

Java – rozszerzenie

dr. A. Dawid

PLAN WYKŁADU

1. Java2D
 - AWT
 - SWING
 - Inicjalizacja
 - Metody
2. JDBC
 - Connector
3. Programowanie sieciowe
 - Socket
 - ServerSocket
4. Servlety
 - Serwer internetowy
 - Klasa HttpServlet
5. JSP
6. Java3D

AWT – *Abstract Window Toolkit*

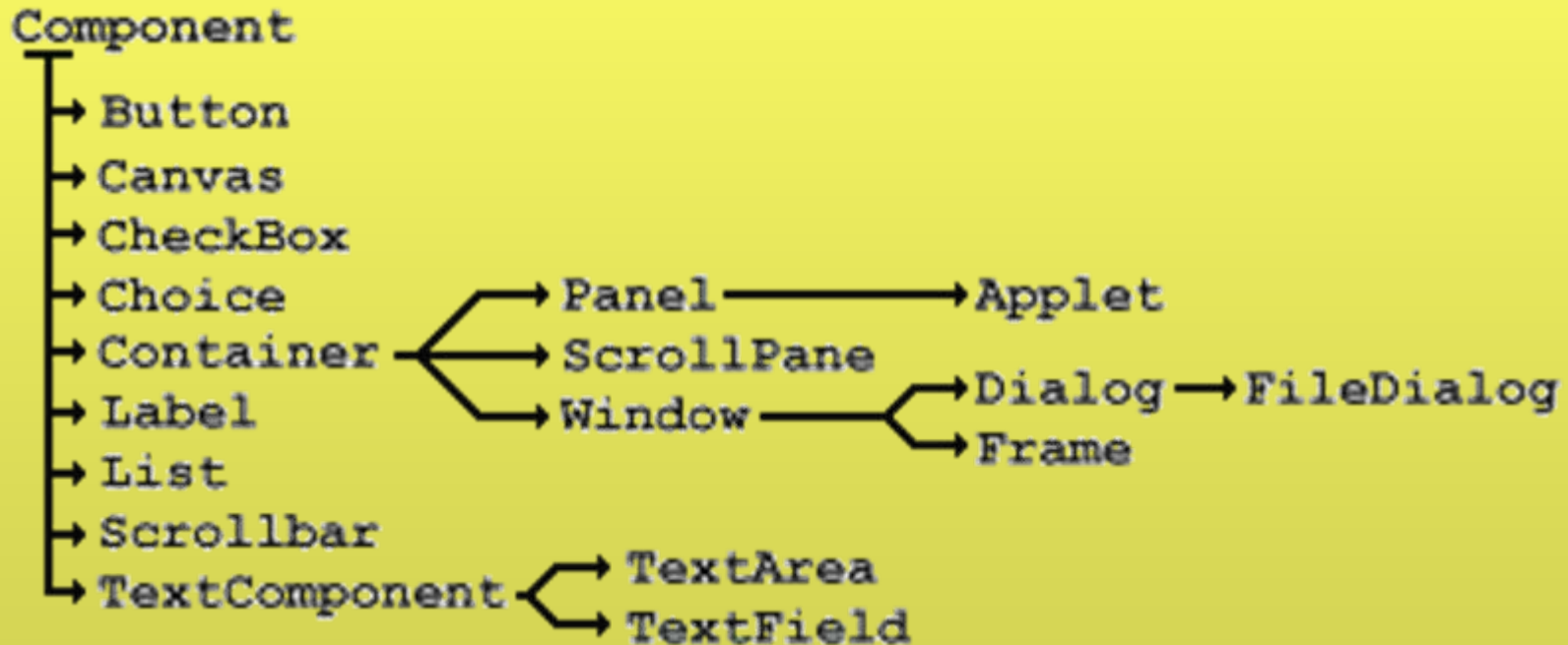
Biblioteka komponentów graficznych

- Okna
- Komponenty
- Grafika rastrowa
- Interakcja (obsługa zdarzeń)

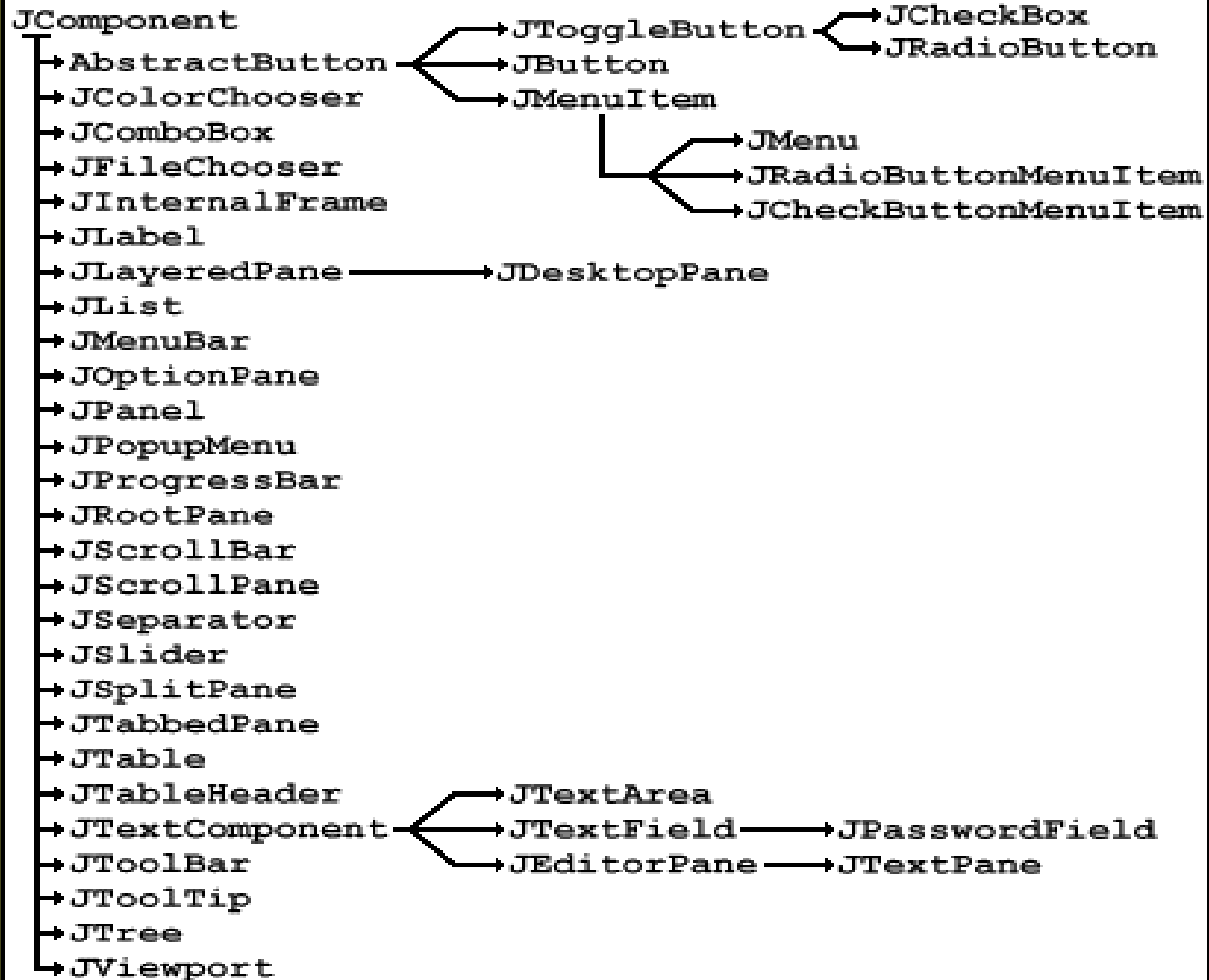
Java – SWING

Platforma JAVA 2

JComponent - rozszerzenie klasy Container biblioteki AWT



Java – SWING



Architektura MVC

- **Model** – generowanie sygnałów (event)
- **View** – renderowanie komponentów „look”
- **Controller** – reakcja komponentu na urządzenie wejściowe „feel”

Niezależność od platformy

Look-and-feel taki sam na Linux, Windows, Macintosh

Rozszerzona grafika 2-wymiarowa

```
Public void Paint (Graphics g) {  
Graphics2D g2 = (Graphics2D) g ;  
...  
}
```

Inicjalizacja poprzez rzutowanie obiektu
Graphics na obiekt **Graphics2D**

Właściwości

- Rysowanie zewnętrznych krawędzi
- Wypełnianie kolorem lub wzorem
- Wiele nowych definicji figur
- Rysowanie trzcionek
- Rysowanie obrazów

```
gp=new GradientPaint(0f,0f,blue,0f,30f,green);  
G2.setPaint(gp);
```

Obiekty

- Graphics2D
- Point2D
- Line2D
- Rectangle2D, RoundRectangle2D, Arc2D, Ellipse2D
- QuadCurve2D, CubicCurve2D
- GeneralPath
- Area
- BufferedImage

Przykłady.

BufferedImage

1. Wczytywanie obrazu do pamięci.

```
BufferedImage img = null;
try {
    img = ImageIO.read(new File("zima.jpg"));
} catch (IOException e) { }
```

Aplet

```
try {
    URL url = new URL(getCodeBase(), "zima.jpg");
    img = ImageIO.read(url);
} catch (IOException e) { }
```

BufferedImage

2. Rysowanie obrazu.

```
void Graphics2D.drawImage(BufferedImage img,  
BufferedImageOp op, int x, int y)
```

```
try {  
    img = ImageIO.read(urlx);  
    int w = img.getWidth(null);  
    int h = img.getHeight(null);  
    bi = new BufferedImage(w, h, BufferedImage.TYPE_INT_ARGB);  
    Graphics g = bi.getGraphics();  
    g.drawImage(img, 0, 0, null);  
} catch (IOException e) { }
```

BufferedImage

3. Zapis obrazu.

```
try {  
    File outputfile = new File("saved.jpg");  
    ImageIO.write(bi, "jpg", outputfile);  
} catch (IOException e){}
```

Przykład PLANSZA.

Java DataBase Connectivity

Ver. 2.0

Połączenie baz danych z interfejsem
programistycznym JAVA

MySQL

<http://dev.mysql.com/downloads/connector/j/3.1.html>

Obiekty

DriverManager - klasa obsługująca warstwę pomiędzy JDBC a użytkownikiem.

Wywołanie bezpośrednio klasy sterownika.

```
Class.forName("com.mysql.jdbc.Driver").newInstance();
```

Nawiązania połączenia

```
getConnection("parse xml format")
```

```
Protocol://computer/path?param1=value1&param2=value2
```

Ex.

```
"jdbc:mysql://localhost/krasnal?user=krasnal&password=krasnal"
```

Statement - klasa wysyłająca proste zapytania SQL.

Obiekty

Statement - klasa wysyłająca proste zapytania SQL.

Podklasy.

PreparedStatement, CallableStatement

Metody

```
Statement st = c.createStatement();
```

Umieszcza wyrażenie w kontenerze.

```
ResultSet rs = st.executeQuery("SELECT * FROM adresy");
```

Wykonuje zapytanie SQL.

```
int rows=st.executeUpdate("CREATE DATABASE test");
```

Obiekty

```
stmt.execute(ZapytanieSQLoNiewiadomymWyniku);
while(true) {

int rowCount = stmt.getUpdateCount();
if(rowCount > 0) // To jest UPDATE
{
    System.out.println("Wiersze zmienione = " + count);
    stmt.getMoreResults();
    continue;
}

if(rowCount = 0) // komendy DDL lub update = 0
{
    System.out.println(" nic się nie zmieniło lub komenda DDL");
    stmt.getMoreResults();
    continue;
}
```

Obiekty

```
//Czy to są może wiersze danych
ResultSet rs = stmt.getResultSet();
if(rs != null) {
    . . . // pobrać informacje na temat kolumn
    while(rs.next()) {
        . . . // przetwarzanie danych
        stmt.getMoreResults();
        continue;
    }
    break; // Nie ma więcej zapytań
}
```

Obiekt ResultSet

```
ResultSet r = stmt.executeQuery("SELECT a, b, c FROM Table1");
while (r.next())
{
// wydruk wartości dla konkretnego wiersza
int i = r.getInt("a");
String s = r.getString("b");
float f = r.getFloat("c");
System.out.println(" i = " + i + " " + s + " " + f);
}
```

Duże ilości danych

```
Statement stmt = con.createStatement();
ResultSet r = stmt.executeQuery("SELECT x FROM Table2");

// Pobieraj wartości z kolumny jeden w porcjach po 4 kB:
byte buff = new byte[4096];

while (r.next())
{
    InputStream fin = r.getAsciiStream(1);
    for (;;) {
        int size = fin.read(buff);
        if (size == -1) { // koniec strumienia
            break;
        }
    }
    // Wypisanie bufora:
    output.write(buff, 0, size);
}
```

PreparedStatement

```
PreparedStatement pstmt = con.prepareStatement(  
"UPDATE table4 SET imie = ? WHERE jezyk = ?");  
  
pstmt.setString(1, "John");  
pstmt.setString(2, "Ang");  
  
pstmt.clearParameters() //reset parametrów
```

Przykład

```
import java.sql.*;
import java.io.*;

public class testSQL
{
    public static void main(String[] Args)
    {
        try
        {
            //ladowanie sterownika jdbc
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        }
        catch (Exception e)
        {
            System.err.println("cos nie tak ze sterownikiem:\n"+e);
        }

        try
        {
            //nawiazanie polaczenia z baza danych krasnal
```

JDBC™

```
Connection c =DriverManager.getConnection
("jdbc:mysql://localhost/krasnal?user=krasnal&password=krasnal");
    //operacja na bazie danych
    Statement st = c.createStatement();
    ResultSet rs = st.executeQuery("SELECT * FROM adresy");

    while (rs.next())
    {
        System.out.println(rs.getString(1)+" |
+rs.getString(2)+" | "+rs.getString(3));
    }

    rs.close();
    st.close();
    c.close();
}
catch (SQLException e)
{
    System.out.println(e);
}
}
```

SERVLET

Aplikacja wykonywana w środowisku Servera www

Zestaw klas do obsługi protokołu.
servlet.jar

Serwery www:

- Tomcat – Jakarta project
- Resin

```
import javax.servlet.http.*;  
import javax.servlet.*;
```

SERVLET

Obiekty pakietu

Klasa HttpServlet

klasa abstrakcyjna/fabrykująca klasy potomne

Serializable, serwery wielowątkowe, dostęp

Metody wywoływane przez serwer.

doGet, doPost, doPut, doDelete, doHead,
doOptions, doTrace, getLastModified

SERVLET

```
public void doGet (  
    HttpServletRequest req,  
    HttpServletResponse res  
    ) throws ServletException, IOException
```

```
Interface HttpServletRequest  
Interface HttpServletResponse
```

Ciała metod specyficzne dla danego serwera.

SERVLET

HttpServletRequest

Wybrane metody

```
getCharacterEncoding()  
getContentLength()  
getContentType()  
getParameter(String name)  
getProtocol()  
getReader()  
getRemoteAddr()  
getRemoteHost()  
getServerName()  
getServerPort()
```

SERVLET

HttpServletResponse

Wybrane metody

getBufferSize()

getWriter()

getCharacterEncoding()

setBufferSize(int rozmiar)

setContentLength(int dlugosc)

setContentType(String typ)

SERVLET

doGet

```
public void doGet(HttpServletRequest req, HttpServletResponse
res) throws ServletException, IOException
{
    res.setContentType("text/html");
    PrintWriter pw = res.getWriter();

    pw.println("<head>");
    pw.println("<title>Nasz Servlet</title>");
    pw.println("</head>");
    pw.println("<body bgcolor=antiquewhite>");
    pw.println("<H1>Pierwszy serwlet</H1>");
    pw.println("</body>");
    pw.close();
}
```

SERVLET

doPost

```
public void doPost(HttpServletRequest req, HttpServletResponse
res) throws ServletException, IOException
{
    res.setContentType("text/html");
    PrintWriter pw = res.getWriter();

    pw.println("<head>");
    pw.println("<title>Nasz Servlet</title>");
    pw.println("</head>");
    pw.println("<body bgcolor=antiquewhite>");
    pw.println("<H1>Pierwszy serwlet</H1>");
    String imie=req.getParameter("imie");
    pw.println(imie+"<br>");
    pw.println("</body>");
    pw.close();
}
```

SERVLET – Przykład 1

```
/* Prosty servlet Obsługa zwrotna */
import java.io.*;
import java.util.*;

import javax.servlet.http.*;
import javax.servlet.*;

public class ServletNasz extends HttpServlet {

    public void doGet (
        HttpServletRequest req,
        HttpServletResponse res
    ) throws ServletException, IOException
    {
        res.setContentType("text/html");
        PrintWriter pw = res.getWriter();

        pw.println("<head>");
        pw.println("<meta http-equiv=\"Content-Type\"
content=\"text/html; charset=Windows-1250\">");
        pw.println("<title>Nasz Servlet</title>");
        pw.println("</head>");
    }
}
```

SERVLET – Przykład 1

```
pw.println("<body bgcolor=antiquewhite>");
    pw.println("<h1>Dodaj dane</h1>");
    pw.println("<form method=POST action=ServletNasz>");
    pw.println("<input type=text name=imie>-imie<br>");
    pw.println("<input type=text name=nazwisko>-
nazwisko<br>");
    pw.println("<input type=text name=telefon>-telefon<br>");
    pw.println("<input type=submit value=wyslij>");
    pw.println("</form>");
    pw.println("</body>");
    pw.close();
}
```

```
public void doPost (
    HttpServletRequest req,
    HttpServletResponse res
) throws ServletException, IOException
{
    res.setContentType("text/html");
    PrintWriter pw = res.getWriter();
```

SERVLET – Przykład 1

```
pw.println("<head>");
pw.println("<meta http-equiv=\"Content-Type\"
content=\"text/html; charset=ISO-8859-1\">");
pw.println("<title>Nasz Servlet</title>");
pw.println("</head>");
pw.println("<body bgcolor=antiquewhite>");
pw.println("<h1>WYNIK</h1>");
String imie=req.getParameter("imie");
String nazwisko=req.getParameter("nazwisko");
String telefon=req.getParameter("telefon");
pw.println(imie+"<br>");
pw.println(nazwisko+"<br>");
pw.println(telefon+"<br>");

pw.println("</body>");
pw.close();
}

public ServletNasz() {}
}
```

SERVLET – Przykład 2 (MySQL)

```
/* Prosty servlet Obsługa zwrotna */
import java.io.*;
import java.util.*;
import java.sql.*;
import javax.servlet.http.*;
import javax.servlet.*;

public class ServletBaza extends HttpServlet {

public void CzytajBaze(PrintWriter pw)
{
try
    {
        //ladowanie sterownika jdbc
Class.forName("com.caucho.jdbc.mysql.Driver").newInstance();
    }
    catch (Exception e)
    {
        pw.println("cos nie tak ze sterownikiem:\n"+e);
    }
}
```

SERVLET – Przykład 2 (MySQL)

```
try
{
    //nawiazanie polaczenia z baza danych
    Connection c =
DriverManager.getConnection("jdbc:mysql_caucho://localhost:33
06/alex");

    //operacja na bazie danych
    Statement st = c.createStatement();
    ResultSet rs = st.executeQuery("SELECT * FROM
adresy");

    pw.println("<table border=1>");
    while (rs.next())
    {
pw.println("<tr><td>" + rs.getString(3) + "</td><td>" + rs.getStrin
g(4) + "</td><td>" + rs.getString(5) + "</td></tr>");
    }
    pw.println("</table>");
}
```

SERVLET – Przykład 2 (MySQL)

```
        rs.close();
        st.close();
        c.close();
    }
    catch (SQLException e)
    {
        pw.println(e);
    }
}

public void ZapiszDoBazy(PrintWriter pw, String imie, String
nazwisko, String telefon)
{
    try
    {
        //ladowanie sterownika jdbc

        Class.forName("com.caucho.jdbc.mysql.Driver").newInstance();
    }
    catch (Exception e)
```

SERVLET – Przykład 2 (MySQL)

```
{
    pw.println("cos nie tak ze sterownikiem:\n"+e);
}
try
{
    //nawiazanie polaczenia z baza danych
    Connection c =
DriverManager.getConnection("jdbc:mysql_caucho://localhost:33
06/alex");

    //operacja na bazie danych
    Statement st = c.createStatement();
    int rows = st.executeUpdate("INSERT INTO adresy
(imie,nazwisko,telefon) VALUES
('"+imie+"', '"+nazwisko+"', '"+telefon+"')");

    pw.println("Wynik dodania="+rows+"<br>");

    st.close();
    c.close();
}
```

SERVLET – Przykład 2 (MySQL)

```
catch (SQLException e)
{
    pw.println(e);
}

}

public void doGet (
    HttpServletRequest req,
    HttpServletResponse res
) throws ServletException, IOException
{
    res.setContentType("text/html");
    PrintWriter pw = res.getWriter();

    pw.println("<head>");
    pw.println("<meta http-equiv=\"Content-Type\"
content=\"text/html; charset=Windows-1250\">");
    pw.println("<title>Nasz Servlet</title>");
    pw.println("</head>");
    pw.println("<body bgcolor=antiquewhite>");
```

SERVLET – Przykład 2 (MySQL)

```
pw.println("<h1>Dodaj dane</h1>");
pw.println("<form method=POST action=ServletBaza>");
pw.println("<input type=text name=imie>-imie<br>");
pw.println("<input type=text name=nazwisko>-
nazwisko<br>");
pw.println("<input type=text name=telefon>-telefon<br>");
pw.println("<input type=submit value=wyslij>");
pw.println("</form>");
pw.println();
CzytajBaze(pw);
pw.println("</body>");
pw.close();
}
```

```
public void doPost (
    HttpServletRequest req,
    HttpServletResponse res
) throws ServletException, IOException
{
    res.setContentType("text/html");
    PrintWriter pw = res.getWriter();
```

SERVLET – Przykład 2 (MySQL)

```
    pw.println("<head>");
    pw.println("<meta http-equiv=\"Content-Type\"
content=\"text/html; charset=ISO-8859-1\">");
    pw.println();
    pw.println("<title>Nasz Servlet</title>");
    pw.println("</head>");
    pw.println("<body bgcolor=antiquewhite>");
    pw.println("<h1>WYNIK</h1>");
    String imie=req.getParameter("imie");
    String nazwisko=req.getParameter("nazwisko");
    String telefon=req.getParameter("telefon");
    pw.println(imie+"<br>");
    pw.println(nazwisko+"<br>");
    pw.println(telefon+"<br>");
    ZapiszDoBazy(pw, imie, nazwisko, telefon);
    pw.println("</body>");
    pw.close();
}
public ServletBaza() {}
}
```

JSP (*Java server pages*)

**Integracja języka JAVA i JAVASCRIPT w
Kod wykonywany przez serwer www.**

ZNACZNIKI :

`<%-- Komentarz --%>`

`<%! public void Pisz(){} %>` deklaracje metod

`<% Pisz(); %>` wykonanie kodu

`<%= imie %>` wypisanie wyrażenia

JSP (*Java server pages*)

PRZYKŁAD FORMULARZ:

```
<%@ page import='java.util.*' %>
<%@ page import='java.io.*' %>
<%@ page import='java.sql.*' %>
<%!
String[] text;
String Error;
int rozmiar;
public void CzytajBaze()
{
    int i=0;
    text = new String[32];
    try
    {
        //ladowanie sterownika jdbc
Class.forName("com.caucho.jdbc.mysql.Driver").newInstance();
    }
    catch (Exception e)
    {
        Error="cos nie tak ze sterownikiem: "+e;
    }
}
```

JSP (*Java server pages*)

```
try
{
    //nawiazanie polaczenia z baza danych
    Connection c =
DriverManager.getConnection("jdbc:mysql_caucho://localhost:33
06/alex");

    //operacja na bazie danych
    Statement st = c.createStatement();
    ResultSet rs = st.executeQuery("SELECT * FROM
adresy");

    while (rs.next())
    {
text[i]="<tr><td>" + rs.getString(3) + "</td><td>" + rs.getString(4
) + "</td><td>" + rs.getString(5) + "</td></tr>";
        i++;
    }
    rozmiar=i;
    if(rozmiar>31){ rozmiar=31;}
}
```

JSP (*Java server pages*)

```
rs.close();
    st.close();
    c.close();
}
catch (SQLException e)
{
    Error="Blad:"+e;
}
}
%>

<%
String fImie=(String) request.getParameter("imie");
String fNazwisko=(String) request.getParameter("nazwisko");
String fTelefon=(String) request.getParameter("telefon");
%>
<html>
<head><title>Wypelnienie formularza</title></head>
<body bgcolor=#ffffff>
```

JSP (*Java server pages*)

```
<h1>Wartosci do zapisania:</h1>
<H4>Imie: <%= fImie %><BR>Nazwisko: <%= fNazwisko
%><BR>Telefon: <%= fTelefon %></H4>
<P>
<% out.println("Aktualny czas: " + new java.util.Date()); %>
<% CzytajBaze(); %>
<%
if(Error==null){
%>
<H2 style="color: blue;">Baza adresowa</H2>
<table border=1>
<tr style="color: green;background-color: yellow;font: 28px
bold"><td>Imie</td><td>Nazwisko</td><td>Telefon</td></tr>
<%
for(int i=0;i<rozmiar;i++)
{
    out.println(text[i]);
}
%>
</table>
```

JSP (*Java server pages*)

```
<%  
    }else{%><p style="color: #FF0000;"><%  
    out.println(Error) ;  
    }  
%>  
</p>  
  
</P>  
</body>  
</html>
```