**[오라클 BLOB에 이미지 저장, 출력 (JSP)](http://expo.daydic.com/entry/오라클-BLOB에-이미지-저장-출력-JSP)**

[언어/JSP](http://expo.daydic.com/category/언어/JSP)

출처 : <http://blog.naver.com/iamjun7/>

**1. JSP 를 이용한 BLOB 저장**

-------------------------------------------------------------------------  
File file = (File) param.get("sajin"); // 등록할 File  
Blob emptyBlob = null;   
OutputStream outstream = null;  
FileInputStream finstream = null;  
ResultSet rs = null;

try{  
// EMPTY\_BLOB() 처리  
sql = "update table set sajin=EMPTY\_BLOB() where id=?";  
ps = con.prepareStatement(sql);  
ps.setString(1, (String)param.get("id"));  
if ( ps.executeUpdate() < 0 ) throw new Exception();

// 저장할 sajin Column 가져온다.  
sql = "select sajin from table where id=?";  
ps = con.prepareStatement(sql);  
ps.setString(1, (String)param.get("id"));  
rs = ps.executeQuery();  
if ( rs.next() ) emptyBlob = rs.getBlob(1);

// db blob output stream  
oracle.sql.BLOB bol = (oracle.sql.BLOB) emptyBlob;  
outstream = bol.getBinaryOutputStream();  
int size = bol.getBufferSize();  
// 파일 input stream  
finstream = new FileInputStream(file);

// 파일 읽어서 db에 넣기  
byte[] buffer = new byte[size];  
int length = -1;  
while ((length = finstream.read(buffer)) != -1) {  
outstream.write(buffer, 0, length);  
}  
} catch (Exception e){  
throw(e);  
} finally {  
if( rs != null ) rs.close();  
if( finstream != null ) finstream.close();   
if( outstream != null ) outstream.close();   
}

-------------------------------------------------------------------------  
즉, 다른 타입처럼 update문이나 insert를 이용하지 않는다.  
다시 한번 정리하면, insert할 column을 EMPTY\_BLOB()로 초기화  
초기화된 column을 select 하여 OutputStream을 통해 file을 DB에 저장한다.

**2. JSP 를 이용한 BLOB 브라우저에서 보기**

-------------------------------------------------------------------------  
package showImage;

import java.io.\*;  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
import java.sql.\*;

public class ShowImageServlet extends HttpServlet  
{

public void doGet(HttpServletRequest request, HttpServletResponse response)  
throws ServletException, IOException  
{  
doPost(request, response);  
}

public void doPost(HttpServletRequest request, HttpServletResponse response)  
throws ServletException, IOException  
{

Connection con = null;  
ResultSet rs = null;  
PreparedStatement ps = null;  
InputStream is = null;

// Image를 가져오기위한 키값들  
String key1 = request.getParameter("key1"); // Primary key 1  
String key2 = request.getParameter("key2"); // Primary key 2  
  
String file\_type = null; // Image 파일 타입  
String content\_type = null; // Image 보여주기위한 Content\_type

// Image 가져올 SQL  
final String SQL =  
" SELECT image\_type, image " +  
" FROM cu\_basic\_t " +  
" WHERE key1 = '"+key1+"' AND key2 = '"+key2+"' ";

try {  
con = UtilDB.getConnection(); // DB 연결  
ps = con.prepareStatement(SQL);  
rs = ps.executeQuery(); // SQL 실행

// Image 가져오는 부분(content type도 정해준다.)  
if (rs!=null && rs.next()){

file\_type = rs.getString("image\_type");  
is = rs.getBinaryStream("image");

if (file\_type.toUpperCase().equals("JPG")) file\_type = "jpeg";  
else if (file\_type.toUpperCase().equals("GIF")) file\_type = "gif";

content\_type = "image/" + file\_type; // "image/jpeg"나 "image/gif"  
response.setContentType(content\_type); // Content Type Set

// Image를 Stream을 통해 out  
ServletOutputStream os = response.getOutputStream();  
int binaryRead;  
while ((binaryRead = is.read()) != -1)  
{  
os.write(binaryRead);  
}

} else {  
throw new Exception("사진이 없습니다.");  
}  
}  
catch(ServletException e) {  
e.printStackTrace();  
throw e;  
}catch(IOException e) {  
e.printStackTrace();  
throw e;  
}  
catch(Exception e) {  
System.out.println("An error occurs : " + e.toString());  
e.printStackTrace();  
}  
finally {  
UtilDB.closeConnection(con, ps, rs); // DB 닫아준다.  
}  
}  
}  
package showImage;

import java.io.\*;  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
import java.sql.\*;

public class ShowImageServlet extends HttpServlet  
{

public void doGet(HttpServletRequest request, HttpServletResponse response)  
throws ServletException, IOException  
{  
doPost(request, response);  
}

public void doPost(HttpServletRequest request, HttpServletResponse response)  
throws ServletException, IOException  
{

Connection con = null;  
ResultSet rs = null;  
PreparedStatement ps = null;  
InputStream is = null;

// Image를 가져오기위한 키값들  
String key1 = request.getParameter("key1"); // Primary key 1  
String key2 = request.getParameter("key2"); // Primary key 2  
  
String file\_type = null; // Image 파일 타입  
String content\_type = null; // Image 보여주기위한 Content\_type

// Image 가져올 SQL  
final String SQL =  
" SELECT image\_type, image " +  
" FROM cu\_basic\_t " +  
" WHERE key1 = '"+key1+"' AND key2 = '"+key2+"' ";

try {  
con = UtilDB.getConnection(); // DB 연결  
ps = con.prepareStatement(SQL);  
rs = ps.executeQuery(); // SQL 실행

// Image 가져오는 부분(content type도 정해준다.)  
if (rs!=null && rs.next()){

file\_type = rs.getString("image\_type");  
is = rs.getBinaryStream("image");

if (file\_type.toUpperCase().equals("JPG")) file\_type = "jpeg";  
else if (file\_type.toUpperCase().equals("GIF")) file\_type = "gif";

content\_type = "image/" + file\_type; // "image/jpeg"나 "image/gif"  
response.setContentType(content\_type); // Content Type Set

// Image를 Stream을 통해 out  
ServletOutputStream os = response.getOutputStream();  
int binaryRead;  
while ((binaryRead = is.read()) != -1)  
{  
os.write(binaryRead);  
}

} else {  
throw new Exception("사진이 없습니다.");  
}  
}  
catch(ServletException e) {  
e.printStackTrace();  
throw e;  
}catch(IOException e) {  
e.printStackTrace();  
throw e;  
}  
catch(Exception e) {  
System.out.println("An error occurs : " + e.toString());  
e.printStackTrace();  
}  
finally {  
UtilDB.closeConnection(con, ps, rs); // DB 닫아준다.  
}  
}  
}  
-------------------------------------------------------------------------  
JSP 기준이다.

보여줄 JSP에 다음을 추가

<img src="/servlet/ShowImageServlet?key1=<%=key1%>&key2=<%=key12%>"

width="100" height="100" />

SRC에 image를 보여주도록 만들어진 서블릿을 입력

참고 (서블릿 실행을 위해 WAS(Jeus) 설정법)

WEB-INF\web.xml 내에 서블릿 등록을 등록하고 서버 재 Start시킨다.

<servlet>  
<servlet-name>ShowImageServlet</servlet-name>  
<servlet-class>showImage.ShowImageServlet</servlet-class>  
</servlet>  
<servlet-mapping>  
<servlet-name>ShowImageServlet</servlet-name>  
<url-pattern>/ShowImageServlet</url-pattern>  
</servlet-mapping>