**대량의 Text data를 저장할수 있는 CLOB 데이터**

CLOB이란 오라클에 존재하는 필드 타입으로 대량의 텍스트 데이터 (4GB)를 저장할 수 있다. oracle 9i까

지는 CLOB 데이터는 다른 필드 타입과는 달리 다루기가 조금 까다로웠으나 10g부터 자바 프로그램 에서

는 일반 문자열 타입과 동일하게 다룰 수 있다.

**clobtable에 데이터를 삽입하고 출력하는 예제 작성하기**

|  |
| --- |
| **CREATE** **TABLE** clobtable (num number,content clob);  **DELETE** CLOBTABLE;  **SELECT** \* **FROM** CLOBTABLE; |

|  |
| --- |
| **import** java.sql.Connection;  **import** java.sql.DriverManager;  **import** java.sql.PreparedStatement;  **import** java.sql.ResultSet;  **import** java.sql.SQLException;  **public** **class** ClobTest {  Connection con;  StringBuffer sb=**null**;  **static**{    }  **public** **void** connect(){  }  **public** **void** insert(){    PreparedStatement pstmt=**null**;  //ResultSet rs=null;  **try**{  connect();  sb=**new** StringBuffer();  **for**(**int** i=0;i<=10000;i++){  sb.append("홍길동");  }  pstmt=con.prepareStatement(sql);  pstmt.setString(1, sb.toString());  **int** count = pstmt.executeUpdate();  **if**( ){  System.***out***.println("insert success!");  }  **else**{  System.***out***.println("insert fail");  }  }**catch**(Exception e){  e.printStackTrace();  }    }  **public** **void** select(){  String sql = "SELECT \* FROM clobtable";  PreparedStatement pstmt = **null**;  ResultSet rs = **null**;  **try**{    }  **catch**(SQLException se){  se.printStackTrace();  }  **finally**{  **try**{  pstmt.close();  con.close();  }  **catch**(Exception e){  e.printStackTrace();  }  }  }  **public** **static** **void** main(String args[]){  ClobTest ct = **new** ClobTest();  ct.insert();  ct.select();  }  } |

