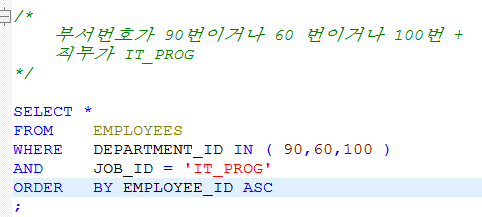
**로고학습일지**

**kt ds University 자바 기반의 데이터 사이언티스트 양성과정**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 학습일시 | 2018. 08. 01 (수) | 장소 | kt ds University B관 201호 | **시 간** | 09:00~18:00 |
| 학습범위 | UI/UX | | | | |
| 작 성 자 |  | | | **강 사** | 장민창 강사 |

|  |  |
| --- | --- |
| 학습안건 | SQL |

|  |  |
| --- | --- |
| 학습내용 | 내용 |

★**IN** 연산자

부서번호가 90번이거나 60 번이거나 100번 +

직무가 IT\_PROG

SELECT \*

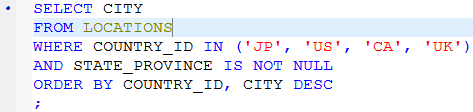
FROM EMPLOYEES

WHERE DEPARTMENT\_ID **IN** ( 90,60,100 )

AND JOB\_ID = 'IT\_PROG'

ORDER BY EMPLOYEE\_ID ASC

;

★**NOT** 연산자

SELECT CITY

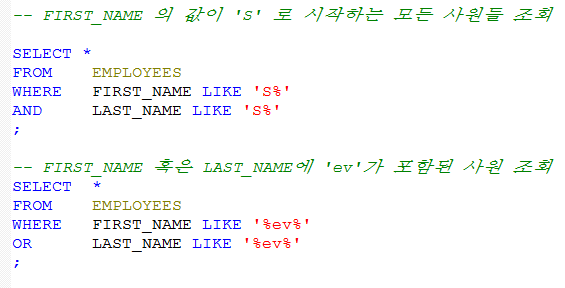
FROM LOCATIONS

WHERE COUNTRY\_ID IN ('JP', 'US', 'CA', 'UK')

AND STATE\_PROVINCE IS **NOT** NULL

ORDER BY COUNTRY\_ID, CITY DESC

;

와일드카드 %

★LIKE연산자 **%** (**NULL 제외**)

-- FIRST\_NAME 의 값이 'S' 로 시작하는

모든 사원들 조회

SELECT \*

FROM EMPLOYEES

WHERE FIRST\_NAME LIKE 'S**%**'

AND LAST\_NAME LIKE 'S**%**'

;

-- FIRST\_NAME 혹은 LAST\_NAME에 'ev'가 포함된 사원 조회

SELECT \*

FROM EMPLOYEES

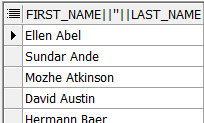
WHERE FIRST\_NAME LIKE '**%**ev**%**'

OR LAST\_NAME LIKE '**%**ev**%**'

;

★필드연결하기 **||** (한줄에)

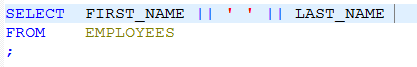
★ **AS** (보이게 하고 싶은 텍스트)

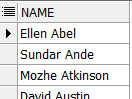


SELECT FIRST\_NAME **||** ' ' **||** LAST\_NAME

FROM EMPLOYEES

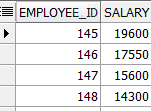
;







/\*

인센티브(COMMISSION\_PCT)를 받는 사람의 총 연봉구하기

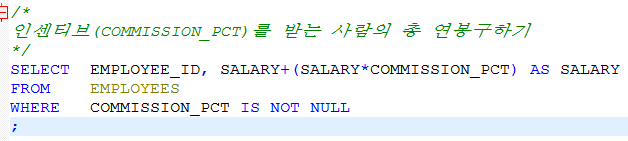
\*/

SELECT EMPLOYEE\_ID, SALARY+(SALARY\*COMMISSION\_PCT) **AS** SALARY

FROM EMPLOYEES

WHERE COMMISSION\_PCT IS NOT NULL

;



함수

★ (문자열을 조작하기 위한 문자함수)

★ **SYSDATE**: 현재 시각 (날짜와 시간을 조작하기 위한 날짜 함수)

**TO\_CHAR (문자로)**

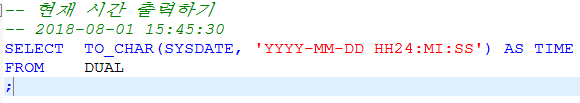
-- 현재 시간 출력하기

-- 2018-08-01 15:45:30

SELECT **TO\_CHAR**(SYSDATE, 'YYYY-MM-DD HH24:MI:SS') AS TIME

FROM DUAL

;





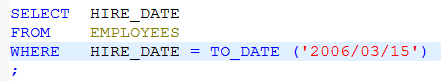
**TO\_DATE** (문자열을 날짜로)

SELECT HIRE\_DATE

FROM EMPLOYEES

WHERE HIRE\_DATE = **TO\_DATE** ('2006/03/15')

;



;

연동(SQL과 JAVA)

**public** **void** getCities(List<String> countryIdList) {

// 1. DB에 접근(HR)

// 오라클에 접근하기 위한 객체 Load

**try** {

Class.*forName*("oracle.jdbc.driver.OracleDriver");

} **catch** (ClassNotFoundException e) {

System.***out***.println(e.getMessage());

}

// Oracle(HR)접속

Connection conn = **null**;

PreparedStatement pstmt = **null**;

ResultSet rs = **null**;

**try** {

conn = DriverManager.*getConnection*("jdbc:oracle:thin:@localhost:1521:XE", "HR", "qweqwe"); << 패스워드

// 2. Query 준비 및 실행

// String sql = "SELECT \* FROM CITY";

pstmt = conn.prepareStatement("SELECT CITY FROM LOCATIONS WHERE COUNTRY\_ID IN ('?', '?', '?', '?')"

+ "AND STATE\_PROVINCE IS NOT NULL ORDER BY COUNTRY\_ID, CITY DESC");

pstmt.setString(1, countryIdList.get(0));

pstmt.setString(2, countryIdList.get(1));

pstmt.setString(3, countryIdList.get(2));

pstmt.setString(4, countryIdList.get(3));

// 3. Query 결과 출력

rs = pstmt.executeQuery();

//결과출력

**while** ( rs.next() ) {

String city = rs.getString("CITY");

System.***out***.printf(city);

}

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}

**finally** {

**if** ( rs != **null** ) {

**try** {

rs.close();

} **catch** (SQLException e) {}

}

**if** ( pstmt != **null** ) {

**try** {

pstmt.close();

} **catch** (SQLException e) {}

}

**if** ( conn != **null** ) {

**try** {

conn.close();

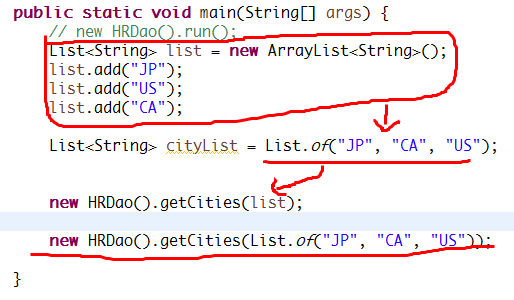
} **catch** (SQLException e) {}

}

}

}

\*\*\*\*\*리스트 작성 줄이기



**public** **static** **void** main(String[] args) {

// new HRDao().run();

List<String> list = **new** ArrayList<String>();

list.add("JP");

list.add("US");

list.add("CA");

List<String> cityList = List.*of*("JP", "CA", "US");

**new** HRDao().getCities(list);

**new** HRDao().getCities(List.*of*("JP", "CA", "US"));

}

-------------------------------------------------------------------------------

**public** **static** **void** main(String[] args) {

// new HRDao().run();

**new** HRDao().getCities(List.*of*("JP", "CA", "US"));

}