

초보자를 위한
Oracle SQL Database



데이터베이스 검색



한국기술교육대학교
온라인평생교육원

학습내용

- SELECT 문
- 검색 순서 및 조건

학습목표

- **SELECT 문**으로 원하는 데이터를 검색할 수 있다.
- **검색 순서 및 조건**을 지정할 수 있다.

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 - 기본 설정

SELECT 문 작성

```
SELECT [DISTINCT] {*} | column [alias] .. }
FROM table_name
```

[]: 생략 가능

|: 여러 옵션 중 한 개 선택

DISTINCT

: 중복 행을 제거하여 출력하는 기능

데이터베이스 검색

SELECT 문

SELECT 문 - 기본 설정

SELECT 문 작성 시 주의사항

- 대소문자를 구분하지 않음
: SQL 문은 대문자로 표시하는 것이 가독성 좋음
- 큰 따옴표 “ ”: 테이블, 컬럼 표시
작은 따옴표 ‘ ’: 데이터
- 쉼표 ,: 여러 개의 컬럼 구분
- 여러 줄로 작성 가능

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the central workspace, there is a SQL script window containing the following code:

```
SELECT * FROM "학생";
```

The word "SELECT" is highlighted with a red box. Below the workspace, a large blue button displays the same query code.

학번	성명	전공코드	학년	성별	학적상태
2021140025	김재민	3002	2	남	재학
2021151015	김지은	2002	2	여	재학
	학번 성명		전공코드	학년	성별
2021151031	윤영찬	2002	3	남	재학
2021151035	이찬우	2002	2	남	재학

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the central workspace, there is a SQL script window containing the following code:

```
SELECT * FROM "학생";
```

The word "FROM" is highlighted with a red box. Below the workspace, a large blue button displays the same query code.

학번	성명	전공코드	학년	성별	학적상태
2021140025	김재민	3002	2	남	재학
2021151015	김지은	2002	2	여	재학
	학번 성명		전공코드	학년	성별
2021151031	윤영찬	2002	3	남	재학
2021151035	이찬우	2002	2	남	재학

SELECT * FROM "학생";

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the center, a worksheet window displays a query result for students. The results are shown in a table:

학번	성명	학년	성별	학적상태	전공코드	
2021140029	김재민	3002	2	남	재학	3
2021151015	김지은	2002	2	여	재학	3
	윤영찬	2002	3	남	재학	5
	이판우	2002	2	남	재학	3

A pink callout box highlights the first two rows of the result set with the text: "여러 개의 칼럼이 있을 때 모든 칼럼을 보여줌". Below the table, a blue button contains the SQL query: "SELECT * FROM “학생”;".

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the center, a worksheet window displays a query result for students. The results are shown in a table:

학번	성명	학년	성별	학적상태	전공코드	
2021151031	윤영찬	2002	3	남	재학	5
2021151035	이판우	2002	2	남	재학	3

A pink callout box highlights the first two rows of the result set. Below the table, a blue button contains the SQL query: "SELECT * FROM “학생”;".

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 컬럼을 지정하여 데이터 추출

컬럼들을 하나씩 구분해주기 위해, 사용

```
SELECT "학번", "성명", "학적상태" ;
```

학번	성명	학적상태
2021151015	김지은	재학
2021151031	윤영찬	재학
2021151035	이찬우	재학
2021172019	배주연	재학

```
SELECT "학번", "성명", "학적상태" FROM "학생";
```

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 컬럼을 지정하여 데이터 추출

“”를 활용해 원하는 컬럼 지정 가능

```
SELECT "학번", "성명", "학적상태" ;
```

학번	성명	학적상태
2021151015	김지은	재학
2021151031	윤영찬	재학
2021151035	이찬우	재학
2021172019	배주연	재학

```
SELECT "학번", "성명", "학적상태" FROM "학생";
```

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 컬럼을 지정하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the '워크시트' (Worksheet) pane, the following SQL query is written:

```
SELECT "학번", "성명", "학적상태" FROM "학생";
```

The results are displayed in a table:

학번	성명	학적상태
2021151031	윤영찬	재학
2021151035	이판우	재학
2021172019	배주연	재학

```
SELECT "학번", "성명", "학적상태" FROM "학생";
```

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 컬럼을 지정하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the '워크시트' (Worksheet) pane, the following SQL query is written:

```
SELECT "학번", "수강년도", "교과목코드", "취득점수" FROM "수강";
```

The results are displayed in a table:

학번	수강년도	교과목코드	취득점수
2016174067	2021	CSE120	68
2016174067	2021	CSE121	87
2016174067	2021	CSE122	95
2016174067	2021	CSE130	94
2016174067	2021	CSE131	73

수강테이블에서
데이터를 가져옴

```
SELECT "학번", "수강년도", "교과목코드", "취득점수" FROM "수강";
```

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 컬럼을 지정하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the center, there is a SQL editor window with the following query:

```
SELECT "학번", "수강년도", "교과목코드", "취득점수"
FROM "수강";
```

The result of the query is displayed in a table below the editor. The table has four columns: 학번, 수강년도, 교과목코드, and 취득점수. The data is as follows:

학번	수강년도	교과목코드	취득점수
2016174067	2021	CSE120	68
2016174067	2021	CSE121	87
2016174067	2021	CSE122	95
2016174067	2021	CSE130	94
2016174067	2021	CSE131	73

한 학기에 여러 과목 수강

SELECT "학번", "수강년도", "교과목코드", "취득점수" FROM "수강";

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 컬럼을 지정하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the center, there is a SQL editor window with the following query:

```
SELECT "학번", "수강년도", "교과목코드", "취득점수"
FROM "수강";
```

The result of the query is displayed in a table below the editor. The table has four columns: 학번, 수강년도, 교과목코드, and 취득점수. The data is as follows:

학번	수강년도	교과목코드	취득점수
2016174067	2021	CSE120	68
2016174067	2021	CSE121	87
2016174067	2021	CSE122	95
2016174067	2021	CSE130	94
2016174067	2021	CSE131	73

동일한 학번에 다른 교과목코드, 다른 취득점수

SELECT "학번", "수강년도", "교과목코드", "취득점수" FROM "수강";

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 컬럼 지정 및 데이터 변경 시 결과 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스(M) 도구(I) 창(W) 도움말(H)

작업 영역

19c_remote

테이블(필터링됨)

- 교과목
- 교수
- 수강
- 전공
- 학부
- 학생
- 뷰
- 인덱스
- 폐기지
- 프로시저
- 연산자
- 함수
- 대기열
- 대기열 테이블
- 트리거

워크시트

```
SELECT "학번", "수강년도", "교과목", "취득점수"+10 "수강";
```

스크립트 출력 x

학번	수강년도	교과목	수강
2016174067	2021	CSE120	78
2016174067	2021	CSE121	97
2016174067	2021	CSE122	105
2016174067	2021	CSE130	104
2016174067	2021	CSE131	83

실제 데이터 변경이 아닌
데이터 변경 시 결과값 표시

+ 뿐만 아니라
다른 함수도
시뮬레이션 가능

SELECT "학번", "수강년도", "교과목코드", "취득점수"+10 FROM "수강";

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 중복 데이터를 제외하여 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스(M) 도구(I) 창(W) 도움말(H)

작업 영역

19c_remote

테이블(필터링됨)

- 교과목
- 교수
- 수강
- 전공
- 학부
- 학생
- 뷰
- 인덱스
- 폐기지
- 프로시저
- 연산자
- 함수
- 대기열
- 대기열 테이블
- 트리거

워크시트

```
DISTINCT "학번" FROM "수강";
```

스크립트 출력 x

학번
2015136057
2016174067
2014172008
2015136065
2018120034

중복되는 것 전부 제외

수강 테이블에서
중복되는 데이터를
제외한 학번 표시

SELECT DISTINCT "학번" FROM "수강";

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 예제 – 중복 데이터를 제외하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the left sidebar, there's a tree view of the database schema under '19c_remote'. A red box highlights the '수강' table node. In the main workspace, the query window contains the following SQL code:

```
SELECT DISTINCT "학번" FROM "수강";
```

The results pane displays five rows of data, each representing a unique school year:

2015136057
2016174067
2014172008
2015136065
2018120034

A purple callout box on the right side of the results pane says '중복되는 데이터가 없음' (No duplicate data).

`SELECT DISTINCT "학번" FROM "수강";`

데이터베이스 검색

SELECT 문

SELECT 문 - WHERE 절 활용

SELECT 문 WHERE 절 작성

```
SELECT [DISTINCT] {*} | column [alias] .. }
FROM table_name
[WHERE condition]
```

[]: 생략 가능
|: 여러 옵션 중 한 개 선택

DISTINCT
: 중복 행을 제거하여 출력하는 기능

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류		설명
산술 연산자	+	덧셈
	-	뺄셈
	*	곱셈
	/	나눗셈
비교 연산자	=	같다
	!=, <>	같지않다
	>	크다
	<	작다
	>=	크거나 같다
	<=	작거나 같다

데이터베이스 검색

SELECT 문

SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류		설명
산술 연산자	+	덧셈
	-	뺄셈
	*	곱셈
	/	나눗셈
비교 연산자	=	같다
	!=, <>	같지않다
	>	크다
	<	작다
	>=	크거나 같다
	<=	작거나 같다

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류	설명	
논리 연산자	AND	조건을 모두 만족
	OR	조건 중 하나만 만족
	NOT	조건을 만족하지 못함
기타 연산자	BETWEEN	특정 범위의 값을 지정 · 와일드 검색 · % : 모든 문자열을 의미. · _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	LIKE	
	IS NULL, IS NOT NULL	NULL 조건

데이터베이스 검색

SELECT 문

SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류	설명	
논리 연산자	AND	조건을 모두 만족
	OR	조건 중 하나만 만족
	NOT	조건을 만족하지 못함
기타 연산자	BETWEEN	특정 범위의 값을 지정 · 와일드 검색 · % : 모든 문자열을 의미. · _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	LIKE	
	IS NULL, IS NOT NULL	NULL 조건

SELECT 문

데이터베이스 검색

SELECT 문

SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류	설명	
논리 연산자	AND	조건을 모두 만족
	OR	조건 중 하나만 만족
	NOT	조건을 만족하지 못함
기타 연산자	BETWEEN	특정 범위의 값을 지정 · 와일드 검색 · % : 모든 문자열을 의미. · _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	LIKE	
	IS NULL, IS NOT NULL	NULL 조건

데이터베이스 검색

SELECT 문

SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류	설명	
논리 연산자	AND	조건을 모두 만족
	OR	조건 중 하나만 만족
	NOT	조건을 만족하지 못함
기타 연산자	BETWEEN	특정 범위의 값을 지정 · 와일드 검색 · % : 모든 문자열을 의미. · _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	LIKE	
	IS NULL, IS NOT NULL	NULL 조건

SELECT 문

데이터베이스 검색

SELECT 문

◆ SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류	설명	
논리 연산자	AND	조건을 모두 만족
	OR	조건 중 하나만 만족
	NOT	조건
기타 연산자	BETWEEN	특정 예시) 김% → 성이 김씨인 사람 · 와는 —————— · % : 모든 문자열을 의미.
	LIKE	· _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	IS NULL, IS NOT NULL	NULL 조건

데이터베이스 검색

SELECT 문

◆ SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류	설명	
논리 연산자	AND	조건을 모두 만족
	OR	조건 중 하나만 만족
	NOT	조건을 만족하지 못함
기타 연산자	BETWEEN	특정 예시) _bc → abc, bbc 모두 만족 · 와는 —————— · % : 모든 문자열을 의미.
	LIKE	· _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	IS NULL, IS NOT NULL	NULL 조건

SELECT 문

데이터베이스 검색

SELECT 문

◆ SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류	설명	
논리 연산자	AND	조건을 모두 만족
	OR	조건 중 하나만 만족
	NOT	조건을 만족하지 못함
기타 연산자	BETWEEN	특정 범위의 값을 지정 필요한 부분에 _를 넣어주면 · 와일드 검색 · % : 모든 문자열을 의미. · _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	LIKE	· _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	IS NULL, IS NOT NULL	NULL 조건

데이터베이스 검색

SELECT 문

◆ SELECT 문 - WHERE 절 활용

+ WHERE 절을 이용한 조건 검색

연산자 종류	설명	
논리 연산자	AND	조건을 모두 만족
	OR	조건 중 하나만 만족
	NOT	조건을 만족하지 못함
기타 연산자	BETWEEN	특정 범위의 값을 지정
	LIKE	· 와일드 검색 · % : 모든 문자열을 의미. · _ : 한 글자를 의미, 반복해서 사용할 경우에는 반복한 수만큼의 글자.
	IS NULL, IS NOT NULL	NULL 조건

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 학생 테이블에서 2학년인 학생 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스(T) 팀(M) 도구(I) 창(W) 도움말(H)

19c_remote

정속 시작 페이지 19c_remote 0.029초

테이블(필터링됨) 학생

학번 성명 성별

2021180100 정상훈 남

2021180111 최미로 남

24개 행이 선택되었습니다.

```
SELECT "학번", "성명", "성별" FROM "학생";
```

SELECT "학번", "성명", "성별" FROM "학생" WHERE "학년"=2;

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 학생 테이블에서 2학년인 학생 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스(T) 팀(M) 도구(I) 창(W) 도움말(H)

19c_remote

정속 시작 페이지 19c_remote 0.029초

테이블(필터링됨) 학생

학번 성명 성별

2021180100 정상훈 남

2021180111 최미로 남

24개 행이 선택되었습니다.

학번 성명 성별

2021180100 정상훈 남

2021180111 최미로 남

24개 행이 선택되었습니다.

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "학년"=2;
```

학생 테이블에서
2학년 데이터만 추출

SELECT "학번", "성명", "성별" FROM "학생" WHERE "학년"=2;

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 학생 테이블에서 2학년인 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the workspace, there is a query window containing the following SQL code:

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "학년"=2;
```

The results of the query are displayed in a table:

학번	성명	성별
2021180100	정상훈	남
2021180111	최미르	남

A message at the bottom of the results pane says "24개 행이 선택되었습니다." (24 rows selected).

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "학년"=2;
```

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 3, 4학년의 남학생인 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the workspace, there is a query window containing the following SQL code:

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "학년">>2 AND "성별"='남';
```

A pink callout box highlights the condition "2학년 이상인 3학년, 4학년 학생".

The results of the query are displayed in a table:

학번	성명	성별
2018180021	김우찬	남
2018180108	전상하	남
2019136078	오현식	남
2019161124	조규열	남
2021151031	윤영찬	남

A message at the bottom of the results pane says "17개 행이 선택되었습니다." (17 rows selected).

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE "학년">>2 AND "성별"='남';
```

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 3, 4학년의 남학생인 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. The '워크시트' (Worksheet) pane contains the following SQL query:

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE "학년">>2 AND "성별"='남';
```

The results of the query are displayed in the '스크립트 출력' (Script Output) pane:

학번	성명	성별
2018161052	백근정	남
2018180021	김우찬	남
2018180108	전상하	남
2019136078	오현식	남
2019161124	조규열	남
2021151031	윤영찬	남

A red box highlights the WHERE clause: WHERE "학년">>2 AND "성별"='남';

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE "학년">>2 AND "성별"='남';
```

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 3, 4학년의 남학생인 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. The '워크시트' (Worksheet) pane contains the following SQL query:

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE "학년">>2 AND "성별"='남';
```

The results of the query are displayed in the '스크립트 출력' (Script Output) pane:

학번	성명	성별
2018161052	백근정	남
2018180021	김우찬	남
2018180108	전상하	남
2019136078	오현식	남
2019161124	조규열	남
2021151031	윤영찬	남

A red box highlights the WHERE clause: WHERE "학년">>2 AND "성별"='남';

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE "학년">>2 AND "성별"='남';
```

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 이수학기가 4~7학기인 학생 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스 팀(M) 도구(I) 창(W) 도움말(H)

19c_remote

SELECT "학번", "성명", "성별" FROM "학생" WHERE "이수학기" BETWEEN 4 AND 7;

작업이 완료되었습니다.(0.027초)

학번	성명	성별
2021151031	윤영찬	남
2021180095	이효진	여

24개 행이 선택되었습니다.

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE "이수학기" BETWEEN 4 AND 7;
```

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 이수학기가 4~7학기인 학생 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스 팀(M) 도구(I) 창(W) 도움말(H)

19c_remote

워크시트 질의 작성기

SELECT "학번", "성명", "성별" FROM "학생" WHERE "이수학기" BETWEEN 4 AND 7;

작업이 완료되었습니다.(0.027초)

학번	성명	성별
2021151031	윤영찬	남
2021180095	이효진	여

24개 행이 선택되었습니다.

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE "이수학기" BETWEEN 4 AND 7;
```

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 이수학기가 4~7학기인 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the '워크시트' (Worksheet) tab, a query is written:

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "이수학기" BETWEEN 4 AND 7;
```

The '이수학기' column in the query is highlighted with a red box. The results pane shows two rows of student information:

학번	성명	성별
2021151031	윤영찬	남
2021180095	이효진	여

A message at the bottom of the results pane says '24개 행이 선택되었습니다.' (24 rows selected).

```
SELECT "학번", "성명", "성별" FROM "학생"  
WHERE "이수학기" BETWEEN 4 AND 7;
```

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 이수학기가 4~7학기인 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the '워크시트' (Worksheet) tab, a query is written:

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "이수학기" BETWEEN 4 AND 7;
```

The entire results pane is highlighted with a red box. The results show the same two student records as the previous screenshot.

학번	성명	성별
2021151031	윤영찬	남
2021180095	이효진	여

A message at the bottom of the results pane says '24개 행이 선택되었습니다.' (24 rows selected).

```
SELECT "학번", "성명", "성별" FROM "학생"  
WHERE "이수학기" BETWEEN 4 AND 7;
```

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 이수학기가 4~7이면서 3학년인 학생 데이터 추출

여러 가지 조건을 줄 수 있음

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE
BETWEEN 4 AND 7) AND "학년"=3;
```

작업이 완료되었습니다.(0.03초)

학번	성명	성별
2018180108	전상하	남
2019136078	오현식	남
2019161124	조규열	남
2021151031	윤영찬	남

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE ("이수학기" BETWEEN 4 AND 7) AND "학년"=3;
```

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 이수학기가 4~7학기이면서 3학년인 학생 데이터 추출

이수학기가 4~7학기이면서 3학년인 학생

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE
("이수학기" BETWEEN 4 AND 7) AND "학년"=3;
```

작업이 완료되었습니다.

학번	성명	성별
2018180108	전상하	남
2019136078	오현식	남
2019161124	조규열	남
2021151031	윤영찬	남

```
SELECT "학번", "성명", "성별" FROM "학생"
WHERE ("이수학기" BETWEEN 4 AND 7) AND "학년"=3;
```

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 김씨 성을 가진 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the central workspace, there is a script window containing the following SQL query:

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "성명" LIKE '김%';
```

Below the query, the results are displayed in a table format. The results show records for students named Kim, with their student ID, name, and gender. The results are as follows:

학번	성명	성별
2020180015	김선강	남
2021136037	김태훈	남
2021140029	김재민	여
2021151015	김지은	여
2021180022	김영민	남

```
SELECT "학번", "성명", "성별" FROM "학생"  
WHERE "성명" LIKE '김%';
```

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 김씨 성을 가진 학생 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the central workspace, there is a script window containing the following SQL query. The WHERE clause is highlighted with a red box.

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "성명" LIKE '김%';
```

Below the query, the results are displayed in a table format. The results show records for students named Kim, with their student ID, name, and gender. The results are as follows:

학번	성명	성별
2020180015	김선강	남
2021136037	김태훈	남
2021140029	김재민	여
2021151015	김지은	여
2021180022	김영민	남

```
SELECT "학번", "성명", "성별" FROM "학생"  
WHERE "성명" LIKE '김%';
```

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 김씨 성을 가진 학생 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스 팀(M) 도구(I) 창(W) 도움말(H)

19c_remote

작성기

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "성명" LIKE '김%';
```

스クリプ트 출력

작업이 완료되었습니다.(0.033초)

학번	성명	성별
2020174011	김민석	남
2020180015	김선강	남
2021136037	김태훈	남
2021140029	김재민	여
2021151015	김지은	남
2021180022	김영민	남

19개 행이 선택되었습니다.

하나의 글자가 아닌 문자열
복수의 문자

SELECT "학번", "성명", "성별" FROM "학생"
WHERE "성명" LIKE '김%';

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 김씨 성을 가진 학생 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스 팀(M) 도구(I) 창(W) 도움말(H)

19c_remote

작성기

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "성명" LIKE '김%';
```

학번	성명	성별
2020174011	김민석	남
2020180015	김선강	남
2021136037	김태훈	남
2021140029	김재민	여
2021151015	김지은	남
2021180022	김영민	남

19개 행이 선택되었습니다.

SELECT "학번", "성명", "성별" FROM "학생"
WHERE "성명" LIKE '김%';

SELECT 문

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 이메일 주소가 naver.com인 학생 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스 팀(M) 도구(I) 장(W) 도움말(H)

작업 폴더 : 학생

작업

이메일 주소가 naver.com인 모든 사람 조회

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "이메일" LIKE '%naver.com';
```

작업이 완료되었습니다.(0.025초)

학번	성명	성별
2021151035	이판우	남
2021172019	배주연	여

```
SELECT "학번", "성명", "성별" FROM "학생"  
WHERE "이메일" LIKE '%naver.com';
```

데이터베이스 검색

SELECT 문

WHERE 절 예제 – 이메일 주소가 naver.com인 학생 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스 팀(M) 도구(I) 장(W) 도움말(H)

작업 폴더 : 학생

작업

이메일 주소가 naver.com인 모든 사람 조회

```
SELECT "학번", "성명", "성별" FROM "학생" WHERE "이메일" LIKE '%naver.com';
```

학번	성명	성별
2021151035	이판우	남
2021172019	배주연	여

```
SELECT "학번", "성명", "성별" FROM "학생"  
WHERE "이메일" LIKE '%naver.com';
```

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

검색 순서 및 조건

검색 순서 및 조건

`SELECT column, group_function(column), GROUPING(column)`

① `FROM table` 테이블의 데이터 가져오기

[`WHERE condition`]

[`GROUP BY [ROLLUP | CUBE] group_by_expression`]

[`HAVING group_condition`]

[`ORDER BY column [asc | desc]`]

- **SELECT 문 순서**

FROM → WHERE → GROUP BY → HAVING → SELECT
→ ORDER BY

데이터베이스 검색

검색 순서 및 조건

검색 순서 및 조건

검색 순서 및 조건

`SELECT column, group_function(column), GROUPING(column)`

① `FROM table`

② `[WHERE condition]` 조건을 만족하는 데이터 가져오기

[`GROUP BY [ROLLUP | CUBE] group_by_expression`]

[`HAVING group_condition`]

[`ORDER BY column [asc | desc]`]

- **SELECT 문 순서**

FROM → WHERE → GROUP BY → HAVING → SELECT
→ ORDER BY

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

검색 순서 및 조건

검색 순서 및 조건

`SELECT column, group_function(column), GROUPING(column)`

① `FROM table`

② `[WHERE condition]`

③ `[GROUP BY [ROLLUP | CUBE] group_by_expression]`

`[HAVING group_condition]`

`[ORDER BY column [asc | desc]]`

데이터 그룹핑

- **SELECT 문 순서**

`FROM → WHERE → GROUP BY → HAVING → SELECT
→ ORDER BY`

데이터베이스 검색

검색 순서 및 조건

검색 순서 및 조건

검색 순서 및 조건

`SELECT column, group_function(column), GROUPING(column)`

① `FROM table`

② `[WHERE condition]`

③ `[GROUP BY [ROLLUP | CUBE] group_by_expression]`

④ `[HAVING group_condition]`

그룹핑의 조건 지정하기

`[ORDER BY column [asc | desc]]`

- **SELECT 문 순서**

`FROM → WHERE → GROUP BY → HAVING → SELECT
→ ORDER BY`

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

검색 순서 및 조건

검색 순서 및 조건

- 5 **SELECT** column, group_function(column), GROUPING(column)
- 1 **FROM** table
- 2 [**WHERE** condition]
- 3 [**GROUP BY** [ROLLUP | CUBE] group_by_expression]
- 4 [**HAVING** group_condition]
- 6 [**ORDER BY** column [asc | desc]]

,로 칼럼 구분

▪ SELECT 문 순서

FROM → WHERE → GROUP BY → HAVING → SELECT
→ ORDER BY

데이터베이스 검색

검색 순서 및 조건

검색 순서 및 조건

검색 순서 및 조건

- 5 **SELECT** column, group_function(column), GROUPING(column)
- 1 **FROM** table
- 2 [**WHERE** condition]
- 3 [**GROUP BY** [ROLLUP | CUBE] group_by_expression]
- 4 [**HAVING** group_condition]
- 6 [**ORDER BY** column [asc | desc]]

오름차순/내림차순 지정

▪ SELECT 문 순서

FROM → WHERE → GROUP BY → HAVING → SELECT
→ ORDER BY

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

◆ ORDER BY 절

ORDER BY 절

- 출력 결과를 정렬하여 보여줄 때 사용
- 기본값은 오름차순(ASC) 정렬
 - 숫자는 가장 작은 수에서 큰 수로 출력
 - 영문자는 알파벳순, 한글은 가나다순으로 출력
 - 날짜는 과거에서 미래 순으로 출력
- 내림차순(DESC)은 오름차순의 반대

데이터베이스 검색

검색 순서 및 조건

◆ ORDER BY 절

ORDER BY 절

```
SELECT [DISTINCT] {*} | column [alias]..
FROM table
[WHERE condition]
[ORDER BY {column, expression} [ASC | DESC]]
```

ORDER BY : 컬럼이나 표현식을 정렬하여 출력할 때 사용
 ASC : 기본값으로 생략 가능(default)
 DESC : 내림차순 정렬시에 사용하며 생략 불가능

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

ORDER BY 절 예제 – 성명을 내림차순으로 정렬하여 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스 팀(M) 도구(I) 창(W) 도움말(H)

정속 시작 페이지 19c_remote 학생

워크시트 질의 작성기

```
SELECT "학번", "성명", "성별" FROM "학생"
ORDER BY "성명"
```

스크립트 출력 x

작업이 완료되었습니다.(0.03초)

학번	성명	성별
2020120019	김민경	여
2020120017	김민경	남
2020136008	구용원	남
2021136004	곽의신	남
2016180004	강지영	여
2016180001	강나영	여

97개 행이 선택되었습니다.

```
SELECT "학번", "성명", "성별" FROM "학생"
ORDER BY "성명" DESC;
```

데이터베이스 검색

검색 순서 및 조건

ORDER BY 절 예제 – 성명을 내림차순으로 정렬하여 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 소스 팀(M) 도구(I) 창(W) 도움말(H)

정속 시작 페이지 19c_remote 학생

워크시트 질의 작성기

```
SELECT "학번", "성명", "성별" FROM "학생" ORDER BY "성명" DESC;
```

스크립트 출력 x

작업이 완료되었습니다.(0.03초)

학번	성명	성별
2020120019	김민경	여
2020120017	김민경	남
2020136008	구용원	남
2021136004	곽의신	남
2016180004	강지영	여
2016180001	강나영	여

97개 행이 선택되었습니다.

내림차순으로 성명 정렬

```
SELECT "학번", "성명", "성별" FROM "학생"
ORDER BY "성명" DESC;
```

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

◆ ORDER BY 절 예제 – 성명을 내림차순으로 정렬하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the center, there's a '워크시트' (Worksheet) pane with the following SQL query:

```
SELECT "학번", "성명", "성별" FROM "학생" ORDER BY "성명" DESC;
```

The results of the query are displayed in a table below the worksheet. The table has three columns: 학번 (Student ID), 성명 (Name), and 성별 (Gender). The results are:

학번	성명	성별
2020120	김민석	남
2020120	김민경	여
2020136	구용원	남
2021136	곽의신	남
2016180	강지영	여
2016180	강나영	여

A red box highlights the '성명' column, specifically the rows for '김민경' and '구용원'.

```
SELECT "학번", "성명", "성별" FROM "학생"  
ORDER BY "성명" DESC;
```

데이터베이스 검색

검색 순서 및 조건

◆ GROUP BY 절

GROUP BY 절

- 테이블의 특정 컬럼 값을 기준으로 그룹핑할 때 사용
 - GROUP BY 절의 규칙
 - GROUP BY 절에 그룹핑하려는 컬럼들을 추가
 - GROUP BY 절에서는 컬럼 별명은 사용할 수 없음
 - 그룹함수를 적용하지 않은 단순 컬럼은
SELECT 절에 사용할 수 없음
- 단, GROUP BY 절에 명시된 컬럼은 예외

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the central workspace, a query is displayed:

```
SELECT "학번", COUNT("학번") AS 신청과목수 , AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

The result set shows the following data:

학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수 , AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the central workspace, a query is displayed:

```
SELECT "학번", COUNT("학번") AS 신청과목수 , AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

The result set shows the following data:

학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수 , AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(U) 이동(N) 실행(R) 소스 팀(M) 도구(I) 창(W) 도움말(H)

19c_remote 시작 페이지 19c_remote 수강 0.028초

워크시트 질의 작성기

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수")
FROM "수강" GROUP BY "학번";
```

스크립트 출력 x 작업이 완료되었습니다.(0.028초)

학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

Oracle SQL Developer : 19c_remote

파일(F) 편집(E) 보기(U) 이동(N) 실행(R) 소스 팀(M) 도구(I) 창(W) 도움말(H)

19c_remote 시작 페이지 19c_remote 수강 0.028초

워크시트 질의 작성기

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수")
FROM "수강" GROUP BY "학번";
```

스크립트 출력 x 작업이 완료되었습니다.(0.028초)

학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

함수값이 그대로일 경우
무슨 내용인지 알 수 없음

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the workspace, there is a query window with the following SQL code:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번";
```

The result of the query is displayed in a table below:

학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

A pink callout box highlights the alias 'AS 평균점수' and contains the text: '별칭을 줘서 다른 값을 대신 보여줌'.

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. In the workspace, there is a query window with the following SQL code:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번";
```

The result of the query is displayed in a table below:

학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

A pink callout box highlights the alias 'AVG("취득점수") AS 평균점수' and contains the text: '신청과목수 평균점수'.

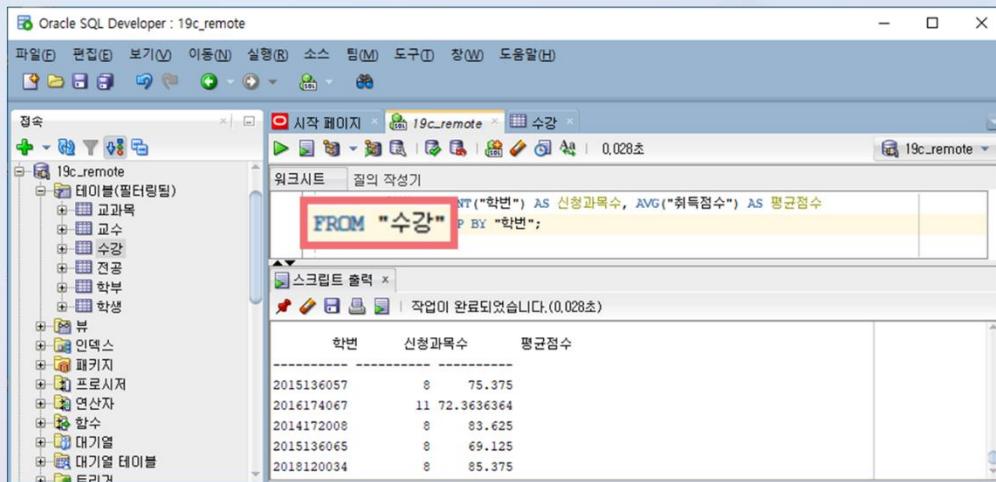
```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출



The screenshot shows the Oracle SQL Developer interface. In the central workspace, there is a query window with the following SQL code:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번";
```

The word 'GROUP' is highlighted with a red box. Below the query window, a results grid displays the following data:

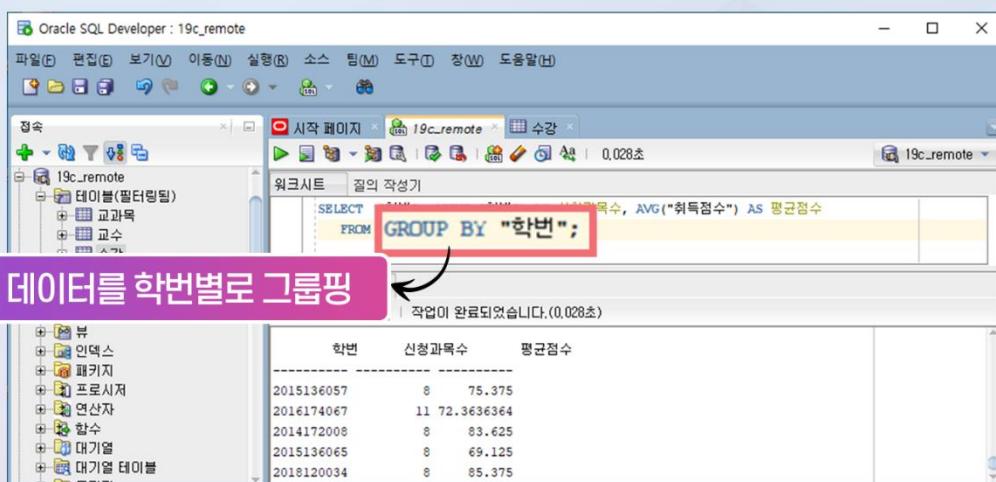
학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

**SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"**

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출



The screenshot shows the Oracle SQL Developer interface. In the central workspace, there is a query window with the following SQL code:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번";
```

A pink callout box points to the 'GROUP BY' clause with the text '데이터를 학번별로 그룹핑'. Below the query window, a results grid displays the following data:

학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

**SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"**

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface with a query window displaying the following results:

학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수 , AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번"
```

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface with a query window displaying the following results:

학번	신청과목수	평균점수
2018120034	8	85.375
2018120034	11	72.3636364
2018120034	8	83.625
2018120034	8	69.125
2018120034	8	75.375
2014172008	8	83.625
2014172008	8	69.125
2014172008	8	75.375

학번을 기준으로 그룹핑 ➡ 신청한 과목수와 평균 계산

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

Oracle SQL Developer : 테이블 STUDY01.수강@19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 팀(M) 도구(I) 창(W) 도움말(H)

정속

19c_remote

데이터 Model 제약 조건 권한 부여 통계 트리거 플래시백 증속성 세부정보 분할 영역 인덱스 SQL

수강

시작 페이지 19c_remote 정렬... 필터:

학번	수강년도	수강학기	이수구분	취득점수	재수강여부	교과목코드	교수사번
1 2018120034	2021	1A		94 X		ARB263	20091015
2 2018120034	2021	1A		94 X		CSE114	20091039
3 2018120034	2021	1A		88 X		CSE120	20092019
4 2018120034	2021	1A		83 X		CSE121	20093032
5 2018120034	2021	1A		91 X		CSE122	20120019
6 2018120034	2021	1A		96 X		CSE130	20120154
7 2018120034	2021	1A		72 X		CSE131	20130019
8 2018120034	2021	1A		65 X		CSE132	20130056
9 2014172008	2021	1A				CSE242	20091015
10 2014172008	2021	1A				CSE244	20091039
11 2014172008	2021	1A				CSE341	20092019
12 2014172008	2021	1A				CSE430	20093032

학번을 기준으로 그룹핑 ➡ 신청한 과목수와 평균 계산

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

Oracle SQL Developer : 테이블 STUDY01.수강@19c_remote

파일(F) 편집(E) 보기(V) 이동(N) 실행(R) 팀(M) 도구(I) 창(W) 도움말(H)

정속

19c_remote

데이터 Model 제약 조건 권한 부여 통계 트리거 플래시백 증속성 세부정보 분할 영역 인덱스 SQL

수강

시작 페이지 19c_remote 정렬... 필터:

학번	수강년도	수강학기	이수구분	취득점수	재수강여부	교과목코드	교수사번
1 2018120034	2021	1A		94 X		ARB263	20091015
2 2018120034	2021	1A		94 X		CSE114	20091039
3 2018120034	2021	1A		88 X		CSE120	20092019
4 2018120034	2021	1A		83 X		CSE121	20093032
5 2018120034	2021	1A		91 X		CSE122	20120019
6 2018120034	2021	1A		96 X		CSE130	20120154
7 2018120034	2021	1A		72 X		CSE131	20130019
8 2018120034	2021	1A		65 X		CSE132	20130056
9 2014172008	2021	1A		74 X		CSE242	20091015
10 2014172008	2021	1A		84 X		CSE244	20091039
11 2014172008	2021	1A		98 X		CSE341	20092019
12 2014172008	2021	1A		87 X		CSE430	20093032

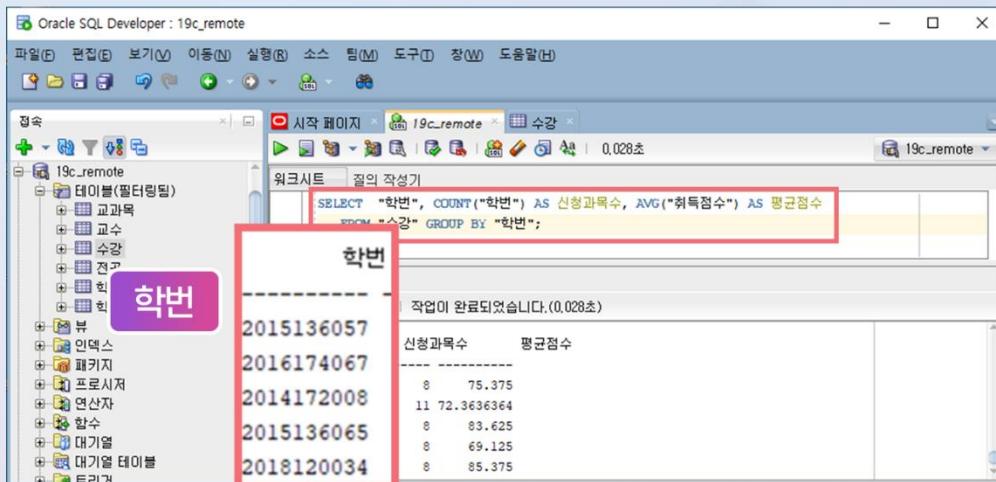
학번을 기준으로 그룹핑 ➡ 신청한 과목수와 평균 계산

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출



The screenshot shows the Oracle SQL Developer interface. On the left, the schema browser displays a table named '19c_remote' with various columns like '교과목', '교수', '수강', etc. A red box highlights the column '학번'. In the central workspace, a query is written:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번";
```

The results pane shows the output of the query:

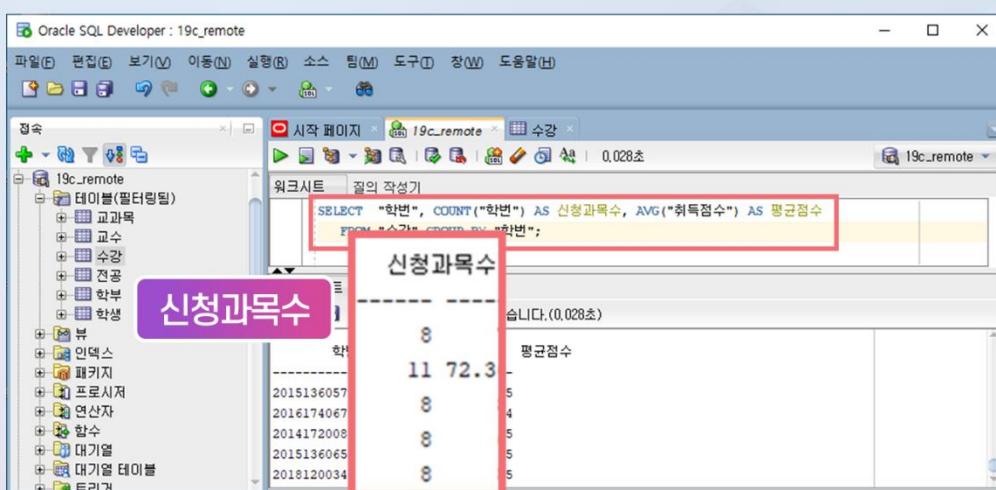
학번	신청과목수	평균점수
2015136057	8	75.375
2016174067	11	72.3636364
2014172008	8	83.625
2015136065	8	69.125
2018120034	8	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번"
```

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출



The screenshot shows the Oracle SQL Developer interface. On the left, the schema browser displays a table named '19c_remote' with various columns like '교과목', '교수', '수강', etc. A red box highlights the column '학번'. In the central workspace, a query is written:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번";
```

The results pane shows the output of the query:

학번	신청과목수	평균점수
2015136057	8	72.3
2016174067	8	5
2014172008	8	4
2015136065	8	5
2018120034	8	5

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번"
```

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

GROUP BY 절 예제 – 학번별로 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of database objects under '19c_remote'. In the center, a query window displays the following SQL code:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번";
```

The results of the query are shown in a table on the right. A red box highlights the column headers '신청과목수' and '평균점수'. Another red box highlights the '평균점수' column values, which are: 75.375, 3636364, 83.625, 69.125, and 85.375.

신청과목수	평균점수
75.375	75.375
3636364	3636364
83.625	83.625
69.125	69.125
85.375	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번"
```

데이터베이스 검색

검색 순서 및 조건

HAVING 절

HAVING 절

- HAVING은 GROUP BY 결과에서 그룹 조건 검색을 할 때 사용
 - ➡ WHERE를 SELECT 문에서 개별 행에 대한 조건 검색을 위해 사용하는 것과 같음
- SELECT 문에는 WHERE 절과 HAVING 절이 모두 사용될 수 있음

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

HAVING 절

HAVING 절의 실행 순서

- WHERE 절에 조건에 만족하는 행을 선택
- 선택된 행을 GROUP BY 절에 의해 그룹핑
- 그룹핑 된 결과에서 HAVING 절의 조건을 만족하는 결과 선택

	WHERE	HAVING
동작	테이블의 행에 대한 조건	GROUP BY에 대한 조건
검색	컬럼	그룹 함수

데이터베이스 검색

검색 순서 및 조건

HAVING 절

HAVING 절의 실행 순서

- WHERE 절에 조건에 만족하는 행을 선택
- 선택된 행을 GROUP BY 절에 의해 그룹핑
- 그룹핑 된 결과에서 HAVING 절의 조건을 만족하는 결과 선택

	WHERE	HAVING
동작	테이블의 행에 대한 조건	GROUP BY에 대한 조건
검색	컬럼	그룹 함수

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

HAVING 절 예제 – 취득점수 80점 이상을 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays a database structure for '19c_remote' containing tables like '교과목', '교수', '수강', etc. The central workspace shows a SQL editor with the following query:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번"
HAVING AVG("취득점수") > 80;
```

A red box highlights the 'GROUP BY' and 'HAVING' clauses. A purple box labeled '학번별로 그룹핑' (Group by grade) is overlaid on the editor area. The results pane below shows the output:

학번	신청과목수	평균점수
2014172008	8	83.625
2018120034	8	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번"
HAVING AVG("취득점수") > 80;
```

데이터베이스 검색

검색 순서 및 조건

HAVING 절 예제 – 취득점수 80점 이상을 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays a database structure for '19c_remote' containing tables like '교과목', '교수', '수강', etc. The central workspace shows a SQL editor with the following query:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번"
HAVING AVG("취득점수") > 80;
```

A red box highlights the 'HAVING' clause. A purple box labeled '취득점수 80점 이상을 그룹핑' (Group by average score of 80 or more) is overlaid on the editor area. The results pane below shows the output:

학번	신청과목수	평균점수
2014172008	8	83.625
2018120034	8	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강"
GROUP BY "학번"
HAVING AVG("취득점수") > 80;
```

검색 순서 및 조건

데이터베이스 검색

검색 순서 및 조건

HAVING 절 예제 – 취득점수 80점 이상을 그룹핑하여 데이터 추출

The screenshot shows the Oracle SQL Developer interface. On the left, the database structure for '19c_remote' is visible, including tables like 교과목, 교수, 수강, 전공, 학부, 학생, and various indexes and statistics. The central workspace displays a query in the '작성기' (Editor) pane:

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번" HAVING AVG("취득점수") > 80;
```

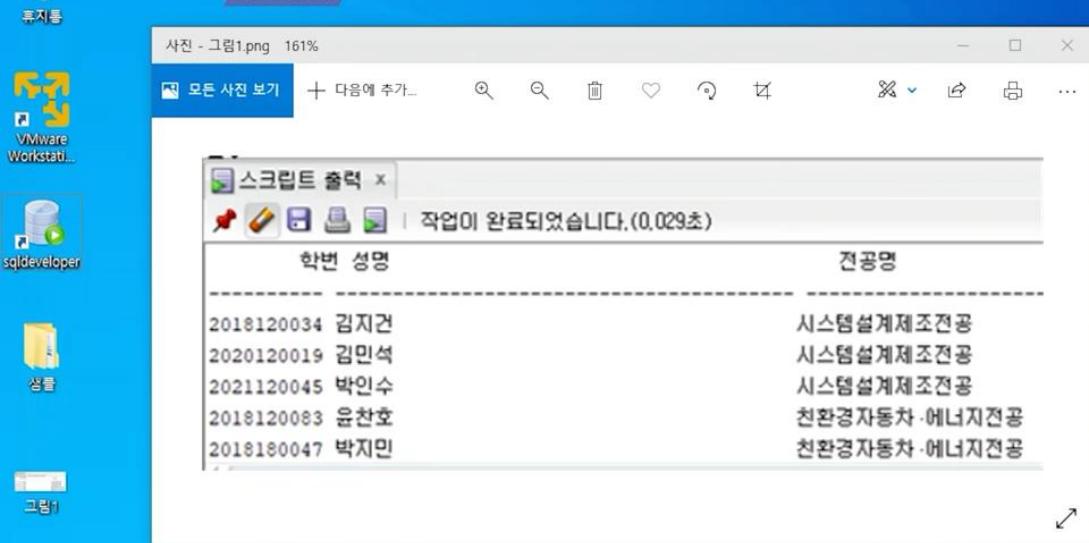
The results are shown in a table in the '결과' (Results) pane:

학번	신청과목수	평균점수
2014172008	8	83.625
2018120034	8	85.375

```
SELECT "학번", COUNT("학번") AS 신청과목수, AVG("취득점수") AS 평균점수
FROM "수강" GROUP BY "학번" HAVING AVG("취득점수") > 80;
```

실습하기

데이터베이스 검색 실습하기



데이터 검색 실습하기

실습단계
데이터 검색 실습하기
학번, 성명 전공명을 출력하기
Oracle 서버 실행 확인
SQL Developer 실행
Oracle 서버(19c_remote)에 원격 접속
전공 테이블 클릭
학생 테이블 클릭
학번, 성명, 전공코드 등이 저장
SELECT 뒤에 조회하고자 하는 학번, 성명, 전공명 지정
FROM 뒤에 사용할 테이블 지정
검색 조건 지정
학생 테이블의 전공코드와 전공테이블의 전공코드가 일치
스크립트 실행