문제 11]

Tvishing

[문제 12]

Pharming

[문제 13]

Ping of Death(죽음의 핑)

[문제 14]

DRM(Digital Rights Management)

원문 해석

(DRM)은 데이터의 안전한 배포를 활성화하거나 불법 배포를 방지하여 인터넷이나 기타 디지털 매체를 통해 유통되는 데이터의 저작권을 보호하기 위한 시스템이다. (DRM) 시스템은 보통 데이터를 암호화하여 인증된 사용자만이 접속할 수 있게 하거나, 디지털워터마크의 사용 또는 이와 유사한 방식으로 콘텐츠를 작성하여 콘텐츠가 제한 없이 보급되지 않도록 하여 지적재산권을 보호한다.

[문제 15]

① Advanced Persistent Threats ② Business Continuity Planning

[문제 16]

DLP(Data Leakage/Loss Prevention)

원문 해석

(DLP)는 내부정보 유출 방지 솔루션 중의 하나이다. 사내 직원이 사용하는 PC와 네트워크상의 모든 정보를 검색하고 사용자의 행위를 탐지·통제해 외부로의 유출을 막는다. 메신저, 웹메일, 웹하드, 프린터, USB 등 다양한 경로로 정보가 유출되는 것을 기록 및 통제하는 기술이다. PC나 서버 등 단말에서의 유출을 통제하는 기술을 엔드포인트 (DLP)라고 하며, 경계 부근의 네트워크 출구 지점에서 유출을 통제하는 기술을 네트워크 (DLP)라고 한다.

[문제 17]

① - c ② - a ③ - e ④ - b ⑤ - d

[문제 18]

RFID(Radio Frequency Identification)

원문 해석

RFID는 이론상 바코드 식별과 유사한 기술이다. RFID는 신호를 송신하는 전자기파 스펙트럼의 RF 부분에 있는 전자식 또는 정전기식 결합장치(coupling)를 사용한다. RFID 시스템은 무선 주파수를 읽어 처리장치로 정보를 전송하는 안테나와 송·수신기, 그리고 RF 회로가 내장된 통합 회로와 전송해야 할 정보를 담고 있는 트랜스폰더 또는 태그로 구성되어 있다.

[문제 19]

DES(Data Encryption Standard)

원문 해석

(DES)는 개인키를 사용하여 데이터를 암호화하는 방법으로서 널리 사용되고 있다. 1977년 미국의 국립표준기술연구에 의해 정부에서 또는 상업적으로 이용할 목적으로 설계된 것으로, 아주 복잡한 데이터 암호화 방법이다. (DES)는 미국방성에서 자료 암호화의 표준으로 채택하여 널리 사용되고 있다.

[문제 20]

① - c ② - a ③ - e ④ - b ⑤ - d

원문 해석

- ① 데이터베이스 시스템의 모든 관리와 운영에 대한 책임을 지고 있는 사람이나 그룹을 의미한다.
- ② 원저작물의 창작자가 저작물의 사용과 배포에 있어 일반적으로 제한된 시간 동안 배타적 권리를 인정하는 법적인 권리이다.
- ③ 숫자로 된 IP 주소를 사람이 이해하기 쉬운 문자 형태로 표현한 것이다.
- ④ 침입자로부터 컴퓨터 안전을 지키는 주요한 수단이다.
- ⑤ 사용자의 편의를 도모하는 동시에 시스템의 생산성을 높이기 위한 프로그램의 집합체로, 사용자와 컴퓨터 사이에서 중계자 역할을 한다.

[문제 21]

① Public Key Infrastructure ② Creative Commons License

[문제 22]

PET(Privacy Enhancing Technology)

원문 해석

(PET)는 컴퓨터 도구, 어플리케이션, 메커니즘 등이 온라인 서비스 또는 어플리케이션과 통합되었거나, 이 같은 서비스 또는 어플리케이션과 함께 사용되었을 때, 온라인 사용자가 이러한 서비스 또는 어플리케이션에 제공하여 취급되는 개인 식별 정보(PI)의 프라이버시를 보호하는데 사용할 수 있도록 하는 기술 일체에 대한 일반 용어이다.

(PET)는 다음과 같이 정의할 수 있다. (PET)이란 정보 시스템의 기능을 상실하지 않고 개인 데이터를 제거하거나 최소화함으로써 개인 데이터에 대한 원치 않거나 불필요한 가공을 방지하여 정보의 프라이버시를 보호하는 ICT 정책 시스템이다.

[문제 23]

① - a ② - d ③ - e ④ - c ⑤ - b

[문제 24]

DDoS(Distributed Denial of Service)

원문 해석

(DDoS)는 단일 시스템을 표적으로 훼손된(보통 트로이 목마에 감염된) 여러 시스템을 사용해 서비스 거부 공격(Denial of Service, DoS)을 유발하는 공격법이다. 표적이 된 시스템 및 해커의 제어 하에 분산 공격에 부당하게 사용된 모든 시스템이 (DDoS) 공격의 피해자가 된다.



Section 103

[문제 1]

AMR(Automatic Meter Reading)

[문제 2]

NFC(Near Field Communication)

[문제 3]

TEIN(Trans-Eurasia Information Network)

[문제 4]

지그비(ZigBee)

[문제 5]

스토리지 전용 네트워크(SAN; Storage Area Network)

[문제 6]

VOD(Video On Demand)

[문제 7]

MVNO

[문제 8]

NAS(Network Attached Storage, 망 부착 저장장치)

[문제 9]

GLONASS(글로나스)

[문제 10]

SDN(Software Defined Networking, 소프트웨어 정의 네트워킹)

[문제 11]

래드섹(RadSec) 프로토콜

[문제 12]

SON(Self Organizing Network, 자동 구성 네트워크)

Section 104

[문제 1]

IMS(Internet protocol Multimedia Subsystem)

[문제 2]

비트코인(Bitcoin)

[문제 3]

IoT(Internet of Things, 사물 인터넷)

[문제 4]

증강현실(Augmented Reality)

[문제 5]

웹바홀리즘(Webaholism)

[문제 6]

HVC

[문제 7]

IPTV(Internet Protocol TeleVision)

[문제 8]

RSS

[문제 9]

위키노믹스(Wikinomics)

[문제 10]

XML(eXtensible Markup Language)

[문제 11]

① IPv6(Internet Protocol version 6) ② 128

[문제 12]

Semantic Web(시맨틱 웹)

[문제 13]

웹 서비스 기술 언어(WSDL; Web Services Description Language)

[문제 14]

미라캐스트(Miracast)

Section **105**

[문제 1]

기가 파이(Giga Fi)

[문제 2]

클라우드 컴퓨팅(Cloud Computing)

[문제 3]

AllJoyn(올조인)

[문제 4]

MST(Magnetic Secure Transmission, 마그네틱 안전 전송)

[문제 5]

멤스(MEMS; Micro Electro Mechanical System)

[문제 6]

에지(EDGE; Enhanced Data rates for Global Evolution)

[문제 7]

고화질 멀티미디어 인터페이스(HDMI; High-Definition Multimedia Interface)

[문제 8]

컴패니언 스크린(Companion Screen)

[문제 9]

지능형 서비스 로봇(URC; Ubiquitous Robotic Companion)

[문제 10]

펄토셀 기지국(Femtocell Base Station)

[문제 11]

VHT(Very High Throughput)

[문제 12]

애드 혹 네트워크(Ad-hoc Network)

[문제 13]

네트워크 슬라이싱(Network Slicing)

Section **106**

[문제 1]

Nomophobia(노모포비아)

[문제 2]

GPU(Graphics Processor Unit, Graphics Processing Unit)

[문제 3]

타조(Tajo)

[문제 4]

AJAX(Asynchronous JavaScript and XML, 비동기 자바 스크립트와 XML)

[문제 5]

클라우드 소싱(Cloud Sourcing)

[문제 6]

커넥티드 카(Connected Car)

[문제 7]

Cresumer(크리슈머)

[문제 8]

저전력 블루투스 기술(BLE; Bluetooth Low Energy)

[문제 9]

디옵스(Devops)

[문제 10]

딥 러닝(Deep Learning)

[문제 11]

Infolust(정보 열광자, 인포러스트)

[문제 12]

베어퍼웨어(Vaporware, 증발품)

[문제 13]

EPG

[문제 14]

스마트 그리드, Smart Grid

[문제 15]

기계 학습, 머신 러닝, Machine Learning

예상문제는행

2장

정답

신기술 동향



[문제 1]

GIS(Geographic Information System)

원문 해석

지리적인 자료를 수집·저장·분석·출력할 수 있는 컴퓨터 응용 시스템이며, 모든 사물들의 위치 정보를 제공하기 위해 위성을 이용한다. 이것을 통해 우리는 지도에서 사물을 확인하는 단계를 넘어서서, 인터넷과 위성과 같은 다양한 미디어를 이용하여 지리 데이터를 수집·구축·분석·처리함으로써 고품질의 공간 정보를 생산하는 것이 가능하다.

[문제 2]

① Internet ② LAN ③ Topology ④ Protocol ⑤ Node

원문 해석

컴퓨터 및 통신 기술의 태동기에는 이 두 기술이 전혀 다른 주제로 취급되었다. 하지만 이러한 기술들이 발전함에 따라 두 가지가 병합되기 시작했다. (Internet)라고 하는 신기술의 혁신이 등장하고 인간의 삶 전체를 변화시켰다.

(Internet)는 컴퓨터와 네트워크가 결합된 데이터 통신 시스템이다. 물리적으로는, (Internet)는 통신 회선으로 연결된 여러 컴퓨터 또는 터미널로 구성된다. (Internet)는 사용자가 다양한 네트워크를 통해 정보에 대한 접근을 공유할 수 있도록 한다. 일부 네트워크는 전 세계에 걸쳐 있고 다른 네트워크는 비교적 작은 지역을 담당한다.

(Internet)에는 다음을 포함한 여러 종류가 있다.

- (LAN) : 컴퓨터들이 지리적으로 가까운 거리에 있다(예를 들어 같은 건물).
- WAN : 컴퓨터들이 멀리 떨어져 있으며 전화 회선이나 무선 전파로 연결된다.
- CAN(Compus Area Network) : 컴퓨터들이 캠퍼스 또는 군사 기지 등과 같은 제한된 지리적 영역 내에 있다.
- MAN(Metropolitan Area Network) : 소도시나 대도시를 위해 설계된 데이터 네트워크이다.
- HAN(Home Area Network) : 사용자의 디지털 기기들을 연결하는 사용자 가정 내의 네트워크이다.

이러한 종류 이외에도 여러 종류의 네트워크로 분류하는 데 다음과 같은 특성이 사용된다.

- (Topology) : 컴퓨터 시스템의 기하학적 배치
일반적인 (Topology)에는 버스, 스타 및 링이 포함된다.
- (Protocol) : 네트워크의 컴퓨터들이 통신을 하기 위해 사용하는 공통된 규칙 및 신호 집합을 정의한다. LAN에 가장 널리 사용되는 (Protocol) 중 하나를 이더넷이라고 한다. PC에서 널리 사용되는 또 다른 LAN (Protocol)은 IBM 토큰링 네트워크이다.
- 아키텍처 : 네트워크는 넓은 의미로 피어 투 피어 또는 클라이언트/서버 아키텍처로 분류할 수 있다.

네트워크상의 컴퓨터를 (Node)라고도 한다. 네트워크에서는 처리 위치이다. 일반적으로 (Node)는 컴퓨터일 수도 있고 프린터와 같은 다른 장치일 수 있다.

모든 (Node)에는 DLC(Data Link Control) 주소 또는 MAC(Media Access Control) 주소라고도 하는 고유한 네트워크 주소가 있다.

네트워크에 자원을 할당하는 컴퓨터 및 장치들을 서버라고 한다.

[문제 3]

DHCP(Dynamic Host Configuration Protocol)

원문 해석

TCP/IP 통신을 실행하기 위해 필요한 설정 정보를 자동적으로 할당하고 관리하기 위한 통신 규약이다. RFC 1541에서 규정하고 있으며, TCP/IP 환경의 통신망에서 IP 주소의 일률적인 관리 서비스를 제공한다. 이것을 제공하는 서버는 클라이언트의 요구에 따라 IP 주소를 할당한다.

[문제 4]

- ① Kernel ② Shell

원문 해석

① 운영체제에서 가장 핵심적인 역할을 한다. 메모리나 프로세서와 같은 자원을 관리하거나, 시스템이 원활히 운영되도록 제어하는 역할을 수행한다. 일반적으로 종료된 입 · 출력 연산과 같이 서비스에 대한 경쟁적인 모든 요청들을 처리하는 인터럽트 처리기, 프로그램들의 처리 시간을 어떤 순서대로 나눌지 결정하는 스케줄러, 그리고 각각의 프로세스에게 권한을 부여하는 관리자를 포함한다.

② 시스템과 사용자간의 인터페이스를 제공하는 명령줄 인터프리터이며, 커널과 함께 Unix의 주요 구성 요소 중 하나이다. DOS 운영체제에서 command. com 파일과 동일한 역할을 수행한다. 메모리에 적재되지 않고 외부 메모리에 파일 형태로 존재한다. 또한 명령 파이프라이닝을 지원하며, 입 · 출력 방향을 변경하는 것이 가능하다.

[문제 5]

Bluetooth

원문 해석

에릭슨, IBM, 인텔, 노키아, 도시바에 의해서 1998년에 설립된 단체에서 나온 무선 개인 영역 네트워크 기술이다. 이것은 점 대 점 또는 다중 점 간의 응용 프로그램을 지원하는 데이터나 디지털 음성 등의 단거리 전송을 위한 공개된 표준이다. 처음에 나온 응용 프로그램들은 핸드프리 방식의 헤드폰이나 자동차 오디오 시스템에 연결하기 위해 휴대폰에 장착되었다.

[문제 6]

OOP(Object-Oriented Programming)

원문 해석

소프트웨어가 기계의 기계적인 부품들인 것처럼 객체가 조립되고 생성될 수 있도록 현실 세계의 개체(Entity)를 기계 부품과 같은 하나의 객체로 만드는 기법이다. 구조적인 문제들로 인한 소프트웨어 위기의 해결책으로 사용된다. 소프트웨어를 재사용하고 확장하여 고품질 소프트웨어를 신속하게 개발하고 유지하기 용이하다.

[문제 7]

STTD(Space Time Transmit Diversity, 시공간 송신 다이버시티)

[문제 8]

method

원문 해석

객체에 수행될 연산, 즉 객체의 속성값에 대한 변경이나 검색을 기술한 것이다. 또한 특정한 이것을 수행하기 위해서는 해당 객체에 메시지를 보내야 한다.

[문제 9]

프로토콜(Protocol)

[문제 10]

NGN(Next Generation Network)

[문제 11]

다중 경로 페이딩(Multipath Fading)

[문제 12]

OFDM(Orthogonal Frequency Division Multiplexing)

[문제 13]

MICS(Medical Implant Communication Service)

[문제 14]

QoS(Quality of Service)

[문제 15]

Stack

원문 해석

이것은 항목이 추가된 순서와 반대로 제거되는 특수한 형태의 데이터 구조이다. 그래서 가장 최근에 추가된 항목이 첫 번째로 제거된 항목이다. 이것은 선 입, 선출(LIFO)이라고 불린다.

[문제 16]

VoIP(Voice over Internet Protocol)

[문제 17]

쿠키(Cookie)

[문제 18]

Queue

원문 해석

이것은 수집에 있어 개체들이 순서대로 유지되고, 수집에 관한 주요(혹은 유일한) 작업은 단말의 전면 위치에서 개체들의 추가와 단말의 후면에서 개체들의 제거인 수집의 특별한 유형이다. 이는 선입선출(FIFO) 데이터 구조와 관련이 있다.

[문제 19]

- ① Tree ② Firewall

원문 해석

(트리)는 정보를 지정하는 노드가 가장 눈에 띄는 구조이다. 부모 노드, 자식 노드, 자손 노드로 구성되며, 분기와 동일한 구조를 가진다. 분기의 끝에 있는 노드, 즉 가장 어린 노드를 리프라고 한다.

(방화벽(Firewall))은 침입자로부터 컴퓨터 안전을 지키는 주요한 수단이다. (방화벽)은 사설 네트워크나 사용자 컴퓨터로부터 들어오거나 나가는 트래픽을 허용하거나 차단한다.

[문제 20]

AlphaGo

원문 해석

런던에 있는 Alphabet 사(社)의 구글 딥마인드에 의해 개발되었으며, 제로 · 마스터 · 리 등의 여러 가지 버전이 있다. 이것은 신경망 구조인 인간의 뇌와 같이 작동한다. 이 신경망은 정책망(Policy Network)과 가치망(Value Network)이라는 두 가지 신경망의 결합을 통해 만들어졌다. 정책망은 다음에 돌을 어디에 두는지 선택하는 알고리즘이고, 가치망은 승자를 예측하는 역할을 수행한다. 2015년 10월에 핸디캡 없이 프로 바둑 기사를 이긴 첫 컴퓨터 프로그램으로 유명하다.

[문제 21]

하둡(Hadoop)

[문제 22]

- ① SQL(Structured Query Language) ② DDL(Data Definition Language) ③ Relation ④ DML(Data Manipulation Language)
⑤ DCL(Data Control Language)

원문 해석

원래 (SQL)은 SEQUEL(구조화된 영문 질의 언어)이라고 불렸으며, IBM 연구소에서 실험적인 관계형 데이터베이스 시스템인 SYSTEM R을 위한 인터페이스로 설계 및 구현되었다.

(DDL)은 데이터베이스의 여러 구조를 정의하는 명령의 표준이다. (DDL) 문은 테이블, 색인 그리고 사용자와 같은 데이터베이스 객체들을 생성, 수정, 그리고 제거한다. 일반적인 (DDL) 문은 CREATE, ALTER, 그리고 DROP이다. (Relation)은 같은 구조를 갖는 튜플의 집합이다. (SQL)에서 (Relation)은 거의 테이블과 같은 의미로 사용된다. (Relation) 변수는 (DDL)에 의해 정의될 수 있다.

(DML)은 애플리케이션이나 쿼리를 통해 저장된 데이터를 처리하기 위해 데이터베이스 사용자가 사용하는 언어이다. 데이터베이스 사용자와 데이터베이스 관리 시스템의 사이에서 인터페이스를 제공한다. 또한 (DML)은 SELECT, INSERT, DELETE 그리고 UPDATE의 네 가지 종류가 있다.

(DCL)은 데이터의 보안, 무결성, 회복 그리고 병렬 수행 제어를 정의하기 위해 사용하는 언어이다. 데이터베이스 관리자는 데이터 관리 목적으로 (DCL)을 사용한다. 또한 이것은 COMMIT, ROLLBACK, GRANT, 그리고 REVOKE의 네 가지 종류가 있다.

[문제 23]

object

원문 해석

데이터와 데이터를 처리하는 함수를 캡슐화한 하나의 소프트웨어 모듈이다. 이것은 상태와 행위를 가지고 있으며, 다른 (객체)들과 구별될 수 있는 이름과 일정한 기억장소를 갖는다.

[문제 24]

Java

원문 해석

보안성이 뛰어나고 유지 보수가 용이한 객체 지향 프로그래밍(OOP) 언어이다. 컴파일 한 코드를 다른 운영체제에서 사용할 수 있도록 클래스를 제공함으로써 뛰어난 이식성을 갖고 있다. 또한 보다 효율적인 프로그래밍이 가능하도록 분산 환경을 지원한다.

[문제 25]

Array

원문 해석

(배열)은 인덱스(혹은 첨자)에 의해 구별되는 동일하게 입력된 데이터 항목의 집합이다. 변수와 다르게 같은 이름을 가지고 있어 같은 형식의 데이터를 저장하고 수집하기에 편리하다.

[문제 26]

TCP/IP

원문 해석

- (TCP/IP)는 전송 제어 프로토콜/인터넷 프로토콜의 약자로, 인터넷상의 호스트 연결에 사용되는 통신 프로토콜의 모음이다.
- (TCP/IP)는 UNIX 운영체제에 내장되어 있으며, 인터넷에 의해 사용된다.
- Netware와 같이 자체적인 고유 프로토콜을 가지고 있는 네트워크 운영체제 또한 (TCP/IP)는 지원한다.

[문제 27]

① - ㉑ ② - ㉓ ③ - ㉒ ④ - ㉔ ⑤ - ㉕

[문제 28]

① User Created Contents ② Space Time Transmit Diversity

[문제 29]

COOKIE

원문 해석

당신의 컴퓨터에 저장되는 소용량의 데이터 파일은 웹서버에 의해 만들어진다. 이것은 수신자 또는 제 3자에 의해 차후에 요청될 수도 있는 ID 또는 세션 번호와 같은 내용의 변환에 관계된 정보들을 제공한다. (COOKIE)는 당신을 웹사이트 상에서 인지하여 계속적으로 당신의 선호도를 파악할 수 있도록 하는 방법을 제공해준다.

[문제 30]

QoS

원문 해석

컴퓨터 네트워킹 및 기타 패킷 교환 형식의 텔레커뮤니케이션 망 분야에 있어 트래픽 공학 용어인 (QoS)는 달성된 서비스 품질보다는 자원 예약 메커니즘을 지칭한다. (QoS)는 각기 다른 응용 프로그램, 사용자 또는 데이터 흐름에 대해 서로 다른 우선순위를 할당하는 능력 또는 데이터 흐름에 대한 일정 수준의 성능을 보증할 수 있는 능력이다. 예를 들어, 요구되는 비트율, 지연, 지터, 패킷 손실 확률 및 비트 오류율 등이 보증되어야 할 수 있다.

[문제 31]

VPN(Virtual Private Network)

원문 해석

- VPN은 공공 회선을 사용해 노드를 연결하는 것으로 구축되는 네트워크이다. 인터넷을 데이터 전송 매체로 사용하는 네트워크를 생성할 수 있도록 하는 시스템이 다수 존재한다는 것을 예로 들 수 있다.
- 이러한 시스템은 인증된 사용자만이 네트워크에 접속할 수 있도록 하고 데이터가 도용될 수 없도록 하기 위해 암호화 및 기타 보안 메커니즘을 사용한다.

[문제 32]

PAD(Program Associated Data)

원문 해석

(PAD)는 다양한 정보를 갖고 전송되는 프로그램을 설명하기 위하여 많은 HD 라디오 및 위성 라디오 수신기들에 표시되는 여러 다양한 필드나 스트림으로 구성된 방송에 관련된 개념이다. 다음과 같은 여러 가지 정보를 포함한다.

- 노래 제목
- 프로그램 제목
- 아티스트 이름
- 앨범 이름
- 음악 장르

이것은 프로그램을 듣는 청취자들에게 보여주기 위한 것이다.

[문제 33]

- ① High performance Video Coding ② Long Term Evolution

[문제 34]

- ① Deadlock ② Exclusion ③ Hold ④ Preemption ⑤ Circular waiting

원문 해석

(교착상태)는 두 개의 프로세스가 수행되기 전 서로가 끝나기를 기다리고 있을 때 발생하는 상황이다. 결국 두 프로세스 모두 동작이 멈춘다. 병행 환경 하에서 요청하는 프로세스들에게 시스템 자원을 무작정 허용함으로써 (교착상태)는 발생할 수 있다. (교착상태)가 발생하기 위해서는 다음의 조건이 충족되어야 한다.

상호 (배제) - 한 번에 한 개의 프로세스만이 공유 자원을 사용할 수 있어야 한다. 이를 달성하기 위해 적어도 한 개 이상의 자원이 비공유 상태로 유지되어야 한다.

(점유) 및 대기 - 각 프로세스들은 다른 자원들을 얻기 위해 대기하는 동안 이미 할당된 자원들을 계속 점유한다.

비(선점) - 프로세스에 할당된 자원은 오직 프로세스의 자발적인 행동에 의해서만 할당이 해제되며, 시스템이 강제로 해제할 수 없어야 한다.

(환형 대기) - 각 프로세스들은 체인의 다음 프로세스가 요청하는 하나 이상의 자원을 점유한다. 예를 들어, 체인이 P1, P2, P3, P4로 구성되어 있다면, P1은 P2를 기다리고 있고 P2는 P3을 기다리고 있고 P3는 P4를 기다리고 있고 P4는 P1을 기다리고 있다.

[문제 35]

ISP(Internet Service Provider)

원문 해석

인터넷 접속 및 중계, 도메인 네임 등록, 웹 호스팅, 유즈넷 서비스, 콜로케이션 등 다양한 인터넷 서비스를 제공하는 사업자를 가리키는 용어다. 최초의 상업적인 서비스 제공자는 1989년 호주와 미국에서 설립된 'The World'다. 한국의 대표적인 제공업자로는 KT, SK 브로드밴드, LG 유플러스 등이 있다.

[문제 36]

MICS(Medical Implant Communication Service)

원문 해석

MICS는 의료용 주입물과의 통신에 402~405MHz의 주파수대를 사용하는데 대한 규격의 명칭이다. 이 규격은 심박 조율기 또는 기타 전자 주입물과의 양방향 무선 통신이 가능하게 한다. 같은 주파수대에 있는 다른 사용자들과의 간섭 위험을 감소시키기 위해 최대 출력 값이 상당히 낮다(EIRP=25). 어느 한 시점에서 사용되는 최대 대역폭은 300KHz이어서 WiFi나 블루투스에 비해 낮은 비트율의 시스템이 된다. 가장 큰 이점은 환자의 피부와 접촉하기 위해 외부 송·수신기를 필요로 했던 기존의 유도 방식의 기술에 비해 융통성이 더해진 것이다.

[문제 37]

① Universal Subscriber Identity Module ② Global Positioning System

[문제 38]

① - b ② - d ③ - e ④ - c ⑤ - a

원문 해석

- ① 디지털 정보를 부호화하기 위해 아날로그 반송파 신호를 변조하고, 수신된 정보를 복호화하기 위해 원래의 신호를 복조하는 장치이다.
- ② 질의를 하기 위한 언어이다. 데이터베이스 상의 데이터를 갱신, 삽입, 삭제 및 선택할 수 있다.
- ③ 사용자와 데이터베이스 사이에 위치하여 인터페이스의 역할을 하는 소프트웨어를 말한다.
- ④ 인터넷을 통한 파일 교환을 위한 프로토콜이다.
- ⑤ 블로그 수록, 주요 뉴스, 음성, 영상 등과 같은 표준 형식으로 자주 갱신되는 작업을 게재하는데 사용하는 웹피드(web feed) 패밀리이다.

[문제 39]

VoIP(Voice over Internet Protocol)

원문 해석

VoIP는 전통적인 PSTN 회선 전송망이 아니라 IP를 사용해 음성 데이터를 패킷 형태로 송신하는 것을 통해 사용자가 인터넷을 전화 통화용 송신매체로 사용할 수 있도록 하는 하드웨어 및 소프트웨어 분류를 말한다.

[문제 40]

USB(Universal Serial Bus)

원문 해석

개인용 컴퓨터와 주변기기들의 연결을 위해 프로토콜 규격을 정하는 산업표준들 중 하나다. 하나의 포트만으로 최대 127개의 장치를 연결하는 것이 가능하다. 초기에는 개인용 컴퓨터를 위해 개발되었으나, 오늘날에는 다양한 전자 제품에도 사용된다.

[문제 41]

RAID

원문 해석

RAID는 데이터 중복성이나 성능 향상 혹은 둘 다를 위해 여러 개의 물리적 디스크 드라이브 구성 요소를 하나의 논리 장치로 결합하는 데이터 저장소 가상화 기술이다.

[문제 42]

AI, Artificial Intelligence

원문 해석

(인공 지능)은 특히 컴퓨터 시스템에 의한 인간 지능 프로세스의 시뮬레이션이다. 이러한 과정에는 학습, 추론 및 자체 연결이 포함된다. (인공 지능)의 특별한 응용 프로그램은 전문가 시스템, 음성 인식, 머신 비전 등을 포함한다. (인공 지능)은 컴퓨터가 인간처럼 행동하는 것과 관련된 컴퓨터 과학의 영역이다. 이 용어는 1956년에 메사추세츠 공과 대학의 존 매커시에 의해 만들어졌다. (인공 지능)은 게임, 전문가 시스템, 자연 언어, 신경 회로, 로봇 공학을 포함한다.

[문제 43]

Universal Serial Bus

[문제 44]

FCFS(First Come First Served) 또는 FIFO(First In First Out)

원문 해석

(FCFS) 스케줄링은 가장 간단한 스케줄링 알고리즘이다. 이 방식에서는 CPU를 제일 처음 요청한 프로세스에게 CPU가 우선 할당된다.

(FCFS)는 기타 편파적인 특혜 없이 고객 또는 클라이언트의 요청이 도착하는 순서에 따라 수행하는 서비스 정책이다. 이 정책은 예를 들어, 판매 주문을 처리하거나 레스토랑의 좌석 배치를 결정할 때, 또는 택시 승차장 등에서 사용될 수 있다.

[문제 45]

① Field ② DBMS(Database Management System) ③ DBA(Database Administrator)

원문 해석

전통적인 데이터베이스는 (필드), 레코드, 파일로 구성된다. (필드)는 단일 정보 조각을 의미하며, 레코드는 (필드)들로 구성된 한 개의 완성된 집합이다. 파일은 레코드들의 모음이다. (DBMS)는 데이터베이스로부터 정보를 저장하고 수정하며 추출하도록 해주는 프로그램들의 집합이다. (DBA)는 (DBMS)에서 최고 수준의 권한을 갖는다. (DBA)는 자신이 가진 권한의 일정 부분을 다른 사람에게 허용할 수 있으며, 또한 어느 때라도 이를 철회할 수 있는 능력을 보유하고 있다.

[문제 46]

① Operating System ② Word ③ Spreadsheet ④ Database

원문 해석

(운영체제(OS))는 사용자와 컴퓨터 사이에서의 인터페이스를 제공하고 시스템 전체의 효율성과 생산성을 향상시키는 프로그램의 집합체이다. (운영체제)의 목적은 응답 시간 단축, 처리 능력 증대, 신뢰도 향상, 사용 가능성 증대로 요약할 수 있다.

(워드) 프로세서는 각종 전자 문서를 읽고 고치거나 작성할 수 있는 사무용 소프트웨어를 뜻한다. 또, 1970년대에서 1980년대에 인기 있던 독립형 사무기기의 일종을 뜻하기도 하는데, 문서 편집을 위한 타자기에 자판 입력과 인쇄 기능을 더한 것이 특징이었다.

(스프레드시트)는 데이터의 조직, 분석 및 보관을 표 형식으로 저장하는 대화형 컴퓨터 응용 프로그램이다. (스프레드시트)는 종이 회계 워크시트의 전산화된 시뮬레이션으로 개발되었다. 이 프로그램은 테이블의 셀에 입력된 데이터에서 작동한다. 각 셀은 숫자 또는 텍스트 데이터를 포함하거나 다른 셀의 내용을 기준으로 값을 자동으로 계산하고 표시하는 공식의 결과를 포함할 수 있다. (스프레드시트)는 그러한 전자 문서 또한 참조할 수도 있다.

(데이터베이스)는 체계화된 데이터의 모음이다. 여러 응용 시스템들의 통합된 정보들을 저장하여 운영할 수 있는 공용 데이터들의 묶음이다.

[문제 47]

- ① multimedia ② text ③ image

원문 해석

MMS는 셀룰러 네트워크를 통해 핸드폰으로 또는 핸드폰으로부터 (멀티미디어) 콘텐츠가 포함된 메시지를 보내는 표준적인 방법이다. MMS 표준은 SMS(Short Message Service) 기능을 확장하여, 160자 이상의 (텍스트) 메시지를 교환 할 수 있도록 한다. (텍스트) 전용 SMS와 달리, MMS는 최대 40초 분량의 비디오, 하나의 (이미지), 여러 (이미지)들의 슬라이드 쇼 또는 오디오를 포함한 다양한 미디어를 전달할 수 있다.

[문제 48]

- ① Domain Name ② Modem

원문 해석

(Domain Name)은 숫자로 된 IP 주소를 사람이 이해하기 쉬운 문자 형태로 표현한 것이다. (Domain Name)은 컴퓨터의 이름과 호스트 서버가 위치한 조직 또는 국가의 타입으로 구성된다. (Domain Name)은 일반적으로 오른쪽에서 왼쪽으로 읽으며, 가장 오른쪽 부분이 최상위 도메인에 해당한다. (Modem)은 부호화된 디지털 정보를 전송하기 위해 아날로그 신호로 변조하고 수신된 정보를 원래의 신호로 복조하는 장치이다. 목적은 전송이 쉽고 원래의 디지털 데이터로 복원할 수 있는 신호를 만드는 것이다.

[문제 49]

- ① Multi-Tasking ② C ③ Linux

원문 해석

- ① 하나 이상의 프로그램을 동시에 실행할 수 있는 기능이다. 예를 들어, MP3 음악 파일을 들으면서 인터넷에서 파일을 다운로드할 수 있다. 컴퓨터 하드웨어를 보다 효율적으로 사용할 수 있도록 하고, 개별 업무의 우선순위를 변경하여 중요한 작업이 덜 중요하게 생각되는 것보다 더 많은 프로세서 자원을 할당하도록 하는 조치를 포함하고 있다.
- ② 전산에 있어 이것은 1969년에서 1973년 사이 AT&T 벨연구소에서 Dennis Ritchie가 최초 개발한 범용 프로그래밍 언어이다. ALGOL 계보의 다른 대부분의 명령형 언어와 마찬가지로 구조적 프로그래밍을 위한 기능을 가지고 있으며, 어휘의 가변적 범위와 재귀를 허용한다.
- ③ Linus B. Torvalds가 개발한 컴퓨터 운영체제이며, 다양한 하드웨어 플랫폼에서 실행되는 매우 인기있는 Unix 운영체제의 한 버전이다. 서버 OS로 널리 사용되며, 데스크톱 시장에서 우위를 점하고 있다. 또한 그것은 누구나 무료로 사용할 수 있는 오픈 소스 소프트웨어이다.

