

데이터베이스

인공지능소프트웨어학과

담당교수: 김희숙
(jasmin11@hanmail.net)

데이터베이스

11주차

담당교수: 김희숙
(jasmin11@hanmail.net)

복습

(SQL: Group by)

담당교수: 김희숙
(jasmin11@hanmail.net)

[요약] 집계함수

-- 사원(직원코드,성명,직책,연봉)

[실습] 집계함수,

그룹화

18 • select * from 성적;

이름	점수
강성범	68
박찬호	75
선동열	70
신동엽	NULL
임꺽정	60
차범근	75
홍길동	87
홍명보	90
NULL	NULL

40 • SELECT * FROM 성적2;

이름	과목	점수
강성범	수학	68
박찬호	국어	75
선동열	영어	70
신동엽	영어	NULL
임꺽정	수학	60
차범근	수학	75
홍길동	영어	87
홍명보	수학	90
NULL	NULL	NULL

COUNT(*): 널 값 포함
COUNT(필드): 널 값 제외

SELECT
FROM

SUM()
AVG()
MAX()
MIN()

SELECT
FROM
GROUP BY
HAVING

-- 성적(이름, 점수)

- 1) 학생수를 구하시오
- 2) 시험에 응시한 학생를 구하시오
- 3) 점수의 평균을 구하시오

-- 성적2(이름, 과목, 점수)

- 4) 과목별 응시한 학생수를 구하시오
- 5) 과목별 평균점수를 구하시오

[실습] [실습 1-02](MySQL) 그룹화(성적2) (ans)

-- [실습] (그룹화)

-- 성적2(이름, 과목, 점수)

-- 2-1) 각 과목수는 몇 개인지 검색하라(DISTINCT 사용)

```
SELECT COUNT(DISTINCT 과목)
FROM 성적2
```

-- 2-2) 과목별 수강생은 몇 명인지 검색하라(GROUP BY)

```
SELECT 과목, COUNT(점수)
FROM 성적2
GROUP BY 과목;
```

-- 2-3) 과목별 평균점수를 검색하라(GROUP BY)

```
SELECT 과목, AVG(점수)
FROM 성적2
GROUP BY 과목;
```

40 • SELECT * FROM 성적2;

< Result Grid Filter Rows:

	이름	과목	점수
▶	강성범	수학	68
	박찬호	국어	75
	선동열	영어	70
	신동엽	영어	NULL
	임꺽정	수학	60
	차범근	수학	75
	홍길동	영어	87
	홍명보	수학	90
*	NULL	NULL	NULL

[실습] [실습 1-02](MySQL) 그룹화(성적2) (ans)

-- [실습] (그룹화)

-- 성적2(이름, 과목, 점수)

-- 2-4) 과목별 평균점수 75 보다 높은 과목의 과목별 평균점수를 검색하라 (HAVING)

```
SELECT 과목, AVG(점수)
FROM 성적2
GROUP BY 과목
HAVING AVG(점수) >= 75 ;
```

-- 2-5) 점수가 70 이상인 과목이름, 과목 평균점수를 과목의 과목별 평균점수가 75 이상인 것만 과목별 평균점수가 높은 순으로 검색하라(ORDER BY)

```
SELECT 과목, AVG(점수)
FROM 성적2
WHERE 점수 >= 70
GROUP BY 과목
HAVING AVG(점수) >= 75
ORDER BY AVG(점수) DESC;
```

/* SELECT 문법 순서 */
5 SELECT
1 FROM
2 WHERE
3 GROUP BY
4 HAVING
6 ORDER BY



	이름	과목	점수
▶	강성범	수학	68
	박찬호	국어	75
	선동열	영어	70
	신동엽	영어	NULL
	임궽정	수학	60
	차범근	수학	75
	홍길동	영어	87
	홍명보	수학	90
*	NULL	NULL	NULL

-- ROUND() 함수

```
select 과목, ROUND(AVG(점수), 1)
from 성적2
group by 과목;
```

실습(Quiz)

담당교수: 김희숙
(jasmin11@hanmail.net)

[실습] [Quiz 1] 테이블(fruit)

[Quiz 1] fruit

1. 다음과 같은 fruit 테이블을 생성하시오
(기본키: 과일번호)

데이터 입력하는 SQL문법을 작성하라

fruit

과일번호	과일이름	가격	색상	원산지
1	사과	2000	빨간색	한국
2	배	5000	노란색	한국
3	바나나	1500	노란색	인도네시아
4	키위	4000	초록색	호주
5	수박	6000	초록색	한국

2. 다음 SQL문법을 작성하시오

- 2-1) 노란색 과일의 이름, 색상을 검색하라.
- 2-2) 과일이름을 오름차순으로 정렬하여 모든 필드를 검색하라
- 2-3) 가격이 5000 이하인 빨간색 과일의 이름을 검색하라
- 2-4) 가격이 2000 ~ 5000 사이의 과일이름과 가격, 원산지를 검색하라(BETWEEN AND)
- 2-5) 색상이 노란색이거나 초록색인 과일이름, 색상을 검색하라 (IN 사용)
- 2-6) 색상이 노란색이거나 초록색인 과일이름, 색상을 검색하라 (논리연산자 OR 사용)
- 2-7) 원산지가 한국이 아닌 과일이름, 원산지를 검색하라

[실습] [Quiz 2] 테이블(고객)

[Quiz 2] 고객

1. 다음 <조건> 만족하는 고객 테이블을 생성 하십시오

- 고객 테이블은
- 고객아이디, 고객이름, 나이, 등급, 직업, 적립금 속성으로
구성되고,
- 고객아이디 속성이 기본키다.
- 고객이름은 값을 반드시 입력해야 하고,
- 적립금 속성은
- 값을 입력하지 않으면 0이 기본으로 입력되도록 한다

```
create table 고객 (  
    고객아이디 varchar(8)    NOT NULL ,  
    고객이름    varchar(20)  NOT NULL ,  
    나이        int ,  
    등급        varchar(10) ,  
    직업        varchar(20) ,  
    적립금      int          DEFAULT 0 ,  
    PRIMARY KEY(고객아이디)  
);
```

2. 다음 SQL문법을 작성하십시오

- 1) 직업이 학생, 간호사, 교사인 고객의 이름, 직업, 등급을 검색
- 2) 적립금이 300 이상인 고객 중에서 나이를 알 수 없는 고객아이디, 나이, 적
립금을 검색하라
- 3) 등급이 gold인 고객 중에서 고객아이디, 등급, 적립금을 검색하라.
단, 등급을 오름차순 정렬하고 등급이 같으면 적립금의 내림차순 정렬하라
-- (group by)
- 4) 등급별 적립금 평균을 검색하라.
(단, 적립금 평균이 1000원 이상인 등급에 대해 검색)

고객

고객아이디	고객이름	나이	등급	직업	적립금
apple	정소화	20	gold	학생	1000
banana	김선우	25	vip	간호사	2500
carrot	고명석	28	gold	교사	4500
orange	김용축	22	silver	학생	NULL
melon	성원용	35	gold	회사원	5000
peach	오형준	NULL	silver	의사	300
pear	채광주	31	silver	회사원	500

[실습] [Quiz 3] 테이블(사원)

[Quiz 3] 사원

사원

직원코드	성명	직책	연봉
161353	김미나	대리	2300
181323	최영락	사원	1900
151453	홍진호	과장	2800
135485	구준표	과장	3000
104895	강나래	팀장	3600
165484	김하늘	대리	2400

실행 순서

5 SELECT
1 FROM
2 WHERE
3 GROUP BY
4 HAVING
6 ORDER BY

SELECT
FROM
GROUP BY
HAVING

다음 SQL문법을 작성하시오

- 1) 최대 연봉을 구하시오
- 2) 사원수를 구하시오
- 3) 직책의 개수를 중복없이 검색하시오
- 4) 직책별 연봉의 평균을 구하시오
- 5) 직책별 사원수를 구하시오
- 6) 연봉이 2500 이상인 사원이 2명 이상인
- 직책의 사원수를 구하시오

SQL: 조인(join)

(내부조인, 외부조인)

담당교수: 김희숙
(jasmin11@hanmail.net)

조인(join)



□조인

- ✓ 2개 이상의 테이블로 부터 원하는 결과를 나타내는 것
- ✓ 2개 이상의 테이블로부터 필요한 열을 선택하여 하나의 테이블로 논리적인 결합하는 것

개인신상

이름	주민번호	생년월일	관계	핸드폰번호	우편번호	주소
박찬호	730201-1034343	1973-02-01	후배	(017) 234-2342	(333-333)	서울시 강남구 역삼동 234-2
선동열	680709-1078656	1968-07-09	동생	(018) 234-2342	(444-444)	서울시 서초구 서초동 234-2
임꺽정	700101-1027362	1970-01-01	고객	(016)2342-2342	(222-222)	서울시 영등포구 여의도동 234-2
차범근	600809-1987766	1960-08-09	친구	(019) 234-2342	(555-555)	서울시 종로구 관철동 234-2
홍길동	651214-1078767	1965-12-14	친구	(011) 234-2342	(111-111)	서울시 구로구 구로본동 234-24

자동차

이름	차종	차명	년식	구입일
임꺽정	트럭	포토	95	97-5
차범근	승용차	에쿠스	01	01-8
홍길동	승용차	소나타	98	98-8



이름	생년월일	차종	차명
임꺽정	1970-01-01	트럭	포토
차범근	1960-08-09	승용차	에쿠스
홍길동	1965-12-14	승용차	소나타

[실습] 조인(vehicle_join-mysql)

(테이블 1개) 고객관리

❖ 테이블 1개인 경우

	이름	주민번호	생년월일	관계	핸드폰번호	우편번호	주소	차종	차명	년식	구입일
▶	박찬호	720201-1034343	1973-02-01	후배	(017) 234-2342	(333-333)	서울시 강남구 역삼동 234-2	NULL	NULL	NULL	NULL
	선동열	680709-1078656	1968-07-09	동생	(018) 234-2342	(444-444)	서울시 서초구 서초동 234-2	NULL	NULL	NULL	NULL
	임꺽정	700101-1027362	1970-01-01	고객	(016)2342-2342	(222-222)	서울시 영등포구 여의도동 234-2	트럭	포토	1995	1997-05
	차범근	600809-1987766	1960-08-09	친구	(017) 234-2342	(555-555)	서울시 종로구 관철동 234-2	승용차	메쿠스	2001	2001-08
	홍길동	651214-1078767	1965-12-14	친구	(011) 234-2342	(111-111)	서울시 구로구 구로본동 234-24	승용차	소나타	1998	1998-08

(테이블 2개) 개인신상, 자동차

❖ 테이블 2개인 경우

	이름	주민번호	생년월일	관계	핸드폰번호	우편번호	주소
▶	박찬호	720201-1034343	1973-02-01	후배	(017) 234-2342	(333-333)	서울시 강남구 역삼동 234-2
	선동열	680709-1078656	1968-07-09	동생	(018) 234-2342	(444-444)	서울시 서초구 서초동 234-2
	임꺽정	700101-1027362	1970-01-01	고객	(016)2342-2342	(222-222)	서울시 영등포구 여의도동 234-2
	차범근	600809-1987766	1960-08-09	친구	(017) 234-2342	(555-555)	서울시 종로구 관철동 234-2
	홍길동	651214-1078767	1965-12-14	친구	(011) 234-2342	(111-111)	서울시 구로구 구로본동 234-24

	이름	생년월일	차종	차명
▶	임꺽정	1970-01-01	트럭	포토
	차범근	1960-08-09	승용차	메쿠스
	홍길동	1965-12-14	승용차	소나타

[요약] 조인

- ❖ 내부조인(theta join, equi join, natural join)
- ❖ 외부조인(left outer join, right outer join, full outer join)

1. 내부조인:

-- 방법1)

SELECT ..

FROM ..

WHERE ..

AND ...

AND ...

AND ...

1. 내부조인:

-- 방법2)

SELECT ..

FROM .. INNER JOIN ...

ON ..

WHERE ...

AND ...

AND ...

[예제] [실습 1] (부서, 사원) (stu)

[실습 1] (조인) 다음 SQL 문법과 실행결과를 작성하라

[내부조인] (테이블 2개) 부서, 사원

부서

부서코드	부서이름
A	인사부
B	연구부
C	홍보부

사원

사원번호	사원이름	부서코드
1	홍길동	C
2	임꺽정	A
3	차명석	A

외래키

[실습]

-- 부서(부서코드, 부서이름)

-- 사원(사원번호, 사원이름, 부서코드)

-- 1) 부서코드, 부서명, 사원이름을 검색하라

[예제] [실습 1] (부서, 사원) (ans)

[실습 1] (조인) 다음 SQL 문법과 실행결과를 작성하라

[내부조인] (테이블 2개) 부서, 사원

부서

부서코드	부서이름
A	인사부
B	연구부
C	홍보부

사원

사원번호	사원이름	부서코드
1	홍길동	C
2	임꺽정	A
3	차명석	A

외래키

[실습]

-- 부서(부서코드, 부서이름)

-- 사원(사원번호, 사원이름, 부서코드)

-- 1) 부서코드, 부서명, 사원이름을 검색하라

-- 1) 부서코드, 부서이름, 사원이름을 검색하라

-- 방법 1)

```
SELECT 부서.부서코드, 부서이름, 사원이름
FROM 부서, 사원
WHERE 부서.부서코드 = 사원.부서코드 ;
```

-- 1) 부서코드, 부서이름, 사원이름을 검색하라

-- 방법 2)

```
SELECT 부서.부서코드, 부서이름, 사원이름
FROM 부서 INNER JOIN 사원
ON 부서.부서코드 = 사원.부서코드 ;
```


[예제] [실습 2] (freshman, dmember)



[실습 2] (조인) 다음을 SQL과 실행결과를 작성하라

[내부조인] (테이블 2개) freshman, dmember

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

[실습] 내부 조인 (INNER JOIN)

```
-- freshman(name, address)
-- dmember(name, dept_name)
```

[실습] 외부 조인 (OUTER JOIN)

```
-- freshman(name, address)
-- dmember(name, dept_name)
```

-- **노트**에 각 실행결과를 작성하시오

-- 1. 내부조인

1-1) 동등 조인(equi join)

1-2) 자연 조인(natural join)

-- 2. 외부조인

2-1) 왼쪽 외부조인

2-2) 오른쪽 외부조인

2-3) 완전 외부조인

[예제] [실습 2] (freshman, dmember)

[실습 2] (조인) 다음을 SQL과 실행결과를 작성하라

[내부조인] (테이블 2개) freshman, dmember

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

[실습]

```
-- freshman(name, address)
-- dmember(name, dept_name)
```



-- 1. 내부 조인 (동등 조인: equi join)

-- 방법1)

```
SELECT *
FROM   freshman , dmember
WHERE  freshman.name=dmember.name;
```

-- 1. 내부 조인 (동등 조인: equi join)

-- 방법2)

```
SELECT *
FROM   freshman INNER JOIN dmember
ON     freshman.name=dmember.name;
```

[예제] [실습 2] (freshman, dmember)

[실습 2] (조인) 다음을 SQL과 실행결과를 작성하라

[내부조인] (테이블 2개) freshman, dmember

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

[실습]

```
-- freshman(name, address)
-- dmember(name, dept_name)
```



-- 1. 내부 조인 (자연 조인: natural join)

-- 방법1)

```
SELECT freshman.name, address, dept_name
FROM   freshman , dmember
WHERE  freshman.name=dmember.name;
```

-- 1. 내부 조인 (자연 조인: natural join)

-- 방법2)

```
SELECT  freshman.name, address, dept_name
FROM    freshman INNER JOIN dmember
ON      freshman.name=dmember.name;
```

[예제] [실습 2] (freshman, dmember)

[실습 2] (조인) 다음을 SQL과 실행결과를 작성하라

[내부조인] (테이블 2개) freshman, dmember

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

[실습]

```
-- freshman(name, address)
-- dmember(name, dept_name)
```



-- 2. 외부 조인 (**왼쪽 외부 조인**: left outer join)

-- 방법2)

```
SELECT *
FROM   freshman LEFT OUTER JOIN dmember
ON     freshman.name=dmember.name;
```

-- 2. 외부 조인 (**오른쪽 외부 조인**: right outer join)

-- 방법2)

```
SELECT *
FROM   freshman RIGHT OUTER JOIN dmember
ON     freshman.name=dmember.name;
```

[예제] [실습 3] (학생, 과목, 수강) (stu)

[실습 3] (조인) 다음 SQL 문법과 실행결과를 작성하라

[내부조인] (테이블 3개) 학생, 수강, 과목

(3개 테이블 조인)

학생

학번	학생명
11002	이홍근
24036	김순미
30419	박상웅
30555	홍서범
31890	유동희
31892	이정무
32000	김윤식

수강

학번	과목번호	성적
11002	CS310	98
11002	CS313	88
24036	CS345	90
30419	CS326	78
31892	CS388	86
32000	CS378	94
31890	CS310	92

과목

과목번호	과목명
CS310	데이터베이스
CS313	운영체제
CS345	자료구조
CS326	자바
CS546	DB프로그래밍
CS378	멀티미디어
CS388	웹프로그래밍

[실습]

- 학생(학번, 학생명)
- 과목(과목번호, 과목명)
- 수강(학번, 과목번호, 성적)
- 3-1) 수강한 학생의 학생명, 과목명, 성적을
- 검색하라(내부조인)
- 3-2) 이홍근의 학생명, 과목명, 성적을 검색하라

[실습] [실습 3] (학생, 과목, 수강) (ans)

[내부조인] (테이블 3개) 학생, 수강, 과목

(3개 테이블 조인)

학생

학번	학생명
11002	이홍근
24036	김순미
30419	박상웅
30555	홍서범
31890	유동희
31892	이정무
32000	김윤식

수강

학번	과목번호	성적
11002	CS310	98
11002	CS313	88
24036	CS345	90
30419	CS326	78
31892	CS388	86
32000	CS378	94
31890	CS310	92

과목

과목번호	과목명
CS310	데이터베이스
CS313	운영체제
CS345	자료구조
CS326	자바
CS546	DB프로그래밍
CS378	멀티미디어
CS388	웹프로그래밍

-- 3-1) 수강한 학생의 학생명, 과목명, 성적을 검색하라(내부조인)

-- 방법 1)

```
SELECT 학생명, 과목명, 성적
FROM 학생, 수강, 과목
WHERE 학생.학번 = 수강.학번
      AND 수강.과목번호 = 과목.과목번호;
```

-- 방법 2)

```
SELECT 학생명, 과목명, 성적
FROM 학생 INNER JOIN 수강
ON 학생.학번 = 수강.학번
   INNER JOIN 과목
ON 수강.과목번호 = 과목.과목번호;
```

[실습] [실습 3] (학생, 과목, 수강) (ans)

[내부조인] (테이블 3개) 학생, 수강, 과목

(3개 테이블 조인)

학생

학번	학생명
11002	이홍근
24036	김순미
30419	박상웅
30555	홍서범
31890	유동희
31892	이정무
32000	김윤식

수강

학번	과목번호	성적
11002	CS310	98
11002	CS313	88
24036	CS345	90
30419	CS326	78
31892	CS388	86
32000	CS378	94
31890	CS310	92

과목

과목번호	과목명
CS310	데이터베이스
CS313	운영체제
CS345	자료구조
CS326	자바
CS546	DB프로그래밍
CS378	멀티미디어
CS388	웹프로그래밍

-- 3-2) 이홍근의 학생명, 과목명, 성적을 검색하라

-- 방법 1)

```
SELECT 학생명, 과목명, 성적
FROM 학생, 수강, 과목
WHERE 학생.학번 = 수강.학번
      AND 수강.과목번호 = 과목.과목번호
      AND 학생명 = '이홍근';
```

-- 방법 2)

```
SELECT 학생명, 과목명, 성적
FROM 학생 INNER JOIN 수강
      ON 학생.학번 = 수강.학번
      INNER JOIN 과목
      ON 수강.과목번호 = 과목.과목번호
WHERE 학생명 = '이홍근';
```

[Quiz]

(답안)

담당교수: 김희숙
(jasmin11@hanmail.net)

[실습] [Quiz 4] 조인(고객, 주문) (stu)

[Quiz 4-1], [Quiz 4-2] : 고객, 주문

고객

고객아이디	고객이름	나이	등급
apple	김현준	20	gold
banana	정소화	25	vip
carrot	원유선	28	gold
orange	정지영	22	silver

주문

주문번호	주문고객	주문제품	수량
1001	apple	진짜우동	10
1002	carrot	맛있는파이	5
1003	banana	그대로만두	11

[실습]

- 고객(고객아이디, 고객이름, 나이, 등급)
- 주문(주문번호, 주문고객, 주문제품, 수량)

- (자연조인)
- 4-1) 고객아이디, 고객이름, 나이,
- 주문번호, 주문제품을 검색하라

- (왼쪽 외부조인)
- 4-2) 고객아이디, 고객이름, 나이,
- 주문번호, 주문제품을 검색하라
- (단, 주문하지 않은 고객도 포함)

[실습] [Quiz 4-1] 조인(고객, 주문) (ans)

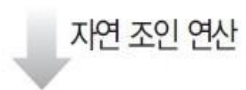
[Quiz 4-1] 고객, 주문

고객 릴레이션

고객아이디	고객이름	나이
apple	김현준	20
banana	정소화	25
carrot	원유선	28
orange	정지영	22

주문 릴레이션

주문번호	주문고객	주문제품
1001	apple	진짜우동
1002	carrot	맛있는파이
1003	banana	그대로만두



고객 ⋈_N 주문

고객아이디	고객이름	나이	주문번호	주문제품
apple	김현준	20	1001	진짜우동
banana	정소화	25	1003	그대로만두
carrot	원유선	28	1002	맛있는파이

-- (자연조인)

-- 4-1) 고객아이디, 고객이름, 나이,
-- 주문번호, 주문제품을 검색하라

-- (답안)

-- 4-1) 고객아이디, 고객이름, 나이,
-- 주문번호, 주문제품을 검색하라

```
SELECT 고객.고객아이디, 고객이름, 나이, 주문번호, 주문제품
FROM 고객, 주문
WHERE 고객.고객아이디 = 주문.주문고객;
```

[실습] [Quiz 4-2] 조인(고객, 주문) (ans)

[Quiz 4-2] 고객, 주문

고객

고객아이디	고객이름	나이
apple	김현준	20
banana	정소화	25
carrot	원유선	28
orange	정지영	22

주문

주문번호	주문고객	주문제품
1001	apple	진짜우동
1002	carrot	맛있는파이
1003	banana	그대로만두

↓ 왼쪽 외부 조인 연산

고객 ⋈ 주문

고객아이디	고객이름	나이	주문번호	주문제품
apple	김현준	20	1001	진짜우동
banana	정소화	25	1003	그대로만두
carrot	원유선	28	1002	맛있는파이
orange	정지영	22	NULL	NULL

- (왼쪽 외부조인)
- 1-2) 고객아이디, 고객이름, 나이,
- 주문번호, 주문제품을 검색하라
- (단, 주문하지 않은 고객도 포함)

-- (답안)

- 4-2) 고객아이디, 고객이름, 나이,
 - 주문번호, 주문제품을 검색하라
 - (단, 주문하지 않은 고객도 포함)
- SELECT 고객.고객아이디, 고객이름, 나이, 주문번호, 주문제품
FROM 고객 LEFT OUTER JOIN 주문
ON 고객.고객아이디 = 주문.주문고객;

[예제] [Quiz 4-2] 조인(고객, 주문)

(조인)

-- 고객(고객아이디, 고객이름, 나이, 등급)

-- 주문(주문번호, 주문고객, 주문제품, 수량)

CREATE TABLE 고객 (

 고객아이디 VARCHAR(20) NOT NULL,

 고객이름 VARCHAR(10),

 나이 INT,

 등급 VARCHAR(10) NOT NULL,

 PRIMARY KEY(고객아이디)

);

CREATE TABLE 주문 (

 주문번호 CHAR(4) NOT NULL,

 주문고객 VARCHAR(20),

 주문제품 varCHAR(20),

 수량 INT,

 PRIMARY KEY(주문번호),

 FOREIGN KEY(주문고객) REFERENCES 고객(고객아이디)

);

[예제] [Quiz 4-2] 조인(고객, 주문)

(조인)

-- 고객(고객아이디, 고객이름, 나이, 등급)
-- 주문(주문번호, 주문고객, 주문제품, 수량)

-- 고객(고객아이디, 고객이름, 나이, 등급)
INSERT INTO 고객 VALUES ('apple', '김현준', 20, 'gold');
INSERT INTO 고객 VALUES ('banana', '정소화', 25, 'vip');
INSERT INTO 고객 VALUES ('carrot', '원유선', 28, 'gold');
INSERT INTO 고객 VALUES ('orange', '정지영', 22, 'silver');

-- 주문(주문번호, 주문고객, 주문제품, 수량)
INSERT INTO 주문 VALUES ('1001', 'apple', '진짜우동', 10);
INSERT INTO 주문 VALUES ('1002', 'carrot', '맛있는파이', 5);
INSERT INTO 주문 VALUES ('1003', 'banana', '그대로만두', 45);

select * from 고객;
select * from 주문;

[과제02] (데이터베이스)

교재 5장 SELECT (01)

담당교수: 김희숙
(jasmin11@hanmail.net)

[과제02] (MySQL)

SQL작성(MySQL)
실습하기
(실습 화면캡처하여
워드에 작성)

□[과제01] **MySQL 프로그램 사용**하여 테이블 각각 작성하고
화면캡처하여 워드에 작성하시오

1) 제출방법: eClass-과제

첨부파일로 제출

2) 파일명: 과제02-분반_이름_학번-작성날짜.doc (또는 hwp)

(예. 과제02-QC_홍길동_24229999-제출날짜)

* (작성한 **SQL**파일은 저장해 두세요: 각자 별도로 보관해 두시기 바랍니다)

[실습] [HW02] (MySQL)

[실습]

- 과목(과목번호, 이름, 강의실, 개설학과, 시수)
- 학생(학번, 이름, 주소, 학년, 나이, 휴대폰번호, 소속학과)
- 수강(학번, 과목번호, 신청날짜, 중간성적, 기말성적, 평가학점)

과목번호	이름	강의실	개설학과	시수
c001	데이터베이스	126	컴퓨터	3
c002	정보보호	137	정보통신	3
c003	모바일웹	128	컴퓨터	3
c004	철학개론	117	철학	2
c005	전공글쓰기	120	교양학부	1
NULL	NULL	NULL	NULL	NULL

[실습 1] 다음에 대하여 SQL문법과 실행결과를 작성하라

- 예제5-1) 전체 학생의 이름과 주소를 검색하시오
- 예제5-2) 전체 학생의 모든 정보를 검색하시오
- (* 로 검색)
- (필드명으로 검색)
- 예제5-3) 전체 학생의 소속학과 정보를 중복없이 검색하시오
- (DISTINCT 사용)
- 예제5-3-2) 전체 학생의 소속학과 정보를 모두 검색하시오
- (ALL 키워드 사용)

학번	과목번호	신청날짜	중간성적	기말성적	평가학점
s001	c002	2019-09-03	93	98	A
s001	c004	2019-03-05	82	89	B
s001	c005	2020-09-03	74	79	C
s002	c001	2018-03-10	31	50	F
s003	c001	2019-03-03	81	82	B
s003	c002	2017-09-06	85	82	B
s004	c002	2018-03-05	92	95	A
s004	c003	2020-09-03	91	94	A
s004	c005	2019-03-03	72	78	C

학번	이름	주소	학년	나이	성별	휴대폰번호	소속학과
s001	김연아	서울 서초	4	23	여	010-1111-2222	컴퓨터
s002	홍길동	미정	1	26	남	NULL	통계
s003	이승엽	NULL	3	30	남	NULL	정보통신
s004	이영애	경기 분당	2	NULL	여	010-4444-5555	정보통신
s005	송윤아	경기 분당	4	23	여	010-6666-7777	컴퓨터
s006	홍길동	서울 종로	2	26	남	010-8888-9999	컴퓨터
s007	이은진	경기 과천	1	23	여	010-2222-3333	경영

[실습] [HW02] (MySQL)

[실습]

- 과목(과목번호, 이름, 강의실, 개설학과, 시수)
- 학생(학번, 이름, 주소, 학년, 나이, 휴대폰번호, 소속학과)
- 수강(학번, 과목번호, 신청날짜, 중간성적, 기말성적, 평가학점)

[실습 1] 다음에 대하여 SQL문법과 실행결과를 작성하라

- 예제5-4) 학생 중에서 2학년 이상인 컴퓨터 학과 학생의
-- 이름, 학년, 소속학과, 휴대폰번호 정보를 검색하시오
- 예제5-5) 1, 2, 3학년 학생이거나 컴퓨터 학과에
-- 소속되지 않은 학생의 이름, 학년, 소속학과, 휴대폰번호
-- 정보를 검색하시오
-- (부등호 사용)
-- (BETWEEN AND 사용)
- 예제5-6) 컴퓨터 학과나 정보통신 학과의 학생의 이름과 학년, 소속학과 정보를
-- 오름차순으로 검색하시오
- 예제5-7) 전체 학생의 모든 정보를 검색하되 학년을 기준으로 1차 오름차순 정렬하고,
-- 학년이 같은 경우에는 이름을 기준으로 2차 내림차순 정렬하여 검색하시오

[실습] [HW02] (MySQL)

[실습]

- 과목(과목번호, 이름, 강의실, 개설학과, 시수)
- 학생(학번, 이름, 주소, 학년, 나이, 휴대폰번호, 소속학과)
- 수강(학번, 과목번호, 신청날짜, 중간성적, 기말성적, 평가학점)

[실습 2] 집계함수, group by

- 예제5-8) 전체 학생수를 검색하시오
- 1) * 을 이용해 계산하는 경우
- 2) 학번 속성을 이용해 계산하는 경우
- 3) 주소 속성을 이용해 계산하는 경우(널 값 제외)

- 예제5-8-2) 전체학생수, 주소를 입력한 학생수, 주소의 종류를 중복없이 검색한 개수를 검색하시오

- 예제5-9) 여 학생의 평균 나이를 검색하시오

- 예제5-10) 전체 학생의 성별 최고 나이와 최저 나이를 검색하시오

- 예제5-11) 20대 학생만을 대상으로 나이별 학생수를 검색하시오
- 예제5-12) 각 학년별로 2명 이상의 학생을 갖는 학년에 대해서만 학생별 학생수를 검색하시오

[실습] [HW02] (MySQL)

[실습]

- 과목(과목번호, 이름, 강의실, 개설학과, 시수)
- 학생(학번, 이름, 주소, 학년, 나이, 휴대폰번호, 소속학과)
- 수강(학번, 과목번호, 신청날짜, 중간성적, 기말성적, 평가학점)

[실습 3] LIKE 연산자/ 널 값 검색 / 집합 연산자

- 예제5-13) 이 씨 성을 가진 학생들의 학번과 학생 이름을 검색하시오
- 예제5-14) 주소지가 '서울'인 학생의 이름, 주소, 학년을
-- 학년순(내림차순)으로 검색하시오
- 예제5-15) 휴대폰번호가 등록되지 않은(널 값을 갖는)
-- 학생의 이름과 휴대폰번호를 검색하시오
- 예제5-16) '여' 학생이거나 'A' 학점을 받은 학생의 학번을 검색하시오

[실습] [HW02] (MySQL)

[실습]

- 과목(과목번호, 이름, 강의실, 개설학과, 시수)
- 학생(학번, 이름, 주소, 학년, 나이, 휴대폰번호, 소속학과)
- 수강(학번, 과목번호, 신청날짜, 중간성적, 기말성적, 평가학점)

[실습 4] 조인

- 예제5-18) 전체 학생의 기본 정보와 모든 수강 정보를 검색하시오
 - 1) (SELECT FROM WHERE 방법)
 - 2) (SELECT FROM ON 방법)
- 예제5-19) 학생 중에서 과목번호가 c002 인 과목을 수강한 학생의 학번과 이름, 과목번호, 변환중간성적(학생별중간성적의 10% 가산점수)을 검색하시오
 - 1) (SELECT FROM WHERE 방법)
 - 2) (SELECT FROM ON 방법)
- 예제5-20) 학생 중에서 정보보호 과목을 수강한 학생의 학번과 이름, 과목번호를 검색하시오
 - 1) (SELECT FROM WHERE 방법)
 - 2) (SELECT FROM ON 방법)

[실습] [HW02] (MySQL)

[실습]

- 과목(과목번호, 이름, 강의실, 개설학과, 시수)
- 학생(학번, 이름, 주소, 학년, 나이, 휴대폰번호, 소속학과)
- 수강(학번, 과목번호, 신청날짜, 중간성적, 기말성적, 평가학점)

과목번호	이름	강의실	개설학과	시수
c001	데이터베이스	126	컴퓨터	3
c002	정보보호	137	정보통신	3
c003	모바일웹	128	컴퓨터	3
c004	철학개론	117	철학	2
c005	전공글쓰기	120	교양학부	1
NULL	NULL	NULL	NULL	NULL

[실습 4] 조인

- 예제5-21) 학생 중에서 과목번호가 c002 인 과목을 수강한 학생의
- 이름, 과목번호를 검색하시오
- (테이블 별칭 as 사용)

[외부 조인]

- 예제5-23) 과목을 수강하지 않은 학생을 포함하여 모든 학생의
- 학번, 이름과 학생이 수강한 교과목의 평가학점을 검색하시오

학번	과목번호	신청날짜	중간성적	기말성적	평가학점
s001	c002	2019-09-03	93	98	A
s001	c004	2019-03-05	82	89	B
s001	c005	2020-09-03	74	79	C
s002	c001	2018-03-10	31	50	F
s003	c001	2019-03-03	81	82	B
s003	c002	2017-09-06	85	82	B
s004	c002	2018-03-05	92	95	A
s004	c003	2020-09-03	91	94	A
s004	c005	2019-03-03	72	78	C

학번	이름	주소	학년	나이	성별	휴대폰번호	소속학과
s001	김연아	서울 서초	4	23	여	010-1111-2222	컴퓨터
s002	홍길동	미정	1	26	남	NULL	통계
s003	이승엽	NULL	3	30	남	NULL	정보통신
s004	이영애	경기 분당	2	NULL	여	010-4444-5555	정보통신
s005	송윤아	경기 분당	4	23	여	010-6666-7777	컴퓨터
s006	홍길동	서울 종로	2	26	남	010-8888-9999	컴퓨터
s007	이은진	경기 과천	1	23	여	010-2222-3333	경영