

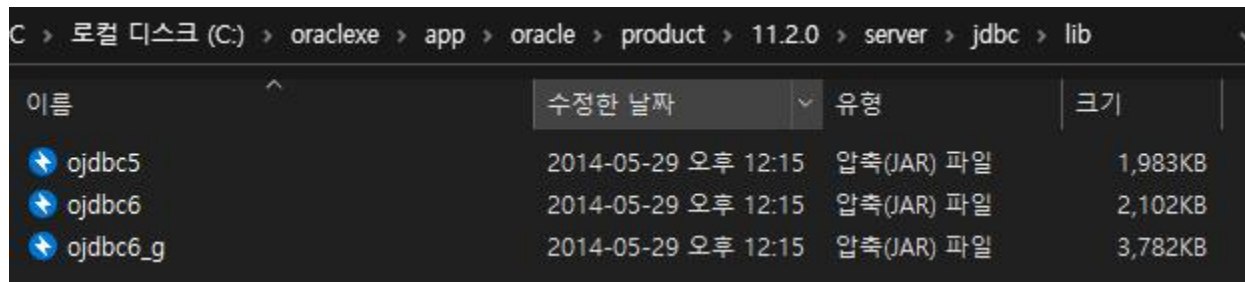
2021-06-02 수요일
ORACLE

2021.06.02, ORACLE, 1.Java와 DB연동하기(1)

1. CONNECTOR 준비하기 : 자바와 데이터베이스는 드라이버로서 포트번호 등을 입력해주면서 연동시켜주는 기능이 있다.

그 기능을 가능하게 해주는 것이 LIBRARY형태로 저장되어 되어 있는 CONNECTOR이다.

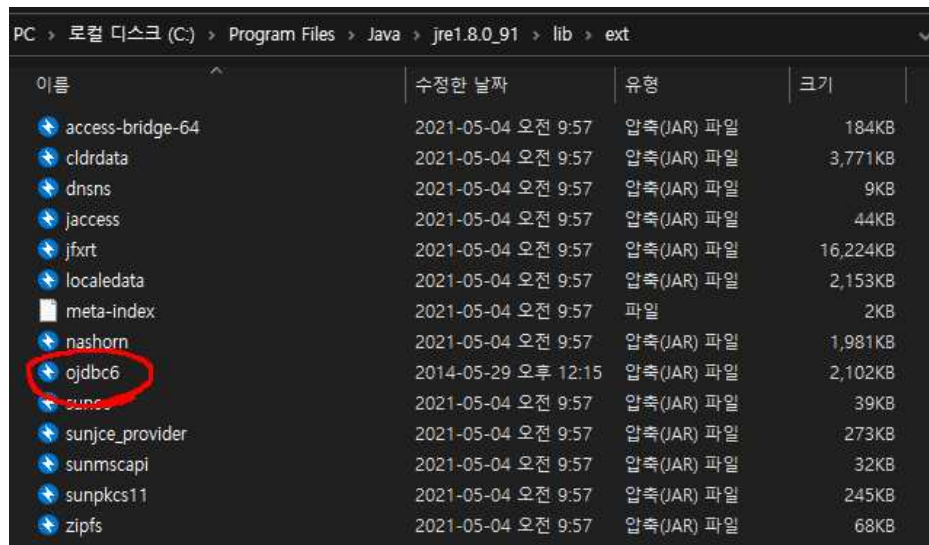
* 이 CONNECTOR는 오라클에서 ojdbc6.jar의 형태로 저장되어 있으며 설치할 때 다른 사항을 선택하지 않았을 경우 아래 경로에 설치되었을 것이다. (드라이브 -> oracle.exe -> app -> oracle -> product -> 11.2.0 -> server -> jdbc -> lib -> ojdbc6



이름	수정된 날짜	유형	크기
ojdbc5	2014-05-29 오후 12:15	압축(JAR) 파일	1,983KB
ojdbc6	2014-05-29 오후 12:15	압축(JAR) 파일	2,102KB
ojdbc6_g	2014-05-29 오후 12:15	압축(JAR) 파일	3,782KB

2. 이 ojdbc6을 복사한 후 아래 경로에 붙여넣는다.

(드라이브 -> Program Files -> Java -> jre1.8.0_91 -> lib -> ext



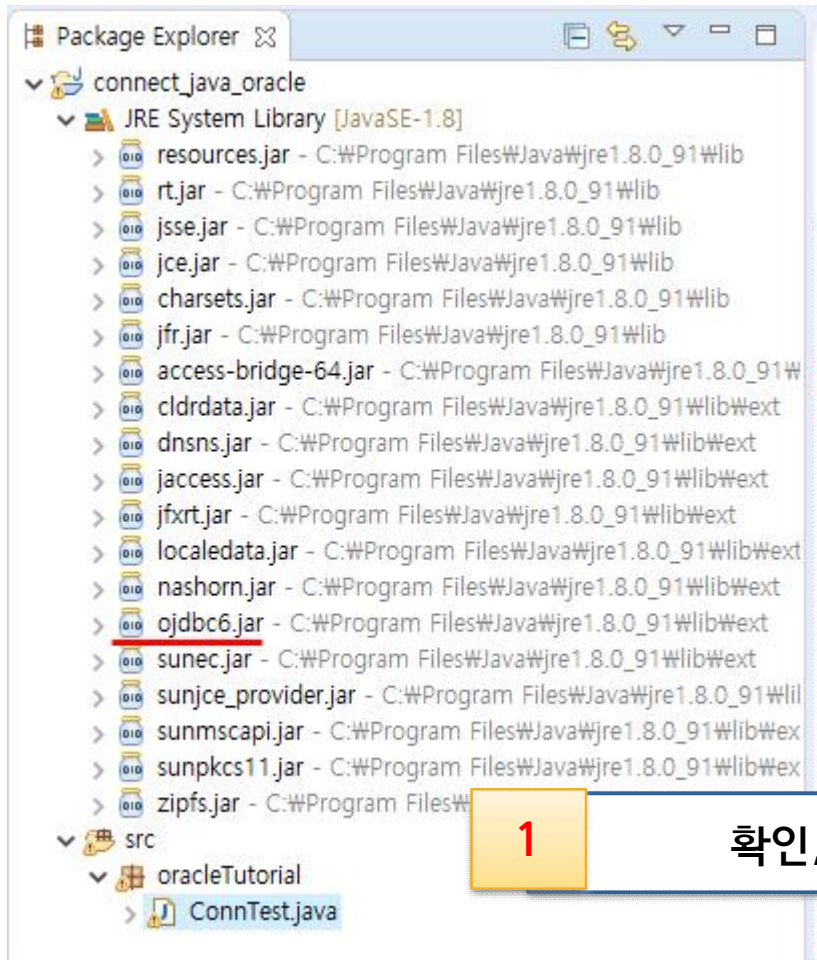
이름	수정된 날짜	유형	크기
access-bridge-64	2021-05-04 오전 9:57	압축(JAR) 파일	184KB
clrddata	2021-05-04 오전 9:57	압축(JAR) 파일	3,771KB
dnsns	2021-05-04 오전 9:57	압축(JAR) 파일	9KB
jaccess	2021-05-04 오전 9:57	압축(JAR) 파일	44KB
jfxrt	2021-05-04 오전 9:57	압축(JAR) 파일	16,224KB
localedata	2021-05-04 오전 9:57	압축(JAR) 파일	2,153KB
meta-index	2021-05-04 오전 9:57	파일	2KB
nashorn	2021-05-04 오전 9:57	압축(JAR) 파일	1,981KB
ojdbc6	2014-05-29 오후 12:15	압축(JAR) 파일	2,102KB
sunec	2021-05-04 오전 9:57	압축(JAR) 파일	39KB
sunjce_provider	2021-05-04 오전 9:57	압축(JAR) 파일	273KB
sunmscapi	2021-05-04 오전 9:57	압축(JAR) 파일	32KB
sunpkcs11	2021-05-04 오전 9:57	압축(JAR) 파일	245KB
zipfs	2021-05-04 오전 9:57	압축(JAR) 파일	68KB

2021.06.02, ORACLE, 1.Java와 DB연동하기(2)

3. Spring Tool Suite 4를 실행한 후 자바 프로젝트를 만든다.

아래 JRE SYSTEM Library를 열어보면 커넥터가 등록된 것을 확인 할 수 있다.

자동으로 되는 듯 하였지만, 저는 추가로 Build Path를 등록해주어야 작동했습니다. (Build Path등록하는 것 다음페이지 확인)



1

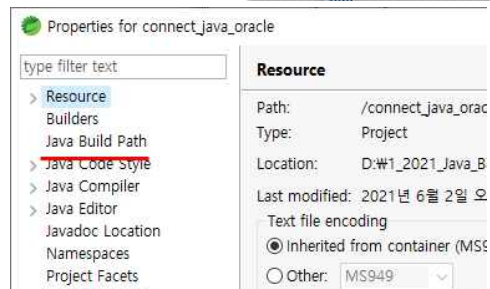
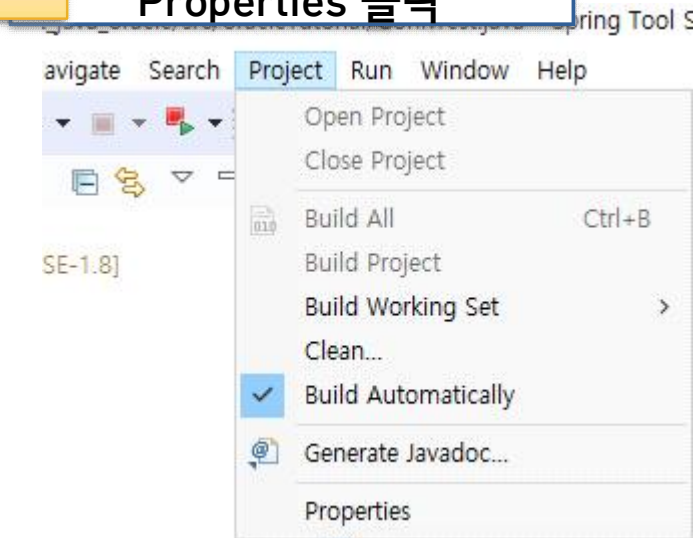
확인, But 저는 Build Path 추가로 등록

2021.06.02, ORACLE, 1.Java와 DB연동하기(3)

4. 프로젝트를 클릭하고 Build Path에서 Configure Build Path를 클릭

1

프로젝트 클릭
Properties 클릭

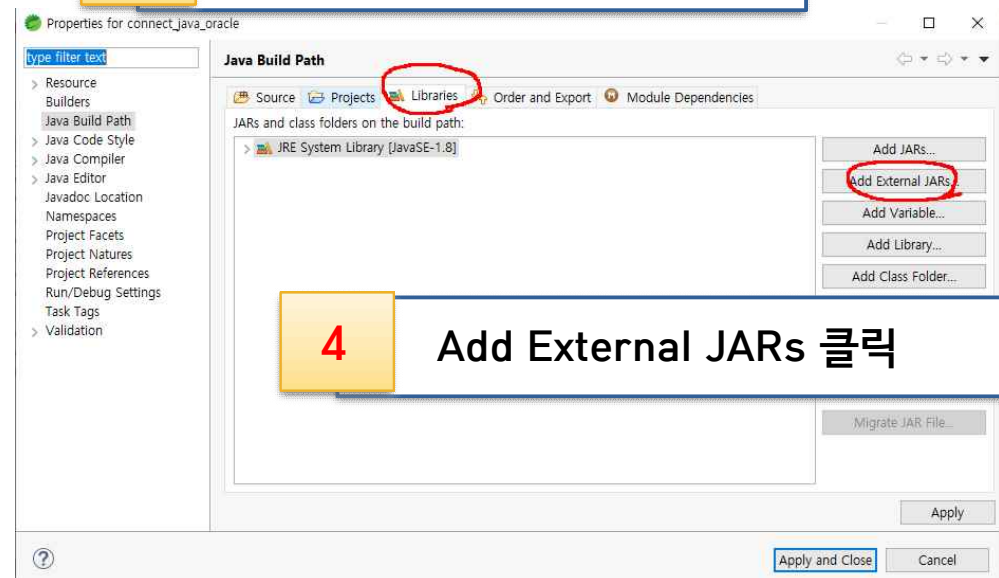


2

Java Build Path 클릭

3

Libraries클릭



4

Add External JARs 클릭

2021.06.02, ORACLE, 1.Java와 DB연동하기(4)

5. 경로에서 ojdbc6.jar를 찾아 등록 한 후 패키지와 클래스를 생성.

The image shows a Windows file explorer window and an IDE (Eclipse) window. The file explorer window displays the path `PC > 로컬 디스크 (C:) > Program Files > Java > jre1.8.0_91 > lib > ext`. It lists several JAR files, with `ojdbc6` highlighted. A yellow box with the number **1** and the text **ojdbc6을 찾아 등록** (Find and register ojdbc6) is overlaid on the file explorer. The IDE window shows the Package Explorer on the right, with the following structure:

- connect_java_oracle
 - JRE System Library [JavaSE-1.8]
 - src
 - oracleTutorial
 - ConnTest.java

A yellow box with the number **2** and the text **코드를 작성할 Class 생성** (Create class to write code) is overlaid on the IDE window.

2021.06.02, ORACLE, 1.Java와 DB연동하기(5)

6. Class를 열고 다음과 같은 코드 작성 => 코드 필요하시면 제가 카톡으로 드릴게요.

```
*ConnTest.java
1 package oracleTutorial;
2 import java.sql.*;
3 public class ConnTest {
4     public static Connection getConnection () {
5         String user = "HR";
6         String pw = "1234";
7         String url = "jdbc:oracle:thin:@localhost:1521:xe";
8         Connection conn = null;
9
10        try {
11            Class.forName("oracle.jdbc.driver.OracleDriver");
12        } catch (ClassNotFoundException e) {
13            e.printStackTrace();
14        }
15        try {
16            conn = DriverManager.getConnection(url, user, pw);
17            System.out.println("성공");
18        } catch (SQLException e) {
19            e.printStackTrace();
20        }
21        return conn;
22    }
23
24    static public void selectList (Connection conn, PreparedStatement pstmt, ResultSet res) throws SQLException {
25        String sql = "SELECT * FROM employee";
26        pstmt = conn.prepareStatement(sql);
27        res = pstmt.executeQuery();
28        while (res.next()) {
29            int eno = res.getInt("ENO");
30            String ename = res.getString("ENAME");
31            String job = res.getString("JOB");
32            String manager = res.getString("MANAGER");
33            String hiredate = res.getString("HIREDATE");
34            Double salary = res.getDouble("SALARY");
35            Double commission = res.getDouble("COMMISSION");
36            Double dno = res.getDouble("DNO");
37            System.out.println("ENO : "+eno);
38            System.out.println("ENAME : "+ename);
39            System.out.println("JOB : "+job);
40            System.out.println("MANAGER : "+manager);
41            System.out.println("HIREDATE : "+hiredate);
42            System.out.println("SALARY : "+salary);
43            System.out.println("COMMISSION : "+commission);
44            System.out.println("DNO : "+dno);
45            System.out.println("");
46        }
47    }
48    static public void main (String[] arg) throws ClassNotFoundException, SQLException {
49
50        Connection conn = getConnection();
51        PreparedStatement pstmt = null;
52        ResultSet res = null;
53        String sql;
54
55        System.out.println("employee 가져오기");
56        System.out.println("=====");
57
58        sql = "SELECT * FROM employee;";
59        selectList(conn, pstmt, res);
60    }
61 }
```

1

HR/1234 입력 : 계정 정보

2

출력할 정보형태 만들기

3

쿼리문 실행

(다음페이지에 별첨으로 INSERT,DELETE 실행하는 예시문도 드릴게요.)

2021.06.02, ORACLE, 1.Java와 DB연동하기(6)

7. 별첨 : INSERT, DELETE

INSERT

```
//      sql = "INSERT INTO employee VALUES(~)";
//      pstmt = conn.prepareStatement(sql);
//      pstmt.setInt(1, eno);
//      pstmt.setString(2, "ename");
//      pstmt.setString(3, "job");
//      pstmt.setString(4, "manager");
//      ... 계속해서 테이블의 형태에 맞추어 등록해 주시면 됩니다.
//      pstmt.executeUpdate();
```

DELETE

```
//      sql = "DELETE FROM sys.board where 조건";
//      pstmt = conn.prepareStatement(sql);
//      pstmt.setInt(1, 1);
//      pstmt.executeUpdate();
```

출처 :

<http://junil-hwang.com/blog/oracle-java-%EC%97%B0%EB%8F%99/>

2021.06.02, ORACLE, 2.Oracle : PL/SQL

<설명> Oracle에서 변수를 선언하고 FOR문과 WHILE문과 같은 반복작업을 할 때 PL/SQL 프로시저를 사용한다.

그래서 System.out.println과 같은 출력을 DBMS출력으로 사용하는데 보기 -> DBMS출력을 통해 활성화 시키거나 SET SERVEROUTPUT ON 의 명령어로 출력창을 활성화 시켜야 볼 수 있다. (결과 값이 안나올 경우 카톡주세요~)

<형태> DECLARE 변수 선언 BEGIN 코드 작성 END; /

<예시> 구구단 (2단) 만들기

```
DECLARE
    dan number := 2;
    i number := 1;
BEGIN
    WHILE i <= 9
    LOOP
        dbms_output.put_line(dan || '*' || i || '=' || (dan*i));
        i := i+1;
    END LOOP;
END;
/
```

```
2*1=2
2*2=4
2*3=6
2*4=8
2*5=10
2*6=12
2*7=14
2*8=16
2*9=18
```

PL/SQL 프로시저가 성공적으로 완료되었습니다.

<문제> 1. 이중 FOR문이나 WHILE문을 사용하여 구구단을 1단부터 9단까지 출력하여라.

2. 이중 FOR문이나 WHILE문 그리고 IF문을 사용하여 2부터 30까지 중 소수를 출력하여라. (HINT : 나머지를 구하는 함수 => MOD(나누어질 수, 나눌 수))

```
1*1=1 2*1=2 3*1=3 4*1=4 5*1=5 6*1=6 7*1=7 8*1=8 9*1=9
1*2=2 2*2=4 3*2=6 4*2=8 5*2=10 6*2=12 7*2=14 8*2=16 9*2=18
1*3=3 2*3=6 3*3=9 4*3=12 5*3=15 6*3=18 7*3=21 8*3=24 9*3=27
1*4=4 2*4=8 3*4=12 4*4=16 5*4=20 6*4=24 7*4=28 8*4=32 9*4=36
1*5=5 2*5=10 3*5=15 4*5=20 5*5=25 6*5=30 7*5=35 8*5=40 9*5=45
1*6=6 2*6=12 3*6=18 4*6=24 5*6=30 6*6=36 7*6=42 8*6=48 9*6=54
1*7=7 2*7=14 3*7=21 4*7=28 5*7=35 6*7=42 7*7=49 8*7=56 9*7=63
1*8=8 2*8=16 3*8=24 4*8=32 5*8=40 6*8=48 7*8=56 8*8=64 9*8=72
1*9=9 2*9=18 3*9=27 4*9=36 5*9=45 6*9=54 7*9=63 8*9=72 9*9=81
```

PL/SQL 프로시저가 성공적으로 완료되었습니다.

```
2 3 5 7 11 13 17 19 23 29
```

PL/SQL 프로시저가 성공적으로 완료되었습니다.