

PERTEMUAN 12 : PEMROGRAMAN DASAR WEB

A. TUJUAN PEMBELAJARAN:

Pada pertemuan ini akan dijelaskan mengenai pembuatan aplikasi/program console menggunakan class, metode, dan objek. Setelah mempelajari materi perkuliahan ini, mahasiswa mampu:

12.1 Pemrograman Dasar Servlet

B. URAIAN MATERI

Tujuan Pembelajaran 12.1:
Pemrograman Dasar Servlet

Beberapa hal yang akan dibahas pada pembelajaran Pemrograman Dasar Servlet:

1. Pengenalan Servlet
2. Contoh Aplikasi JSP-Servlet
3. Contoh Aplikasi Servlet
4. Pengaturan Servlet

Servlet

- Java Servlet adalah objek java yang memproses interaksi HTTP di sisi server
- Java Servlet mendukung terciptanya berbagai konten web dinamis, dapat melakukan hampir semua yang dapat dilakukan dengan HTTP dan bahasa markup, dari katalog online, aplikasi chatting, situs perdagangan, dan lain-lain
- Java Servlet banyak digunakan secara luas sebagai komponen java web, walaupun saat ini biasa digunakan secara tidak langsung

Sebuah aplikasi web berkomunikasi dengan perangkat lunak client melalui HTTP. HTTP sebagai protocol yang berbicara menggunakan request dan response

menjadikan aplikasi web bergantung kepada siklus ini untuk menghasilkan dokumen yang ingin diakses oleh pengguna.

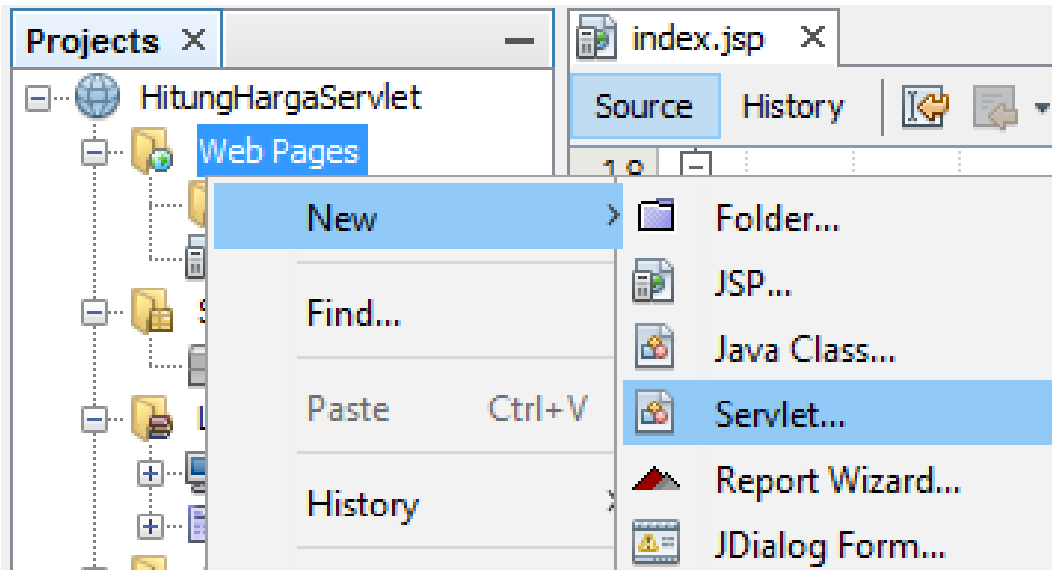
Aplikasi web yang dibuat harus memiliki cara untuk membaca HTTP request (permintaan client) dan mengembalikan HTTP response (jawaban atas permintaan client) ke pengguna.

Contoh Aplikasi JSP-Servlet

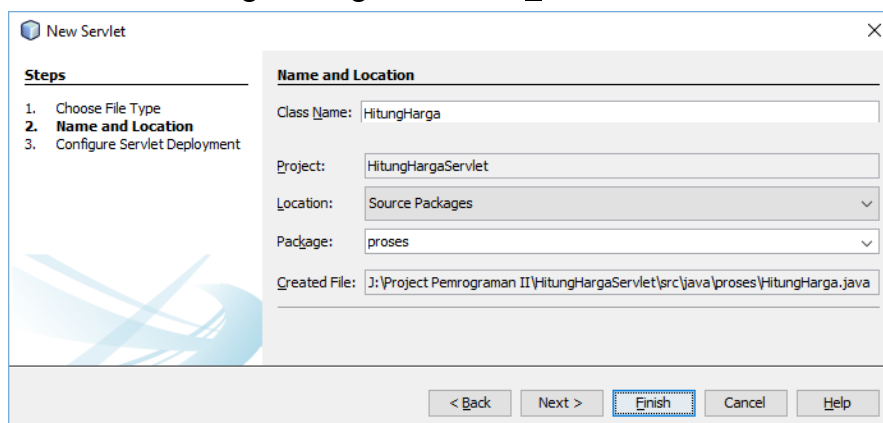
- Buat proyek baru dengan nama HitungHargaServlet
- Tambahkan file JSP dengan nama index
- Ubah source code menjadi seperti di bawah ini

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Menghitung Harga</title>
  </head>
  <body>
    <h2>Form Memasukkan Nilai</h2>
    <form action="HitungHarga" method="post">
      <table>
        <tr>
          <td>Nama Barang</td>
          <td><input type="text" name="namaBarang"></td>
        </tr>
        <tr>
          <td>Harga Satuan</td>
          <td><input type="text" name="hargaSatuan"></td>
        </tr>
        <tr>
          <td>Jumlah</td>
          <td><input type="text" name="jumlah"></td>
        </tr>
        <tr>
          <td colspan="2">Diskon diberikan sebesar 5% jika jumlah >= 100</td>
        </tr>
        <tr>
          <td colspan="2">dan total harga sebelum diskon >= 1000000</td>
        </tr>
        <tr>
          <td colspan="2"><input type="submit" value="Hitung"></td>
        </tr>
      </table>
    </form>
  </body>
</html>
```

- Tambahkan file Servlet, klik kanan pada Web Pages, pilih New ► Servlet



- Ubah nama file menjadi HitungHarga
- Ubah nama package menjadi “proses”
- Akhiri dengan mengklik tombol Finish



- Klik tanda plus (+) di sebelah kiri HttpServlet Methods di bagian bawah source code servlet
- Maka akan tampil dua metode, yaitu doGet dan doPost.
 - ✓ doGet diproses ketika metode di tag form diisi get
 - ✓ doPost diproses ketika metode di tag form diisi post
- Secara default keduanya memanggil metode processRequest, pada contoh saat ini tidak perlu ada perubahan

⊞ HttpServlet methods. Click on the + sign on the left to edit the code.

```
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}
```

- Cari metode processRequest
- Di bawah pernyataan response.setContentType tambahkan beberapa baris kosong
- Untuk versi lama pembuatan objek PrintWriter (PrintWriter out = response.getWriter()) ditulis di atas try, tidak perlu diubah/disamakan

```
protected void processRequest(HttpServletRequest request, HttpServletResponse
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");

    try (PrintWriter out = response.getWriter()) {
        /* TODO output your page here. You may use following sample code. */
    }
}
```

Tambahkan source code untuk membaca parameter dan untuk menghitung total harga

```
response.setContentType("text/html;charset=UTF-8");
```

```
String namaBarang = request.getParameter("namaBarang");
String hargaSatuan = request.getParameter("hargaSatuan");
String jumlah = request.getParameter("jumlah");

int harga=0, jumlahBarang = 0, diskon=0, total;
try {
    harga = Integer.parseInt(hargaSatuan);
} catch (NumberFormatException ex) {}

try {
    jumlahBarang = Integer.parseInt(jumlah);
} catch (NumberFormatException ex) {}

total = harga * jumlahBarang;

if ((jumlahBarang >= 100) && (total >= 1000000)){
    diskon = ((int) (total * 0.05));
    total -= diskon;
}
```

```
try (PrintWriter out = response.getWriter()) {
```

- Ubah source code di dalam try menjadi seperti di bawah ini

```
try (PrintWriter out = response.getWriter()) {
    /* TODO output your page here. You may use following sample code. */
    out.println("<!DOCTYPE html>");
    out.println("<html>");
    out.println("<head>");
    out.println("<title>Menghitung Harga (Servlet)</title>");
    out.println("</head>");
    out.println("<body>");
    out.println("<h2>Hasil Penghitungan Harga</h2>");
    out.println("<form action=index.jsp method=post>");
    out.println("<table>");
    out.println("<tr>");
    out.println("<td>Nama Barang</td><td>:</td><td>"+namaBarang+"</td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Harga Satuan</td><td>:</td><td>"+hargaSatuan+"</td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Jumlah</td><td>:</td><td>"+jumlah+"</td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Diskon</td><td>:</td><td>"+diskon+"</td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Total</td><td>:</td><td>"+total+"</td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td colspan=3><input type=submit value=Kembali></td>");
    out.println("</tr>");
    out.println("</table>");
    out.println("</form>");
    out.println("</body>");
    out.println("</html>");
}
```

- Running aplikasi (project)

The image shows two screenshots of a web application running in a browser window titled 'Aries'.

Top Screenshot: Form Memasukkan Nilai

The browser address bar shows `localhost:32084/HitungHargaServlet/`. The page title is 'Menghitung Harga'. The form contains the following fields and values:

- Nama Barang: Nama Barang
- Harga Satuan: 10000
- Jumlah: 100

Below the form, there is a text description: "Diskon diberikan sebesar 5% jika jumlah ≥ 100 dan total harga sebelum diskon ≥ 1000000 ". A button labeled 'Hitung' is at the bottom.

Bottom Screenshot: Hasil Penghitungan Harga

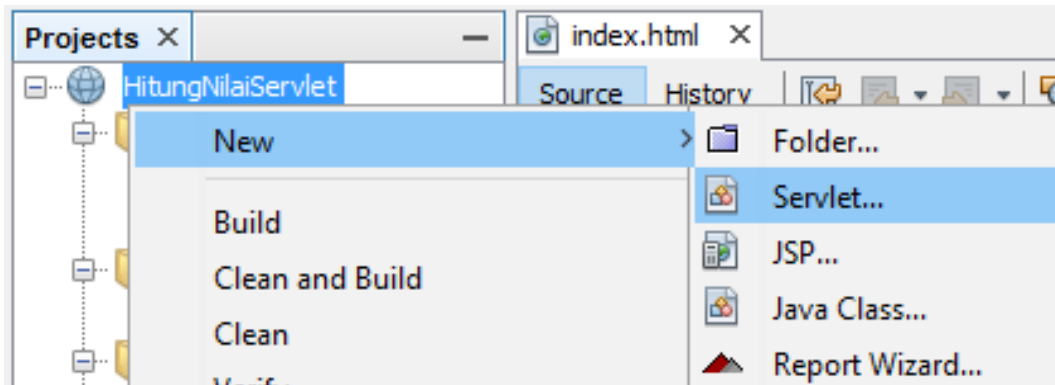
The browser address bar shows `084/HitungHargaServlet/HitungHarga`. The page title is 'Menghitung Harga (Servle x'. The results are displayed as follows:

- Nama Barang : Nama Barang
- Harga Satuan : 10000
- Jumlah : 100
- Diskon : 50000
- Total : 950000

A button labeled 'Kembali' is at the bottom.

. Contoh Aplikasi Servlet

- Buat peojek baru dengan nama HitungNilaiServlet
- Tambahkan file Servlet, caranya klik kanan pada Web Pages, pilih New ► Servlet ...



- Ubah nama file menjadi HitungNilai
- Mengubah Source Code

```
protected void processRequest(HttpServletRequest request, HttpServletResponse
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");

    try (PrintWriter out = response.getWriter()) {
        /* TODO output your page here. You may use following sample code. */
    }
}
```

- Di bawah setContentType tambahkan source code untuk membaca parameter, dan menghitung nilai

```
response.setContentType("text/html;charset=UTF-8");

String hadir = request.getParameter("hadir");
String pertemuan = request.getParameter("pertemuan");
String tugas = request.getParameter("tugas");
String uts = request.getParameter("uts");
String uas = request.getParameter("uas");

if (hadir == null) hadir="";
if (pertemuan == null) pertemuan="";
if (tugas == null) tugas="";
if (uts == null) uts="";
if (uas == null) uas="";

int jumlahHadir=0, jumlahPertemuan=0;
double nilaiTugas=0, nilaiUts=0, nilaiUas=0, nilaiAkhir;
String grade, status;
```

```

try {
    jumlahHadir = Integer.parseInt(hadir);
} catch (NumberFormatException ex) {}

try {
    jumlahPertemuan = Integer.parseInt(pertemuan);
} catch (NumberFormatException ex) {}

try {
    nilaiTugas = Double.parseDouble(tugas);
} catch (NumberFormatException ex) {}

try {
    nilaiUts = Double.parseDouble(uts);
} catch (NumberFormatException ex) {}

try {
    nilaiUas = Double.parseDouble(uas);
} catch (NumberFormatException ex) {}

nilaiAkhir = (10 * (double)jumlahHadir/jumlahPertemuan)
            + 0.2*nilaiTugas + 0.3*nilaiUts + 0.4*nilaiUas;

if ((nilaiAkhir >= 0) && (nilaiAkhir <=100)){
    if (nilaiAkhir >= 80){
        grade = "A";
        status = "Lulus";
    } else if (nilaiAkhir >= 70){
        grade = "B";
        status = "Lulus";
    } else if (nilaiAkhir >= 60){
        grade = "C";
        status = "Lulus";
    } else if (nilaiAkhir >= 50){
        grade = "D";
        status = "Tidak Lulus";
    } else {
        grade = "E";
        status = "Tidak Lulus";
    }
} else {
    grade = "X";
    status = "X";
}

try (PrintWriter out = response.getWriter()) {

```


- Ubah source code di dalam try menjadi seperti di bawah ini

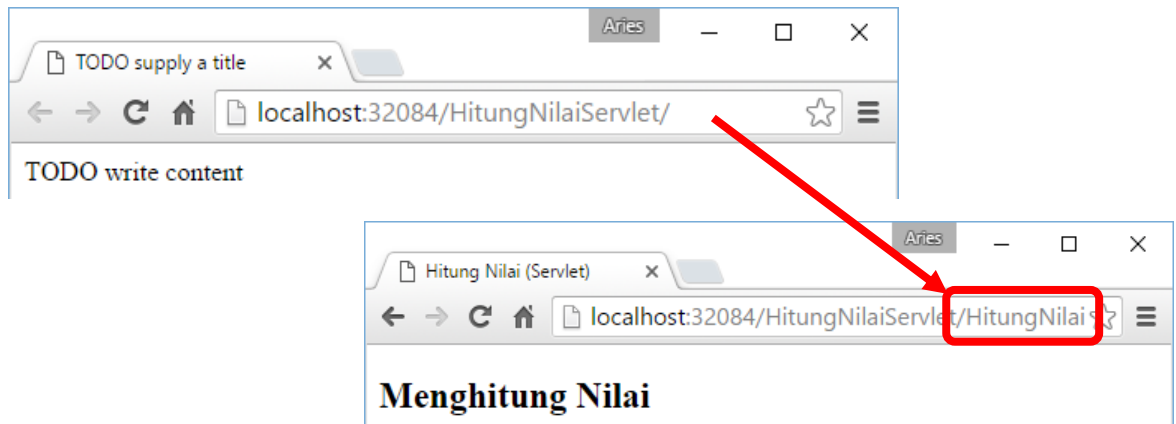
```
try (PrintWriter out = response.getWriter()) {
    /* TODO output your page here. You may use following sample code. */
    out.println("<!DOCTYPE html>");
    out.println("<html>");
    out.println("<head>");
    out.println("<title>Hitung Nilai (Servlet)</title>");
    out.println("</head>");
    out.println("<body>");
    out.println("<h2>Menghitung Nilai</h2>");
    out.println("<form action=HitungNilai method=post>");
    out.println("<table>");
    out.println("<tr>");
    out.println("<td>Jumlah hadir</td>");
    out.println("<td><input type=text name=hadir value="+hadir+"></td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Jumlah pertemuan</td>");
    out.println("<td><input type=text name=pertemuan value="+pertemuan+"></td>");
    out.println("</tr>");

    out.println("<tr>");
    out.println("<td>Nilai tugas</td>");
    out.println("<td><input type=text name=tugas value="+tugas+"></td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Nilai UTS</td>");
    out.println("<td><input type=text name=uts value="+uts+"></td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Nilai UAS</td>");
    out.println("<td><input type=text name=uas value="+uas+"></td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Nilai Akhir</td>");
    out.println("<td><input type=text readonly value="+nilaiAkhir+"></td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td>Grade</td>");
    out.println("<td><input type=text readonly value="+grade+"></td>");
    out.println("</tr>");

    out.println("<tr>");
    out.println("<td>Status</td>");
    out.println("<td><input type=text readonly value='"+status+"'></td>");
    out.println("</tr>");
    out.println("<tr>");
    out.println("<td colspan=2 align=center><input type=submit value=Hitung></td>");
    out.println("</tr>");
    out.println("</table>");
    out.println("</form>");
    out.println("</body>");
    out.println("</html>");
}
```

- Ketika dieksekusi akan ditampilkan file index.html

- Untuk mengeksekusi servlet HitungNilai, tambahkan HitungNilai di belakang alamat pada address bar seperti pada gambar



A screenshot of a web browser showing a form titled "Menghitung Nilai". The form contains several input fields for data entry:

Jumlah hadir	<input type="text"/>
Jumlah pertemuan	<input type="text"/>
Nilai tugas	<input type="text"/>
Nilai UTS	<input type="text"/>
Nilai UAS	<input type="text"/>
Nilai Akhir	<input type="text" value="NaN"/>
Grade	<input type="text" value="X"/>
Status	<input type="text" value="X"/>

Below the input fields is a button labeled "Hitung".

A screenshot of the same "Menghitung Nilai" form, but now with data entered and calculated results displayed:

Jumlah hadir	<input type="text" value="15"/>
Jumlah pertemuan	<input type="text" value="18"/>
Nilai tugas	<input type="text" value="68"/>
Nilai UTS	<input type="text" value="70"/>
Nilai UAS	<input type="text" value="75"/>
Nilai Akhir	<input type="text" value="72.93333333333334"/>
Grade	<input type="text" value="B"/>
Status	<input type="text" value="Lulus"/>

The "Hitung" button is still present at the bottom.

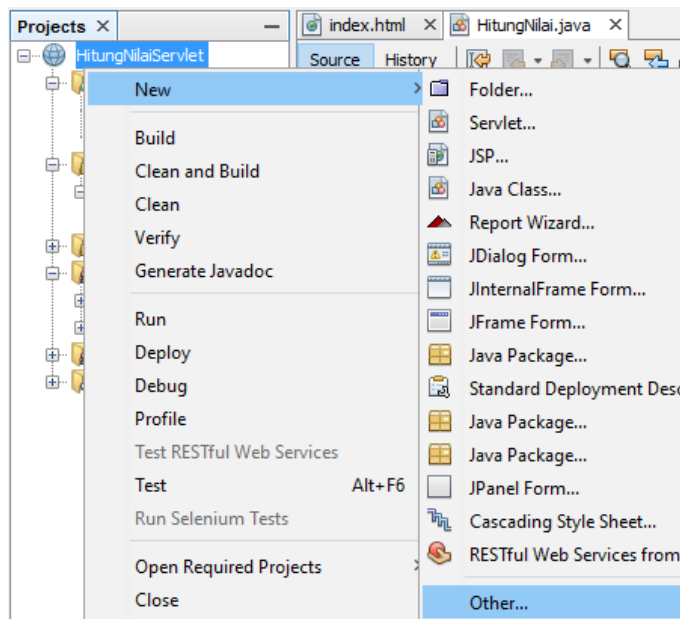
Pengaturan Servlet

Setting/pengaturan servlet dilakukan agar file servlet dapat langsung dijalankan. Pengaturan dilakukan melalui file Descriptor dan melalui mapping.

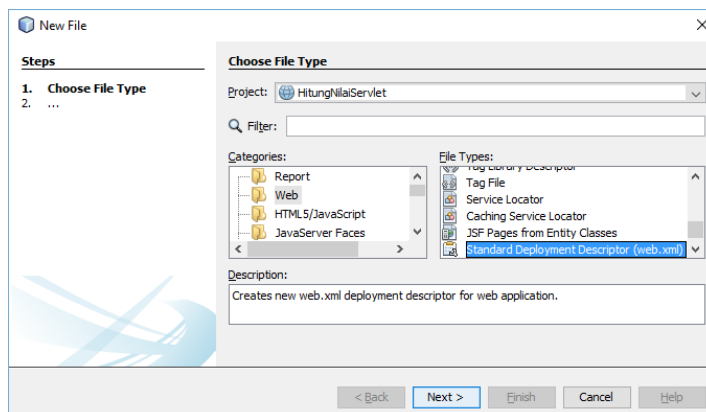
Cara 1: melalaui file Descriptor

Caranya:

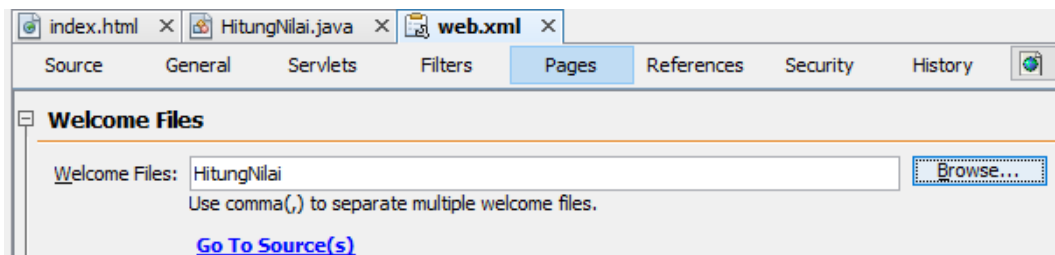
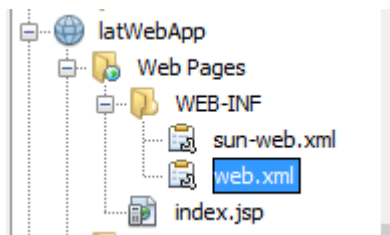
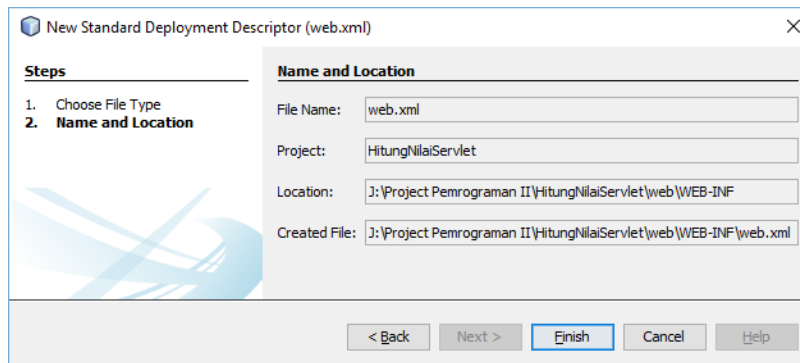
- ✓ Tambahkan File Descriptor dengan Klik kanan pada nama projek, pilih New ► Other... atau buka file **web.xml** (jika sudah ada)



- ✓ Pada Categories pilih Web, dan pada File Types pilih Standard Deployment Descriptor (web.xml)
- ✓ Lanjutkan dengan mengklik tombol Next >



- Akhir dengan mengklik tombol Finish

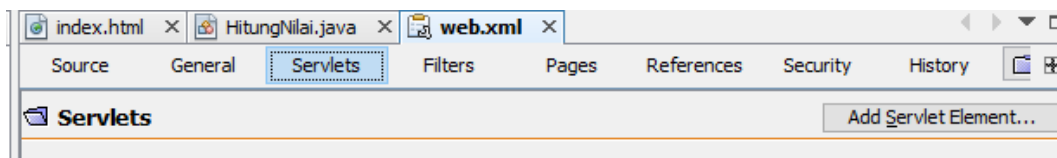


- Klik pada tab pages, ubah file yang dibuka pertama kali menjadi HitungNilai dengan mengubah Welcome Files
- Welcome Files bisa diisi lebih dari satu dengan menambahkan tanda koma sebagai pemisah
- Setelah diubah klik tab yang lain (Source, General, Servlets, dll) untuk memperbaharui perubahan
- Jalankan aplikasi, maka akan langsung membuka HitungNilai

Cara1: melalaui mapping

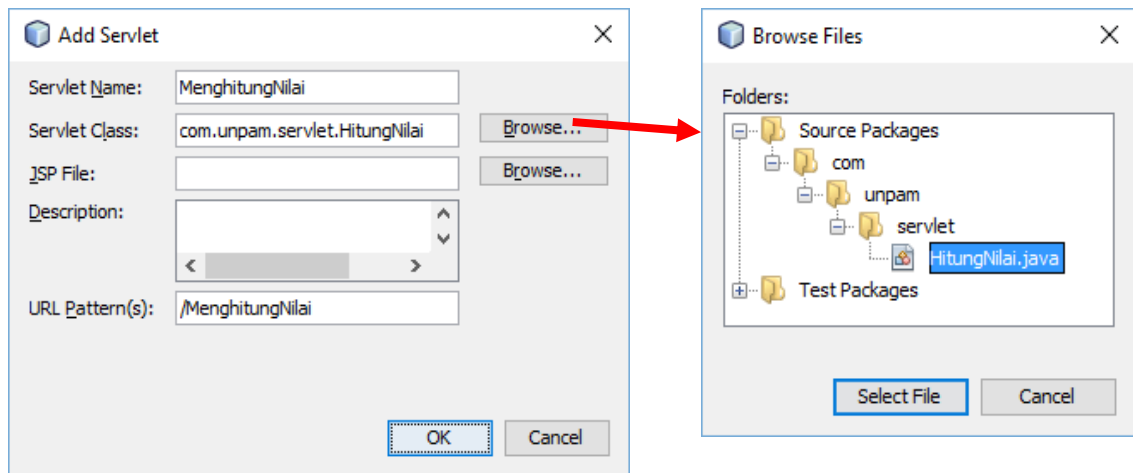
Caranya:

- Klik tab Servlets, kemudian klik Add Servlet Element...

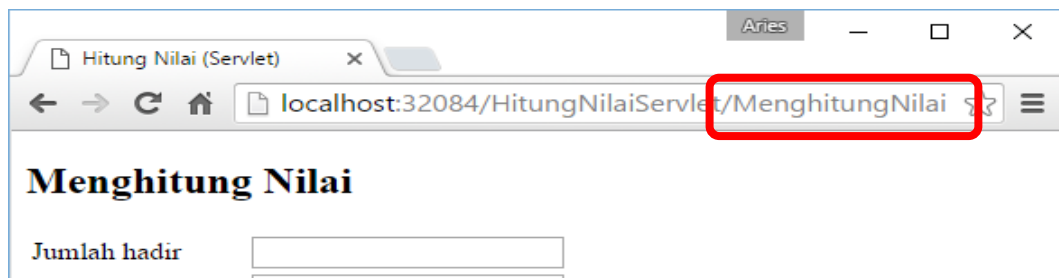


- Ubah nama servlet menjadi MenghitungNilai
- Pada Servlet Class klik tombol browse..., pilih HitungNilai.java

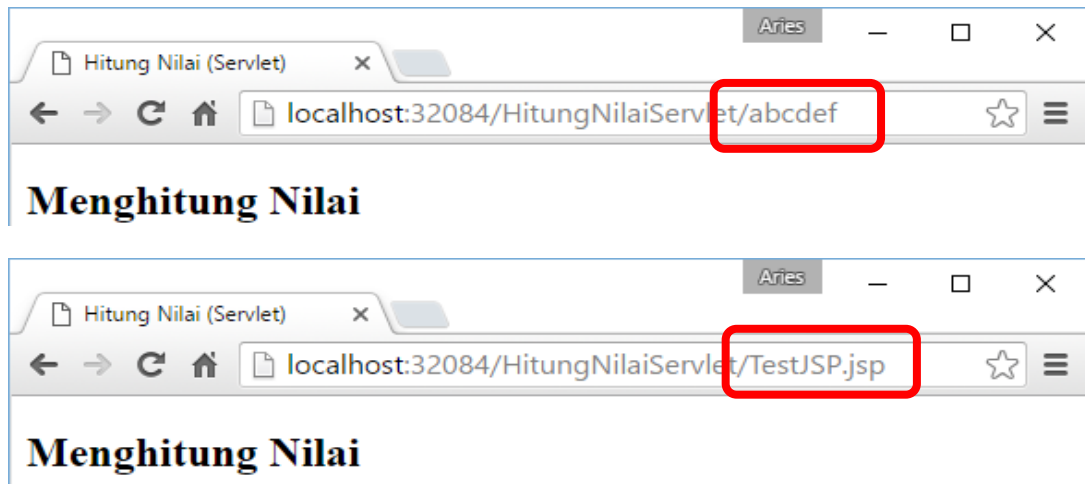
- Pada URL Pattern(s) ubah menjadi /MenghitungNilai, ini digunakan untuk memanggil servlet



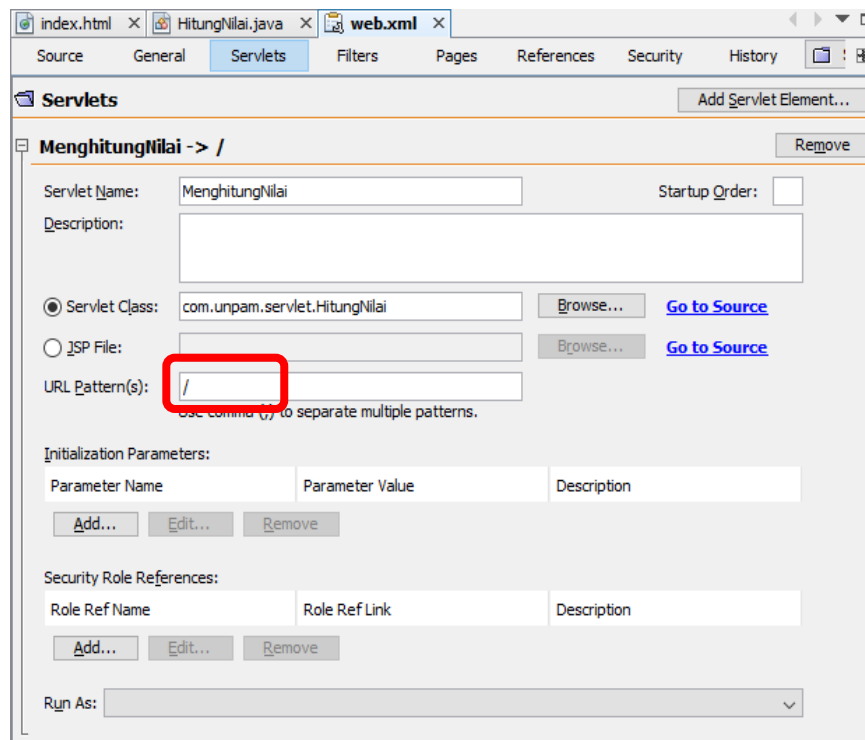
- Akhiri dengan mengklik tombol Ok
- Klik tab lain untuk memperbaharui perubahan
- Jalankan aplikasi, panggil dengan MenghitungNilai



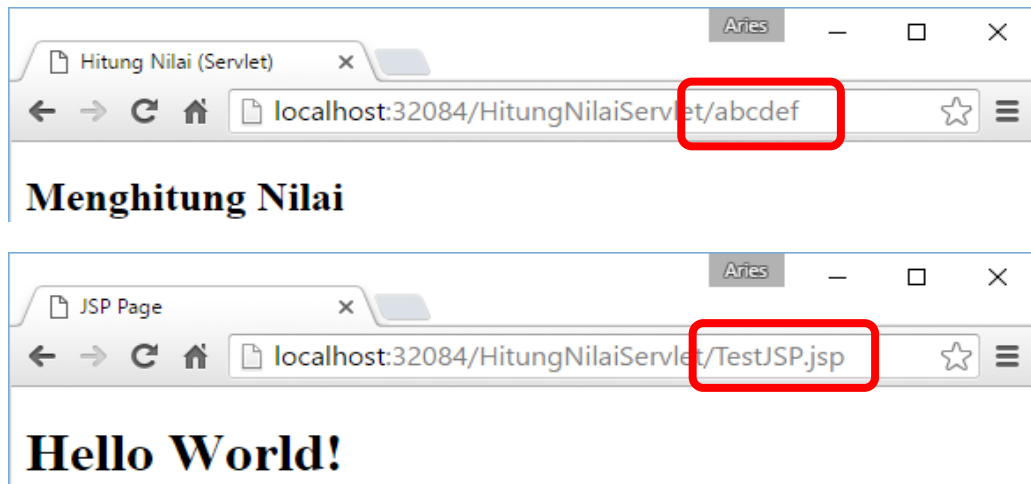
- Tambahkan file JSP dengan nama TestJSP
- Klik tab Servlets lagi, ubah URL Pattern(s) menjadi /*, maka ketika di address bar diketik apa saja akan memanggil servlet HitungNilai



Klik tab Servlets lagi, ubah URL Pattern(s) menjadi /



- Klik tab Servlets lagi, ubah URL Pattern(s) menjadi /
- Maka ketika di address bar memanggil file yang ada dalam proyek, maka akan membuka file tersebut
- Tetapi jika filenya tidak ditemukan maka akan membuka servlet HitungNilai



C. LATIHAN SOAL

Hitung luas persegi panjang

Panjang :

Lebar :

Panjang	Lebar	Luas
		0

Panjang :

Lebar :

Panjang	Lebar	Luas
15	7	105

D. DAFTAR PUSTAKA

Budiharto, W. (2004). *Pemrograman Web Mneggunakan J2EE*. Jakarta:

Ellexmedia Komputindo.

JENI, T. P. (2007). JENI 1-6.

Wijono, S. H., Suharto, B. H., & Wijono, M. S. (2006). *Pemrograman Java*

Servlet dan JSP dengan Netbeans. Yogyakarta: C.V ANDI OFFSET.