[**JDBC**\_Java DataBase Connectivity ]

Jdbc는 자바언어에서 데이터 베이스에 접근할수 있는 프로그래밍 Api이다

자바언어에서 jdbc를 이용해서 sql문을 데이터베이스에 요청할수 있다

[JDBC실행전 필수]

**데이터베이스 정보확인**

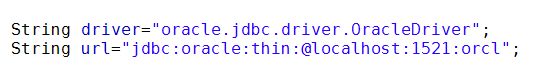
**오라클데이터 베이스 2가지 실행여부 확인**

제어판-관리도구-서비스에서 오라클 실행여부 확인

**jre라이브러리에서 우클릭후 buildpath-edd Extend Jars에서 ojdbc.jar파일이 있나 확인**

[JDBC 실행순서]

**오라클 데이터 베이스 연동위해 4가지 정보저장**



K-20150719-231931-9.jpg

**드라이버 로딩**

K-20150719-232030-4.jpg

**Connection맺기**

자바코드와 오라클 데이터 베이스를 연결하는 것을 의미한다

연결은 java.sql 패키지의 Connection 을 사용

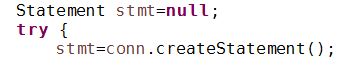
-K-20150719-232156-7.jpg

**Statement 생성**

요청할 SQL문을 전송할 때 사용하는 API이다-connection을 사용해서 생성한다

새로운 레코드를 여러 번 저장하는 경우에 동일한 Insert문을 여러 번 작성해야하는 단점이 있다.문자,날짜데이터의 경우 반드시 **‘ ‘**를 지정한다

PreparedStatement가 SQL문을 한번작성후 데이터만 추가로 설정하는 작업을 할수있다



**SQL문 작성**

자바에서 데이터베이스에 요청할 때 sql문은 문자열로 저장한다.

K-20150719-232558-8.jpg

**SQL문 전송 및 결과값 얻기**

SQL문 전송은 statement 객체를 이용해서 데이터베이스에 전송한다

요청하는 sql문에 따라서 사용하는 메소드가 달라진다

**INSERT,DELETE,UPDATE**

- **ExcuteUpdate** 메소드 사용,메소드 인자는 SQL

-메소드를 요청시 항상 몇 개수정,몇 개삽입등으로 나온걸 알수있다

-따라서 ExcuteUpdate 메소드의 리턴값은 정수값은 변경된 레코드의 개수이다

**SELECT요청**

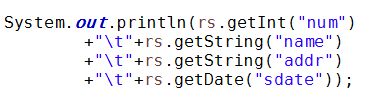
-**ExquteQuery** 메소드 사용

-메소드인자에는 SQL

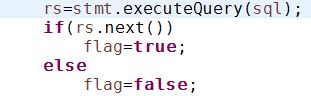
-SELECT 문을 지정하면 항상 테이블 형태로 결과가 출력되었다.이객체를 **ResultSet**이라한다

-ResultSet객체는 포인터를 이용해서 레코드를 선택하고 나중에 레코드의 컬럼을 지정해서 데이터를 얻는다 레코드를 선택하는 메소드는 **next( )**메소드를 사용하고 컬럼은 데이터형에 따라

**getInt(컬럼명),getString(컬럼명)**메소드를 사용할수 있다



-결과값이 여러 개 있을수 있으므로 **while( )문 사용하며** next( )메소드는 레코드가 있는 경우에는 true,값이 없으면 false값을 리턴한다



**Close( )-자원반납**

파일 및 데이터 베이스는 자바에서 사용하는 외부자원이기 때문에 반드시 사용후 자원을 해제시켜야 한다

사용한 자원은 **ResultSet,Statement,Connection** 이다

사용한 역순으로 close( )시킨다-일반적으로 finally구문에서 코드구현

