

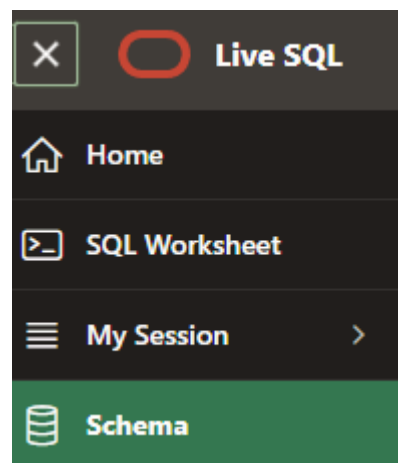
Day28_20220808_Oracle

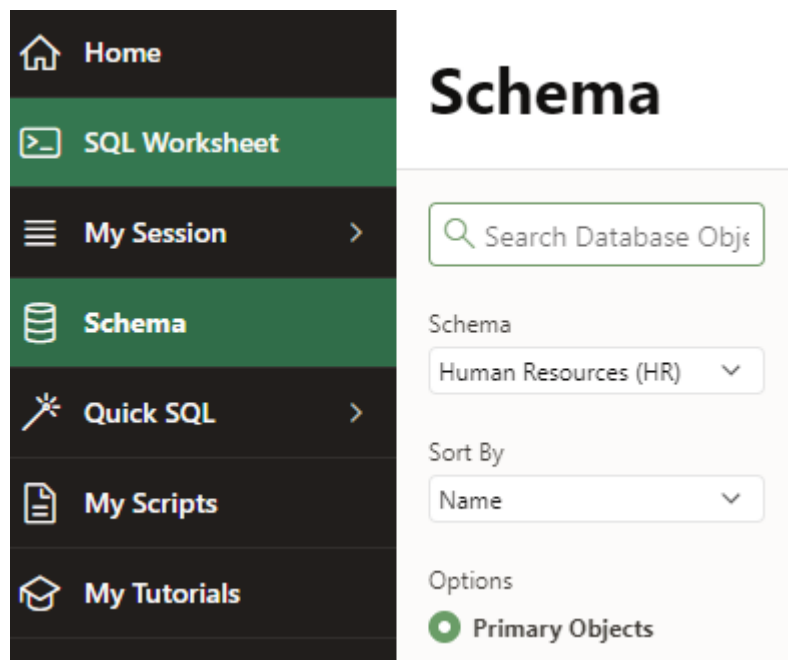
🕒 작성일시	@2022년 8월 8일 오후 5:35
▼ 강의 번호	
📅 유형	@2022년 8월 8일
🔗 자료	
☑ 복습	<input type="checkbox"/>
☰ 속성	

Oracle

오라클 로그인 하고 들어감

스키마 클릭





휴먼 리소스 Menu 들어감

You have read-only access to the **HR** sample schema.

ADD_JOB_HISTORY p() Procedure Status: Valid Created 4.5 years ago	COUNTRIES [Table] Table Status: Valid Created 4.5 years ago	DEPARTMENTS [Table] Table Status: Valid Created 4.5 years ago
DEPARTMENTS_SEQ 1/2 ↓ Sequence Status: Valid Created 4.5 years ago	EMPLOYEES [Table] Table Status: Valid Created 4.5 years ago	EMPLOYEES_SEQ 1/2 ↓ Sequence Status: Valid Created 4.5 years ago
EMP_DETAILS_VIEW [View] View Status: Valid Created 4.5 years ago	JOBS [Table] Table Status: Valid Created 4.5 years ago	JOB_HISTORY [Table] Table Status: Valid Created 4.5 years ago
LOCATIONS [Table] Table Status: Valid Created 4.5 years ago	LOCATIONS_SEQ 1/2 ↓ Sequence Status: Valid Created 4.5 years ago	REGIONS [Table] Table Status: Valid Created 4.5 years ago
SECURE_DML p() Procedure Status: Valid Created 4.5 years ago		

EMPLOYEES 다운로드.

Schema \

EMPLOYEES

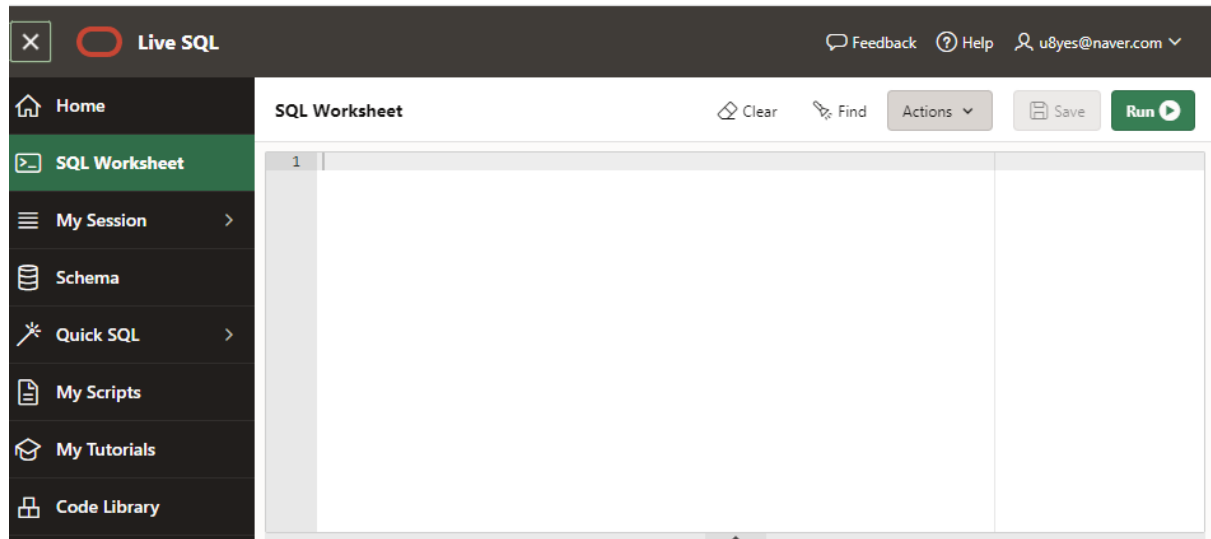
Syntax Help Actions View All Objects

Show All Table Attributes Columns Indexes Triggers Constraints

Columns

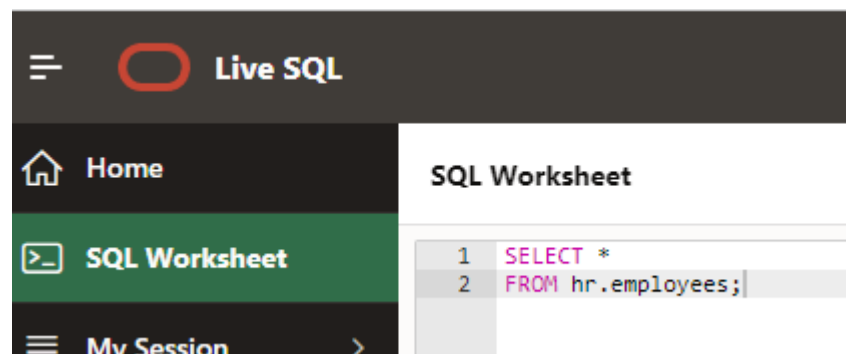
#	Column	Type	Length	Precision	Scale	Nullable	Semantics	Comment
---	--------	------	--------	-----------	-------	----------	-----------	---------

이제 실행할 차례



test 해보자!

구문을 작성한 후에 Ctrl + Enter를 누르면 실행!



Oracle Database

1. SELECT 구문

SELECT 은 데이터를 조회만 하여 화면에 출력할 뿐이지 실제 데이터 값이 변경되거나 삭제되는 것은 아니다.

SELECT 열이름(or 별명 alias)

FROM 테이블 이름

SELECT 열이름(or 별명 alias)

FROM 테이블 이름

[WHERE 조건식]

[ORDER BY 정렬기준열]

- SQL 은 기본적으로 대소문자를 가리지 않는다. SELECT 과 select 을 동일하게 인식.
- SQL 구문은 대문자로 입력하는 것이 관례이다. 열이름 등은 소문자로... 관례
- SQL 구문은 한줄 또는 여러줄로 작성할 수 있다. SELECT 열 이름 FROM 테이블 이름;
- SQL 구문은 들여쓰기도 상관하지 않지만 관례적으로 사용을 권장한다.

employees 테이블의 모든 정보를 출력하자.

(로컬이 아니고 인터넷 무료 버전이라 hr.을 붙인 것임)

```
1 SELECT *
2 FROM hr.employees;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_
100	Steven	King	SKING	515.123.4567	17-JU
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SE

‘ * 에스터리스크’는 ‘모든’을 의미한다.

따라서 위의 구문은 hr데이터베이스의 employees테이블에서 모든 데이터를 조회하라. 라는 의미이다. (hr은 생략 가능하다)

원하는 열들만 조회하기.

Quiz 2. employees 테이블에서 employee_id, first_name, last_name 을 출력하세요

SQL Worksheet

```
1 SELECT employee_id, first_name, last_name
2 FROM hr.employees;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME
100	Steven	King
101	Neena	Kochhar
102	Lex	De Haan
103	Alexander	Hunold
104	Bruce	Ernst
105	David	Austin
106	Valli	Pataballa
107	Diana	Lorentz
108	Nancy	Greenberg
109	Daniel	Faviet
110	John	Chen
111	Ismael	Sciarra
112	Jose Manuel	Urman
113	Luis	Popp
114	Den	Raphaely
115	Alexander	Khoo

SELECT 명령문 뒤에 열들의 이름을 나열하면 순서대로 조회된 결과를 확인할 수 있다.

만약 모든 열을 조회할 때도 * 대신에 모든 열들의 이름을 전부 나열하면 같은 결과를 볼 수 있다.

ORDER BY 정렬

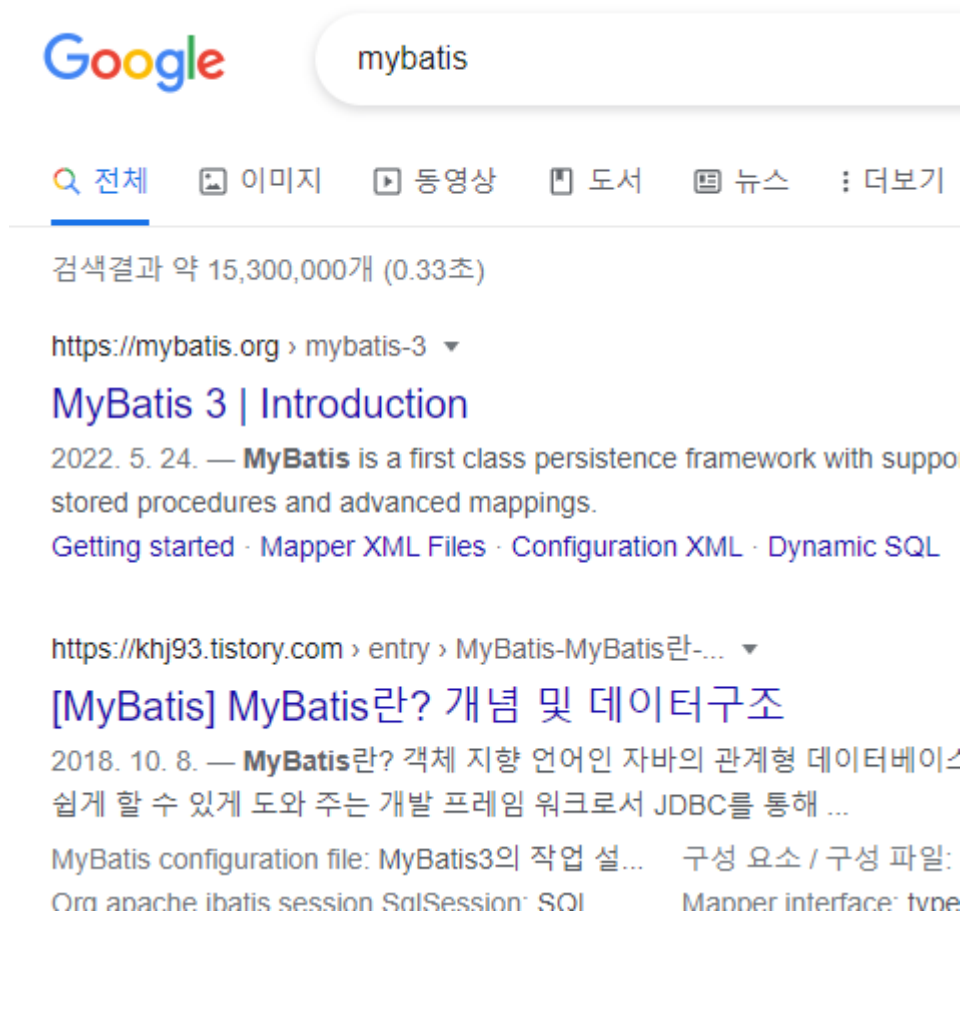
Quiz 3. employees 테이블에서 employee_id, first_name, last_name 을 출력하는데 employee_id 를 기준으로 내림차순 정렬하세요

SQL Worksheet		
1	SELECT employee_id, first_name, last_name	
2	FROM hr.employees	
3	ORDER BY employee_id DESC;	

EMPLOYEE_ID	FIRST_NAME	LAST_NAME
206	William	Gietz
205	Shelley	Higgins
204	Hermann	Baer
203	Susan	Mavris
202	Pat	Fay
201	Michael	Hartstein
200	Jennifer	Whalen
199	Douglas	Grant
198	Donald	OConnell
197	Kevin	Feeney
196	Alana	Walch

정렬 순서를 생략하면 기본 순서는 오름차순이고, 내림차순은 DESC를 붙여준다.

SQL이 한 두개가 아니다.



DISTINCT 명령어

SQL 연산이나 보고서를 작성할 때 데이터 값의 행에 중복된 데이터를 제거하고 출력해야 할 때도 있다. 이때 사용되는 것이 DISTINCT 명령어이다.

SQL Worksheet

```
1 SELECT DISTINCT job_id  
2 FROM hr.employees;  
3
```

JOB_ID
AC_ACCOUNT
AC_MGR
AD_ASST
AD_PRES
AD_VP
FI_ACCOUNT
FI_MGR
HR_REP
IT_PROG
MK_MAN
MK_REP
PR_REP
PU_CLERK
PU_MAN
SA_MAN
SA_REP
SH_CLERK
ST_CLERK
ST_MAN

[Download CSV](#)

DISTINCT 명령어를 사용하면 데이터 값이 종류별로 하나만 출력된 것을 확인할 수 있다. DISTINCT 명령어 뒤에 계속 이름을 나열하면 그 순서대로 DISTINCT 가 모두 적용된다.

별명 사용 Alias

원래 이름 대신에 임의로 잠시 열의 이름을 바꿔서 사용하고자 할 때 쓴다.

AS 접속사를 사용하여 이름을 바꾸는 것이 기본이나 생략하고 바로 별명을 붙이는 경우도 많다.

- 별명은 열 이름을 임시로 변경한다. 원래의 이름이 진짜 변경되는 것은 아니다.
- 별명은 열이름 바로 뒤에 사용하고 AS로 붙이지만 AS는 생략 가능하다.
- 별명에는 공백, 특수문자, 대문자를 사용하려면 큰 따옴표 “ ”로 묶어서 사용할 수 있다.

실무에서는 보통 열의 이름을 의미가 있는 단어보다는 코드 형태로 열의 이름을 관리하는 경우가 많다. `employee_id` → `e1`, 따라서 출력시에는 알아보기 쉽게 의미가 있는 단어 형태로 출력하는 것이 바람직하다.

Quiz 5 employees 테이블에서 `employee_id` 를 ‘사원번호’, `first_name` 은 ‘이름’ `last_name`을 성으로 출력하세요

SQL Worksheet

Clear

```

1 SELECT employee_id AS 사원번호, first_name AS 이름, last_name AS 성
2 FROM hr.employees;
3

```

사원번호	이름	성
100	Steven	King
101	Neena	Kochhar
102	Lex	De Haan
103	Alexander	Hunold
104	Bruce	Ernst
105	David	Austin
106	Valli	Pataballa
107	Diana	Lorentz
108	Nancy	Greenberg
109	Daniel	Faviet
110	John	Chen
111	Ismael	Sciarra

데이터 값 연결

연결 연산자 || (MySQL에서는 사용되지 않음)

기존의 열들의 값을 묶어서 하나로 출력 또는 기존 열의 값과 글자를 붙여서 표현하는 방법

열이름 || 열이름 or 열이름 || “문자”

quiz 7

employees 테이블에서 employee_id를 출력하고 first_name과 last_name을 붙여서 출력하세요.

SQL Worksheet

```
1 SELECT employee_id, first_name || last_name || '씨'
2 FROM hr.employees;
```

EMPLOYEE_ID	FIRST_NAME LAST_NAME '씨'
100	StevenKing씨
101	NeenaKochhar씨
102	LexDe Haan씨
103	AlexanderHunold씨
104	BruceErnst씨

```
SELECT employee_id, first_name || last_name || '씨'
FROM hr.employees;
```

연결 연산자

```
1 SELECT employee_id, concat(concat(first_name, ' '), last_name) AS "full name"
2 FROM hr.employees;
```

EMPLOYEE_ID	full name
100	Steven King
101	Neena Kochhar
102	Lex De Haan

```
SELECT employee_id, concat(concat(first_name, ' '), last_name) AS "full name"
FROM hr.employees;
```

산술 계산

Quiz 8

employee 테이블에서 employee_id, salary, salary에 500을 더한값, salary에 500을 뺀 값, salary의 10%을 추가한 값을 출력하시오.

SQL Worksheet



```
1 SELECT employee_id, salary, salary + 500, salary - 500, salary * 1.1
2 FROM hr.employees;
```

EMPLOYEE_ID	SALARY	SALARY+500	SALARY-500	SALARY*0.1
100	24000	24500	23500	2400
101	17000	17500	16500	1700
102	17000	17500	16500	1700

```
SELECT employee_id, salary, salary + 500, salary - 500, salary * 1.1
FROM hr.employees;
```

WHERE 조건 절

행들의 특정 데이터 값을 조회하거나 비교하여 연산 처리하는 방법

- 조회하려는 조건을 지정할 수 있다.
- 반드시 FROM 다음에 온다.
- 조건절에는 비교 논리 연이름 문자열 등이 올 수 있다.

SELECT 열이름

FROM 테이블 이름

WHERE 조건;

어느 테이블로부터 조건에 맞는 값으로 해당 열들을 선택하여 조회한다.

from

where

select

employees 테이블에서 first_name 이 David인 직원의 정보를 출력하세요.

SQL Worksheet

Clear Find Actions Save Run

```
1 SELECT *
2 FROM hr.employees
3 WHERE first_name = 'David';
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	-	103	60
151	David	Bernstein	DBERNSTE	011.44.1344.345268	24-MAR-05	SA_REP	9500	.25	145	80
165	David	Lee	DLEE	011.44.1346.529268	23-FEB-08	SA_REP	6800	.1	147	80

```
SELECT *
FROM hr.employees
WHERE first_name = 'David';
```

Quiz 12

employees 테이블에서 employees, employee_id 105 이상인 직원의 정보를 출력하세요

SQL Worksheet

```
1 SELECT *
2 FROM hr.employees
3 WHERE employee_id >= 105;
```



EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID
105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG
106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG
107	Diana	Lorentz	DLORENTZ	590.423.5567	07-FEB-07	IT_PROG


```
SELECT *
FROM hr.employees
WHERE employee_id ≥ 105;
```

Quiz 13

employees테이블에서 salary 가 10000 이상이고, 20000 이하인 직원들 정보를 표시하세요.

SQL Worksheet

 Clear 

```
1 SELECT *
2 FROM hr.employees
3 WHERE salary >= 10000 AND salary <= 20000;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	C
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	

```
SELECT *
FROM hr.employees
WHERE salary >= 10000 AND salary <= 20000;
```

```

1 SELECT *
2 FROM hr.employees
3 WHERE salary BETWEEN 10000 AND 20000;

```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	-
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	-
108	Nancy	Greenberg	NGREENBE	515.124.4569	17-AUG-02	FI_MGR	12008	-
114	Den	Raphaely	DRAPHEAL	515.127.4561	07-DEC-02	PU_MAN	11000	-
145	John	Russell	JRUSSEL	011.44.1344.429268	01-OCT-04	SA_MAN	14000	.4
146	Karen	Partners	KPARTNER	011.44.1344.467268	05-JAN-05	SA_MAN	13500	.3

```

SELECT *
FROM hr.employees
WHERE salary BETWEEN 10000 AND 20000;

```

Quiz 14

employees 테이블에서 salary 가 10000, 17000, 24000인 직원 정보를 표시하세요

SQL Worksheet



```
1 SELECT *
2 FROM hr.employees
3 WHERE salary IN(10000, 17000, 24000);
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID
100	Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP
150	Peter	Tucker	PTUCKER	011.44.1344.129268	30-JAN-05	SA_REP
156	Janette	King	JKING	011.44.1345.429268	30-JAN-04	SA_REP
169	Harrison	Bloom	HBLOOM	011.44.1343.829268	23-MAR-06	SA_REP

```
SELECT *
FROM hr.employees
WHERE salary IN(10000, 17000, 24000);
```

Quiz 15

employees 테이블에서 job_id 에서 AD를 포함하는 모든 직원 정보를 표시하세요



```

1 SELECT *
2 FROM hr.employees
3 WHERE job_id LIKE '%AD%';

```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALA
100	Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES	2400
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	1700
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	1700
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-SEP-03	AD_ASST	4400

```

SELECT *
FROM hr.employees
WHERE job_id LIKE '%AD%';

```

- LIKE %와 _ 같은 기호와 함께 사용된다.
- '%' 는 모든 문자, '_'는 한글자 라는 의미이다.

Quiz 16

employees 테이블에서 job_id 에서 AD를 포함하면서 AD뒤에 3글자인 데이터 값을 갖는 모든 직원 정보를 표시하세요

SQL Worksheet

Clear



```
1 SELECT *
2 FROM hr.employees
3 WHERE job_id LIKE 'AD_____';
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	-
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	-

Download CSV

```
SELECT *
FROM hr.employees
WHERE job_id LIKE 'AD_____';
```

Quiz 17

employees 테이블에서 salary가 4000을 초과하면서 job_id 가 'IT_PROG' 인 모든 직원 정보를 표시하세요

SQL Worksheet

Clear

Find

Actions



```
1 SELECT *
2 FROM hr.employees
3 WHERE salary > 4000 AND job_id LIKE 'IT_PROG';
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	-	102	60
104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	-	103	60
105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	-	103	60
106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG	4800	-	103	60
107	Diana	Lorentz	DLORENTZ	590.423.5567	07-FEB-07	IT_PROG	4200	-	103	60

SQL Worksheet

Clear

Find

```

1 SELECT *
2 FROM hr.employees
3 WHERE salary > 4000
4 AND job_id IN 'IT_PROG';

```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	-
104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	-
105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	-
106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG	4800	-
107	Diana	Lorentz	DLORENTZ	590.423.5567	07-FEB-07	IT_PROG	4200	-

```

SELECT *
FROM hr.employees
WHERE salary > 4000
AND job_id IN 'IT_PROG';

```

Quiz 18

employees 테이블에서 salary가 4000을 초과하면서 job_id가 IT_PROG 이거나 FI_ACCOUNT 인 모든 직원 정보를 표시하세요

SQL Worksheet

Clear

Find

Actions ▾

Save

```

1 SELECT *
2 FROM hr.employees
3 WHERE salary > 4000
4 AND job_id = 'IT_PROG'
5 OR job_id = 'FI_ACCOUNT';

```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	-	102	60
104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	-	103	60
105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	-	103	60
106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG	4800	-	103	60

```
SELECT *
FROM hr.employees
WHERE salary > 4000
AND job_id = 'IT_PROG'
OR job_id = 'FI_ACCOUNT';
```

Quiz 19

employees 테이블에서 manager_id 가 null 값이 아닌 모든 직원 정보를 표시하세요

SQL Worksheet

Clear

Find

Actions

Save

Run

```
1 SELECT *
2 FROM hr.employees
3 WHERE manager_id is not null;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	-	100	90
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	-	100	90
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	-	102	60

```
SELECT *
FROM hr.employees
WHERE manager_id is not NULL;
```

많이 사용되는 함수들...

단일행 함수 - 각각의 행들의 값에 적용되는 함수, 한번에 한행씩 전부 처리

다중행 함수 (그룹함수) - 여러 행들을 사용한 결과 값을 반환하는 함수 sum

자료형 복습

CHAR - 고정 문자열

VARCHAR2 - 가변 문자열

NUMBER - 숫자 (정수와 실수를 함께 표현)

DATE - 날짜

단일행 함수

- LOWER 소문자로
- UPPER 대문자로
- INITCAP 첫 글자만 대문자로 (MySQL에는 없는 함수)
- SUBSTR 일부 글자만 추출
- REPLACE 특정 문자열 교환
- CONCAT 문자열 연결 (|| 연결 연산자와 같다. 단, 주의할 것은 2개만 연결할 수 있다.)
- LENGTH 문자열의 길이
- INSTR 문자의 위치 즉 인덱스 반환
- LPAD 왼쪽부터 특정 문자로 채움
- RPAD 오른쪽부터 특정 문자로 채움
- LTRIM 왼쪽 문자 삭제
- RTRIM 오른쪽 문자 삭제

LOWER / UPPER / INITCAP

Quiz 17

employees 테이블에서 last_name 을 소문자와 대문자로 각각 출력하고 email의 첫 번째 문자는 대문자로 출력하세요

LAST_NAME LOWER적용 UPPER적용 EMAIL INITCAP 적용

SQL Worksheet

 Clear

```
1 SELECT last_name, LOWER(last_name), UPPER(last_name), email, INITCAP(email)
2 FROM hr.employees;
```

LAST_NAME	LOWER(LAST_NAME)	UPPER(LAST_NAME)	EMAIL	INITCAP(EMAIL)
Abel	abel	ABEL	EABEL	Eabel
Ande	ande	ANDE	SANDE	Sande
Atkinson	atkinson	ATKINSON	MATKINSO	Matkinso
Austin	austin	AUSTIN	DAUSTIN	Daustin
Baer	baer	BAER	HBAER	Hbaer
Baida	baida	BAIDA	SBAIDA	Sbaida
Banda	banda	BANDA	ABANDA	Abanda
Bates	bates	BATES	EBATES	Ebates
Bell	bell	BELL	SBELL	Sbell
Bernstein	bernstein	BERNSTEIN	DBERNSTE	Dbernste
Bissot	bissot	BISSOT	LBISSOT	Lbissot
Bloom	bloom	BLOOM	HBLOOM	Hbloom

```
SELECT last_name, LOWER(last_name), UPPER(last_name), email, INITCAP(email)
FROM hr.employees;
```

Quiz 18

employees 테이블에서 job_id 값의 첫째 자리부터 2개의 문자만 추출하세요.

JOB_ID 적용 결과

AD_VP AD

SQL Worksheet

```
1 SELECT SUBSTR(job_id, 1, 2)|  
2 FROM hr.employees;
```

SUBSTR(JOB_ID,1,2)
AC
AC
AD
AD
AD
AD
FI
FT

```
SELECT SUBSTR(job_id, 1, 2)  
FROM hr.employees;
```

기본문법

SUBSTR(열이름, 시작, 끝)

Quiz 19

employees 테이블에서 job_id에서 ACCOUNT를 전부 ACCNT로 출력하세요.

SQL Worksheet

```
1 SELECT REPLACE(job_id, 'ACCOUNT', 'ACCNT')
2 FROM hr.employees;
```

REPLACE(JOB_ID, 'ACCOUNT', 'ACCNT')
AC_ACCNT
AC_MGR
AD_ASST
AD_PRES
AD_VP
AD_VP
FI_ACCNT
FI_ACCNT
FI_ACCNT
FI_ACCNT
FI ACCNT

```
SELECT REPLACE(job_id, 'ACCOUNT', 'ACCNT')
FROM hr.employees;
```

기본 문법 REPLACE(열이름, 찾는 값, 바꿀값)

LPDA / RPAD

Quiz 20

employees 테이블에서 first_name 에 12자리의 문자열로 만들어 12보다 글자수가 작으면 왼쪽부터 * 채워 출력 하고 열이름은 'LPAD적용결과'지정하고 오른쪽부터 #채워 출력하고 열이름은 'RPAD적용결과'로 지정하여 출력하세요

```

1 SELECT LPAD(first_name,12,'*'), RPAD(first_name,12,'#'), LPAD(first_name,12,' ')
2 FROM hr.employees;
3
4

```

LPAD(FIRST_NAME,12,'*')	RPAD(FIRST_NAME,12,'#')	LPAD(FIRST_NAME,12,'')
*****Ellen	Ellen#####	Ellen
*****Sundar	Sundar#####	Sundar
*****Mozhe	Mozhe#####	Mozhe
*****David	David#####	David
*****Hermann	Hermann#####	Hermann
*****Shelli	Shelli#####	Shelli

```

SELECT LPAD(first_name, 12, '*'), RPAD(first_name, 12, '#'), LPAD(first_name, 12, ' ')
FROM hr.employees;

```

Quiz 21

employees 테이블에서 job_id 열의 값에 대해 왼쪽에서 부터 'F'문자를 만나면 삭제하고 오른쪽 으로부터 'T'문자를 만나면 삭제하시오.

JOB_ID LTRIM적용결과 RTRIM적용결과

SQL Worksheet

```
1 SELECT JOB_ID, LTRIM(JOB_ID, 'F'), RTRIM(JOB_ID, 'T')|
2 FROM hr.employees;
```


JOB_ID	LTRIM(JOB_ID, 'F')	RTRIM(JOB_ID, 'T')
AC_ACCOUNT	AC_ACCOUNT	AC_ACCOUN
AC_MGR	AC_MGR	AC_MGR
AD_ASST	AD_ASST	AD_ASS
AD_PRES	AD_PRES	AD_PRES
AD_VP	AD_VP	AD_VP
AD_VP	AD_VP	AD_VP
FI_ACCOUNT	I_ACCOUNT	FI_ACCOUN
FI_ACCOUNT	I_ACCOUNT	FI_ACCOUN
FI_ACCOUNT	I_ACCOUNT	FI_ACCOUN
FI_ACCOUNT	I_ACCOUNT	FI_ACCOUN
FI_ACCOUNT	I_ACCOUNT	FI_ACCOUN

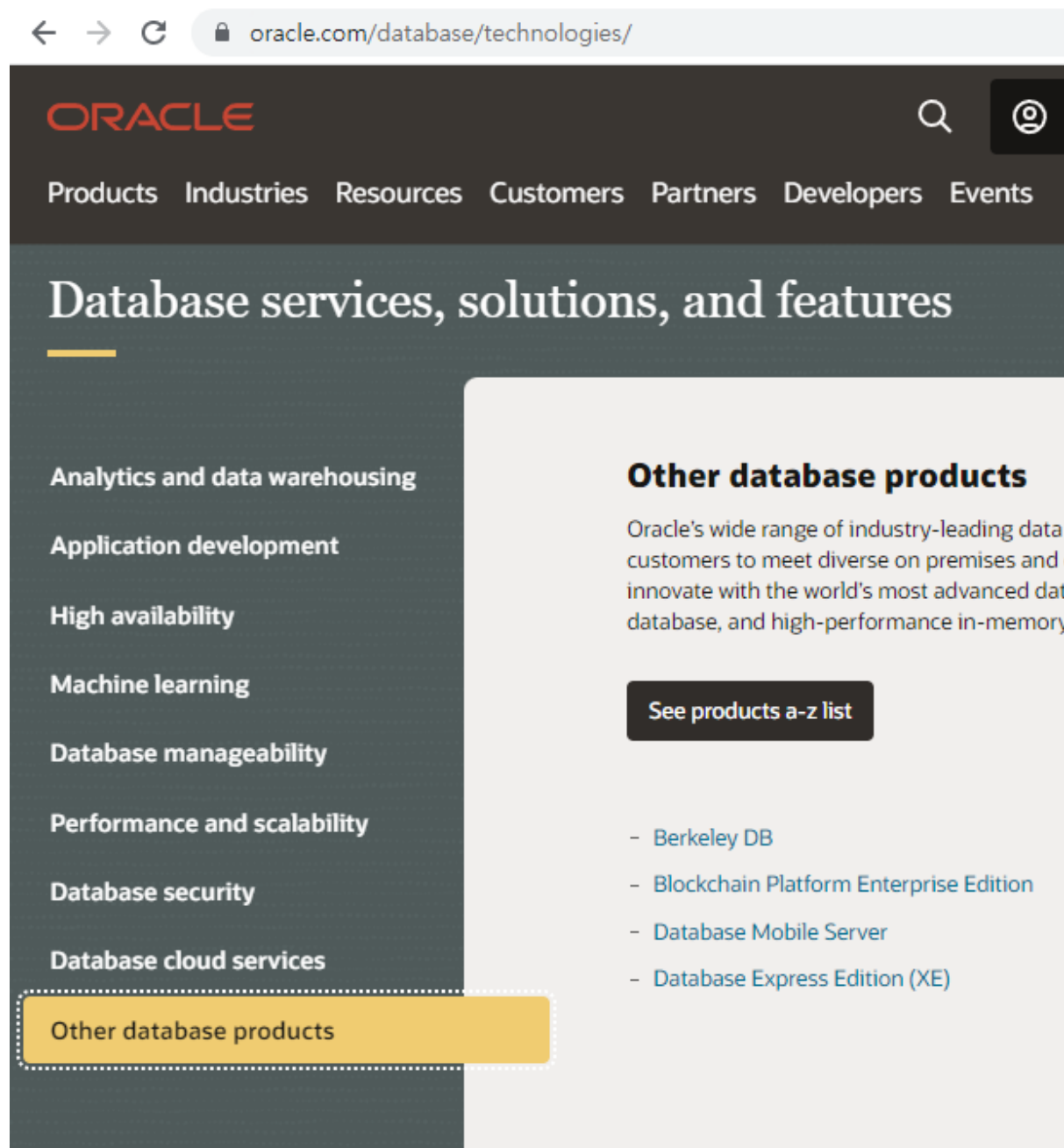
```
SELECT JOB_ID, LTRIM(JOB_ID, 'F'), RTRIM(JOB_ID, 'T')
FROM hr.employees;
```

Oracle 다운로드

Oracle Database Express Edition (XE) Downloads


Oracle Database XE Downloads Oracle Database 21c Express Edition Support Oracle Database Express Edition (XE) is a community supported edition of the Oracle Database family. Please go to the Oracle Database XE Community Support Forum for help, feedback, and enhancement requests.

 <https://www.oracle.com/database/technologies/xe-downloads.html>



Free Oracle Database for Everyone → [Download](#) Oracle Database XE

Database / Technologies / Application Development /
Oracle Database XE






Free Oracle Database for Everyone

Whether you are a developer, a DBA, a data scientist, an educator, or just curious about data, Oracle Database Express Edition (XE) is the ideal way to get started. It is the same powerful Database that enterprises rely on worldwide, packaged for simple download, ease-of-use, a featured experience. You get an Oracle Database to use in any environment, plus the ability and redistribute – all completely free!

[Quick Start](#)

[Download Oracle Database XE](#)

윈도우 버전

Oracle Database XE Downloads	
Oracle Database 21c Express Edition	
Download	Description
 Oracle Database 21c Express Edition for Windows x64	(1,967,615,483 bytes - October 08, 2021) [Sha256sum: 939742c3305c466566a55f607638621b6aa7032
 Oracle Database 21c Express Edition for Linux x64 (OL8)	(2,339,651,768 bytes - September 08, 2021) [Sha256sum: f8357b432de33478549a76557e8c5220ec24371
 Oracle Database 21c Express Edition for Linux x64 (OL7)	(2,339,017,432 bytes - September 08, 2021) [Sha256sum: 4c8f40a19d4d1a2f00e46df022943a04cc13fe62

SQL Developer Downloads

Oracle SQL Developer Downloads

<https://www.oracle.com/database/sqldeveloper/technologies/download/>

개발자 도구

ADF Faces	Developer Tools for Visual Studio	Oracle REST Data Services(ORDS)	VM: Oracle VM Templates
Application Express(APEX)	Enterprise Pack for Eclipse	SOA Suite	VM: Pre-Built VirtualBox VMs
BI Publisher	Forms & Reports Services	Spatial Studio	Warehouse Builder
BPEL Process Manager	JavaScript Extension Toolkit	SQL Developer	WebRTC Session Controller Client SDK
Developer Studio	JDeveloper & ADF	SQL Developer Data Modeler	
Developer Suite 10g	NetBeans IDE	StorageTek	

oracle.com/database/sqldeveloper/technologies/download/

ORACLE Products Industries Resources Customers Partners Developers Events

Database / SQL Developer / SQL Developer Downloads

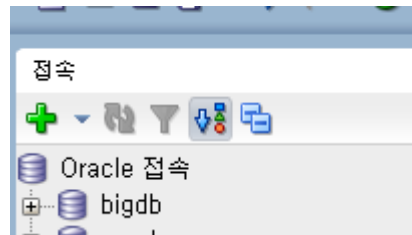
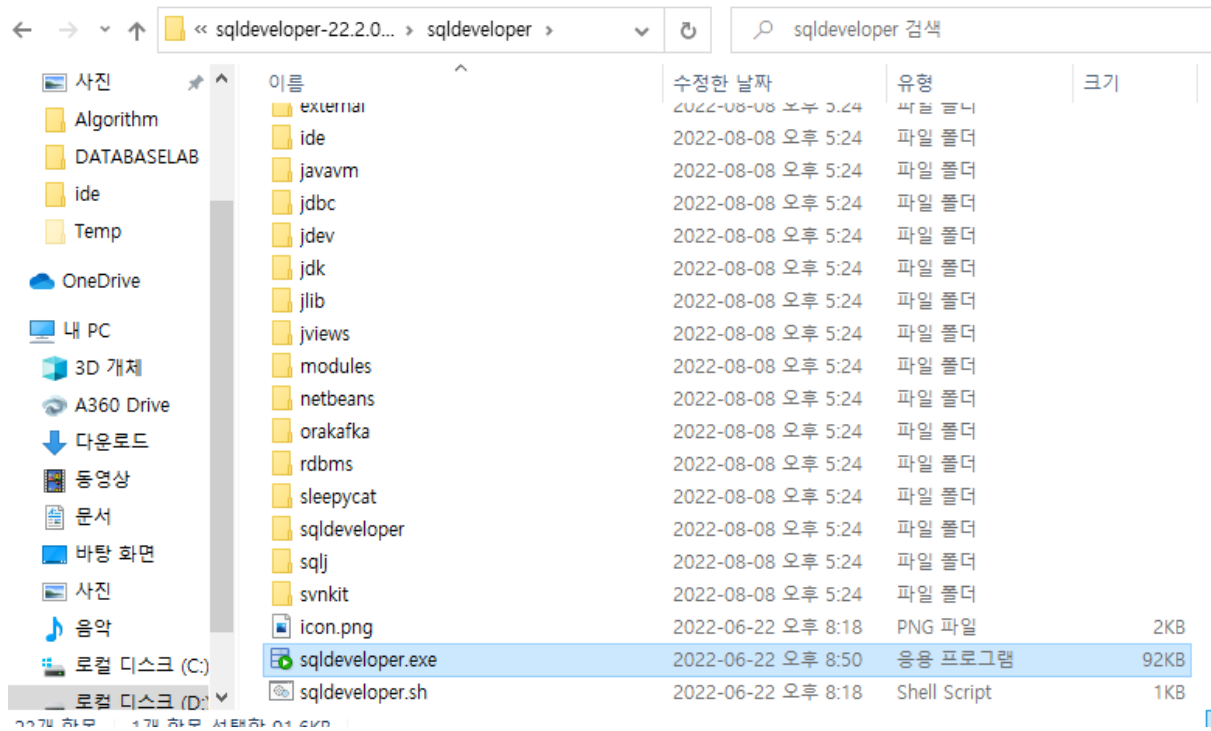
SQL Developer 22.2 Downloads

Version 22.2.0.173.2018 - June 28, 2022

- Release Notes
- Documentation

Platform	Download	Notes
Windows 64-bit with JDK 11 included	Download (434 MB)	<ul style="list-style-type: none">MD5: 3ba8e2cb9c2a21c08b7f72ef1d735ff0SHA1: 1842cdc4caf9f8b89e4d30ede5f3ddba06909Installation Notes
Windows 32-bit/64-bit	Download (480 MB)	<ul style="list-style-type: none">MD5: 79694563728868e5d3ae4364bca8c0e9SHA1: bca651f1efad5b7d93b1647ab614bdb999e7714Installation NotesJDK 11 required

실행함



test 해보자



