

# Java Web

## AULA 02 – SERVLETS E JSP BÁSICO

## Objetivos e Conceitos

- Objetivos:
  - Apresentar conceitos básicos sobre Servlets e JSP. Executar uma aplicação mínima
- Conceitos:
  - Servlets, JSP, Aplicação, Deploy, Undeploy

## Tópicos

---

1. JSP Básico
2. Servlets Básico
3. Arquivo de uma Aplicação Java Web
4. Deploy/Undeploy

## 1 JSP Básico

---

## JSP Básico

---

Criação de páginas dinâmicas

Baseado na tecnologia Java

- Vasta gama de bibliotecas
- Independente de plataforma

Processamento no servidor (*server-side*)

Seu código é convertido em Servlet na primeira execução

Tags especiais inseridas na página HTML

- `<% ... %>`
- Dentro coloca-se código Java
- Pode-se ter várias tags

## JSP Básico

---

Comando para imprimir um HTML

```
out.println("...");
```

Exemplo

```
out.println("Oi Mundo");  
out.println("Teste de String<br/>");  
out.println("<h2>Oi Mundo</h2>");
```

## JSP Básico – Exemplo 1

```
<html>
  <head><title>Titulo</title>
</head>
  <body>
    <b>Página HTML</b><br/>
    <h1><% out.println("Hello World!"); %></h1>
  </body>
</html>
```



Código Java

## JSP Básico – Exemplo 2

```
<html>
  <head><title>Titulo</title>
</head>
  <body>
    <b>Página HTML</b><br/>
    <h1><% out.println("Hello World!"); %></h1>
    <%
      for (int i=0; i<10; i++) {
        out.println("<h2>Valor = " + i + "</h2>");
      }
    %>
  </body>
</html>
```

## JSP Básico.

Pode-se usar qualquer API do Java

Deve-se fazer as importações de pacotes necessários com:

```
<%@page import="java.util.Date" %>
<%@page import="java.time.*,java.io.*" %>
```

Exemplo 3

```
<%@page import="java.util.Date" %>
<html><head><title>Titulo</title></head>
<body>
    <b>Data/Hora Atual</b><br/>
    <h1><% out.println( new Date() ); %></h1>
</body>
</html>
```

Prof. Dr. Razer A N R Montaña

JAVA WEB

9



## Exercícios..

1. Executar os JSPs de exemplo
  - a. Criar novo projeto no Netbeans
  - b. Adicionar um JSP para cada exemplo
  - c. Executar

Prof. Dr. Razer A N R Montaña

JAVA WEB

10

## 2 Servlets Básico

---

### Servlets Básico

---

Classes Java que dão como saída um texto em HTML

São invocadas pelo Servidor (Tomcat, JBoss, Glassfish, etc)

O servidor invoca métodos específicos destas classes

Toda a saída gerada em HTML é repassado ao cliente que requisitou a página

## Servlets Básico

O servidor invoca um método da Servlet conforme a requisição HTTP:

- Se for uma requisição GET -> invoca o `doGet()`
- Se for uma requisição POST -> invoca o `doPost()`

Dentro usa-se um `PrintWriter` para gerar o HTML

```
PrintWriter out = response.getWriter();
response.setContentType("text/html");

...

out.println("<h2>minha saída em HTML</h2>");
```

## Servlets Básico – Exemplo 1.

```
1. import java.io.*;
2. import javax.servlet.*;
3. import javax.servlet.http.*;
4.
5. @WebServlet("Teste")
6. public class Teste extends HttpServlet {
7.     public void doGet(HttpServletRequest request,
8.         HttpServletResponse response)
9.         throws ServletException, IOException {
10.
11.         PrintWriter out = response.getWriter();
12.         response.setContentType("text/html");
13.         out.println("<html><head><title>Teste</title></head>");
14.         out.println("<body>Teste de Servlet</body></html>");
15.         out.flush();
16.     }
17. }
```

## Servlets Básico

O plugin do Netbeans cria um método

`processRequest()`

Quando ocorre uma invocação, seja via GET ou via POST, os métodos `doGet()` e `doPost()` invocam o `processRequest()`

Então, para facilitar, basta você implementar o `processRequest()`

**CUIDADO: Não apagar as implementações de `doGet()` e `doPost()` senão a Servlet NÃO FUNCIONA**

## Servlets Básico - Netbeans

```
import *;

@WebServlet(urlPatterns = {"/TesteServlet1"})
public class TesteServlet1 extends HttpServlet {
    protected void processRequest(HttpServletRequest request,
                                  HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html; charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet TesteServlet1</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Servlet TesteServlet1 at " + request.getContextPath() + "</h1>");
            out.println("</body>");
            out.println("</html>");
        }
    }
}

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

Implementações de  
`doGet()` e  
`doPost()` escondidas.  
**NÃO APAGAR**



## Servlets Básico – Exemplo 2.

```
import *;  
  
@WebServlet(urlPatterns = {"/TesteServlet1"})  
public class TesteServlet1 extends HttpServlet {  
    protected void processRequest(HttpServletRequest request,  
        HttpServletResponse response)  
        throws ServletException, IOException {  
        response.setContentType("text/html;charset=UTF-8");  
        try (PrintWriter out = response.getWriter()) {  
            out.println("<!DOCTYPE html>");  
            out.println("<html>");  
            out.println("<head>");  
            out.println("<title>Teste TADS</title>");  
            out.println("</head>");  
            out.println("<body>");  
            out.println("<h1>Minha Primeira Servlet</h1>");  
            out.println("</body>");  
            out.println("</html>");  
        }  
    }  
}  
+  HttpServlet methods. Click on the + sign on the left to edit the code.  
}
```

Prof. Dr. Razer A N R Montañó

JAVA WEB

17



## Exercícios..

1. Executar a primeira Servlet
  - a. Criar novo projeto no Netbeans
  - b. Adicionar uma Servlet
  - c. Alterá-lo para o Exemplo 2 anterior
  - d. Executar

Prof. Dr. Razer A N R Montañó

JAVA WEB

18

# 3 Arquivo de uma Aplicação Java Web

---

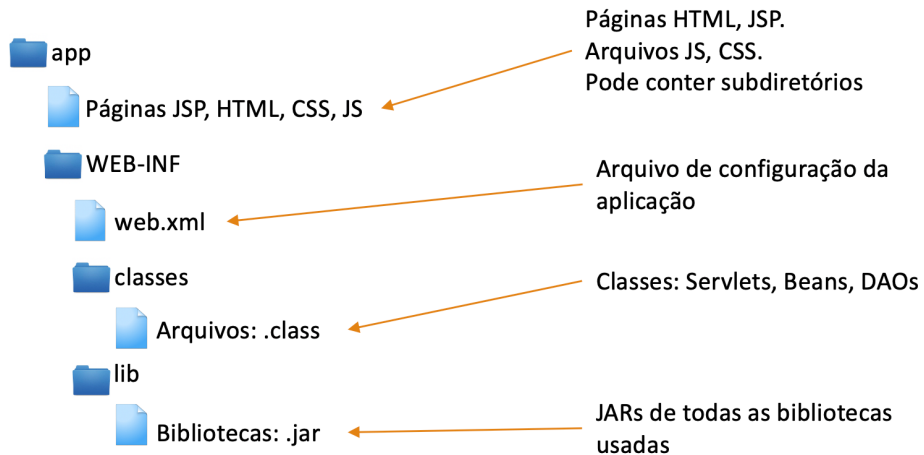
## Aplicação Java Web

---

- Aplicações Java Web estão empacotadas em arquivos **.WAR**
  - *Web Archive*
  - Zip
  - Estrutura de diretórios bem definida
  - Contém:
    - HTML, CSS, JS
    - JSPs
    - Servlets compiladas (.class)
    - Bibliotecas usadas na aplicação
    - Arquivos de configuração : web.xml

## Aplicação Java Web.

Arquivo: app.war



Prof. Dr. Razer A N R Montaña

JAVA WEB

21



## Exercícios..

1. Verificar a hierarquia de diretórios de um .WAR
  - a. Obter o .war de uma aplicação contendo um JSP e uma Servlet
    - **Netbeans**: Está no diretório DIST dentro do projeto
    - **Eclipse**: No projeto, escolher EXPORTAR para .WAR
  - b. Renomear o arquivo .war para .zip
  - c. Descompactar

Prof. Dr. Razer A N R Montaña

JAVA WEB

22

# 4 Deploy / Undeploy

## Deploy / Undeploy.


- **Deploy**
  - Ato de instalar a aplicação no servidor de aplicação
  - Já deve-se ter um arquivo .WAR
  - Portanto a aplicação já deve estar empacotada
  - Torna a aplicação disponível para ser executada
  - Em tempo de desenvolvimento
    - Pelo Netbeans
  - Em tempo de execução
    - Via portal do servidor
- **Undeploy**
  - Ato de desinstalar a aplicação do servidor de aplicação
  - Torna a aplicação indisponível

# Tomcat


- Servidor de Aplicações WEB
  - Não implementa toda a especificação Java EE
  - Serve aplicações WEB : Servlets, JSP
- Responde aplicação na porta 8080 por default  
`http://localhost:8080/app`  
`http://www.servidor.com.br:8080/app`
- Sem o nome da aplicação entra na página do Tomcat  
`http://localhost:8080`  
`http://www.servidor.com.br:8080`

# Tomcat: `http://localhost:8080`

Home Documentation Configuration Examples Wiki Mailing Lists Find Help

**Apache Tomcat/8.0.26**  The Apache Software Foundation  
<http://www.apache.org/>

If you're seeing this, you've successfully installed Tomcat. Congratulations!

 Recommended Reading:  
[Security Considerations HOW-TO](#)  
[Manager Application HOW-TO](#)  
[Clustering/Session Replication HOW-TO](#)

Server Status  
 Manager App  
 Host Manager

**Developer Quick Start**  
[Tomcat Setup](#) [Realms & AAA](#) [Examples](#) [Servlet Specifications](#)  
[First Web Application](#) [JDBC DataSources](#) [Tomcat Versions](#)

**Managing Tomcat**  
 For security, access to the [manager webapp](#) is restricted. Users are defined in:  
`$CATALINA_HOME/conf/tomcat-users.xml`  
 In Tomcat 8.0 access to the manager application is split between different users.  
[Read more...](#)  
[Release Notes](#)  
[Changelog](#)  
[Migration Guide](#)  
[Security Notices](#)

**Documentation**  
[Tomcat 8.0 Documentation](#)  
[Tomcat 8.0 Configuration](#)  
[Tomcat Wiki](#)  
 Find additional important configuration information in:  
`$CATALINA_HOME/RUNNING.txt`  
 Developers may be interested in:  
[Tomcat 8.0 Bug Database](#)  
[Tomcat 8.0 JavaDocs](#)  
[Tomcat 8.0 SVN Repository](#)

**Getting Help**  
**FAQ and Mailing Lists**  
 The following mailing lists are available:  
[tomcat-announce](#)  
 Important announcements, releases, security vulnerability notifications. (Low volume).  
[tomcat-users](#)  
 User support and discussion  
[taglibs-user](#)  
 User support and discussion for [Apache Taglibs](#)  
[tomcat-dev](#)  
 Development mailing list, including commit messages

# Tomcat

Apache Tomcat/8.0.26

The Apache Software Foundation  
http://www.apache.org/

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:  
[Security Considerations HOW-TO](#)  
[Manager Application HOW-TO](#)  
[Clustering/Session Replication HOW-TO](#)

Server Status  
**Manager App**  
 Host Manager

Developer Quick Start  
 Tomcat Setup  
 First Web Application  
 Realms & AAA  
 JDBC DataSources  
 Examples  
 Servlet Specifications  
 Tomcat Versions

Managing Tomcat  
 For security, access to the [manager webapp](#) is restricted. Users are defined in:  
`$CATALINA_HOME/conf/tomcat-users.xml`  
 In Tomcat 8.0 access to the manager application is split between different users.  
[Read more...](#)  
[Release Notes](#)  
[Changelog](#)  
[Migration Guide](#)  
[Security Notices](#)

Documentation  
[Tomcat 8.0 Documentation](#)  
[Tomcat 8.0 Configuration](#)  
[Tomcat Wiki](#)  
 Find additional important configuration information in:  
`$CATALINA_HOME/RUNNING.txt`  
 Developers may be interested in:  
 Tomcat 8.0 Bug Database  
 Tomcat 8.0 JavaDocs  
 Tomcat 8.0 SVN Repository

Getting Help  
[FAQ and Mailing Lists](#)  
 The following mailing lists are available:  
[tomcat-announce](#)  
 Important announcements, releases, security vulnerability notifications. (Low volume).  
[tomcat-users](#)  
 User support and discussion  
[taglibs-user](#)  
 User support and discussion for [Apache Taglibs](#)  
[tomcat-dev](#)  
 Development mailing list, including commit messages

Prof. Dr. Razer A N R Montaña      JAVA WEB      27

# Tomcat

Apache Tomcat/8.0.26

The Apache Software Foundation  
http://www.apache.org/

Inicie a sessão em localhost:8080  
 A sua senha será enviada sem ser criptografada.

Nome de Usuário  
 Senha  
☐ Memorizar esta senha

Cancelar    Iniciar Sessão

Developer Quick Start  
 Tomcat Setup  
 First Web Application  
 Realms & AAA  
 JDBC DataSources  
 Examples  
 Servlet Specifications  
 Tomcat Versions

Managing Tomcat  
 For security, access to the [manager webapp](#) is restricted. Users are defined in:  
`$CATALINA_HOME/conf/tomcat-users.xml`

Documentation  
[Tomcat 8.0 Documentation](#)  
[Tomcat 8.0 Configuration](#)  
[Tomcat Wiki](#)

Getting Help  
[FAQ and Mailing Lists](#)  
 The following mailing lists are available:  
[tomcat-announce](#)

Prof. Dr. Razer A N R Montaña      JAVA WEB      28

# Tomcat

## SE não conseguir acessar

### Usuário e Senha:

- Arquivo **tomcat-users.xml**
  - Dentro do subdiretório **CONF** da instalação do Tomcat
  - Criar um usuário/senha com as roles: manager-script, admin e manager-gui

### Deve ter o seguinte conteúdo:

```
<?xml version="1.0" encoding="UTF-8"?>
<tomcat-users>
  <role rolename="manager-gui"/>
  <user password="tomcat"
        roles="manager-script,admin,manager-gui"
        username="tomcat"/>
</tomcat-users>
```

# Tomcat



- ✓ Página Inicial
- ✓ Rolar até o final

## Tomcat Web Application Manager

Manager					
List Applications	HTML Manager Help		Manager Help		Server Status
Applications					
Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	<div>Start Stop Reload Undeploy</div> <div>Expire sessions with idle ≥ 30 minutes</div>
/PrimeiroProjetoLindo	None specified		true	0	<div>Start Stop Reload Undeploy</div> <div>Expire sessions with idle ≥ 30 minutes</div>
/TADSTarde10	None specified		true	0	<div>Start Stop Reload Undeploy</div> <div>Expire sessions with idle ≥ 30 minutes</div>
/TadsNoite2	None specified		true	0	<div>Start Stop Reload Undeploy</div> <div>Expire sessions with idle ≥ 30 minutes</div>
/WebApplication1	None specified		true	0	<div>Start Stop Reload Undeploy</div> <div>Expire sessions with idle ≥ 30 minutes</div>

# Tomcat

/manager	None specified	Tomcat Manager Application	true	1	<a href="#">Start</a> <a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Undeploy</a> Expire sessions with idle ≥ 30 minutes
----------	----------------	----------------------------	------	---	--

**Deploy**  
 Deploy directory or WAR file located on server

Context Path (required):   
 XML Configuration file URL:   
 WAR or Directory URL:

**WAR file to deploy**  
 Select WAR file to upload  nenhum arquivo selecionado

**Diagnostics**  
 Check to see if a web application has caused a memory leak on stop, reload or undeploy  
 This diagnostic check will trigger a full garbage collection. Use it with extreme caution on production systems.

**SSL connector configuration diagnostics**  
 List the configured ciphers for each connector

**Server Information**

Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture	Hostname	IP Address
Apache Tomcat/8.0.26	1.8.0_60-b27	Oracle Corporation	Mac OS X	10.13.2	x86_64	Neumann.local	192.168.25.100

# Tomcat

/manager	None specified	Tomcat Manager Application	true	1	<a href="#">Start</a> <a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Undeploy</a> Expire sessions with idle ≥ 30 minutes
----------	----------------	----------------------------	------	---	--

**Deploy**

Context Path (required):   
 XML Configuration file URL:   
 WAR or Directory URL:

**WAR file to deploy**  
 Select WAR file to upload  nenhum arquivo selecionado

**Diagnostics**  
 Check to see if a web application has caused a memory leak on stop, reload or undeploy  
 This diagnostic check will trigger a full garbage collection. Use it with extreme caution on production systems.

**SSL connector configuration diagnostics**  
 List the configured ciphers for each connector

**Server Information**

Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture	Hostname	IP Address
Apache Tomcat/8.0.26	1.8.0_60-b27	Oracle Corporation	Mac OS X	10.13.2	x86_64	Neumann.local	192.168.25.100

✓ Selecionar um arquivo .war



# Tomcat.



✓ Aplicação instalada

## Tomcat Web Application Manager

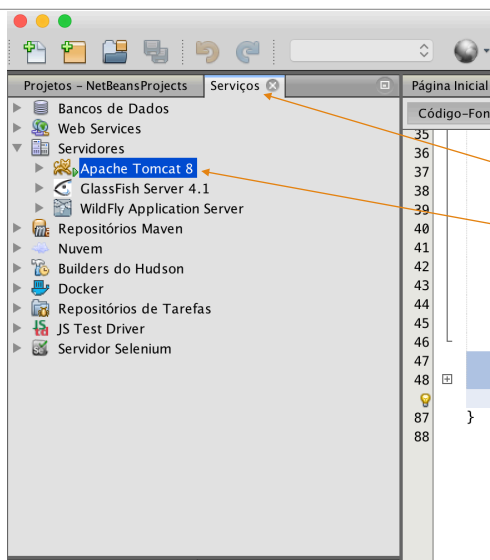
Message:		OK			
<b>Manager</b>					
List Applications	HTML Manager Help		Manager Help		Server Status
<b>Applications</b>					
Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/PrimeiroProjetoLindo	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/TADSTarde10	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/TadeNoite2	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/WebApplication1	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Prof. Dr. Razer A N R Montaña

JAVA WEB

33

# Tomcat no Netbeans



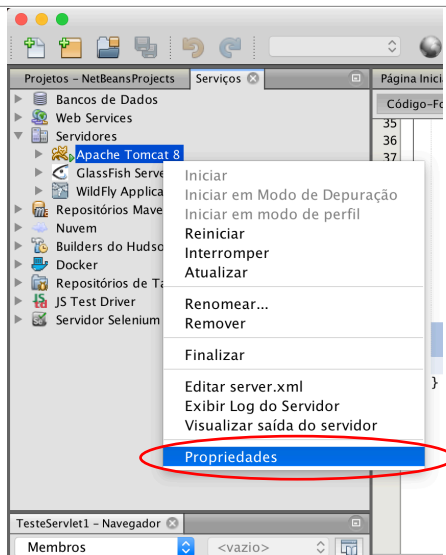
1. Aba Serviços
2. Botão Direito

Prof. Dr. Razer A N R Montaña

JAVA WEB

34

## Tomcat no Netbeans

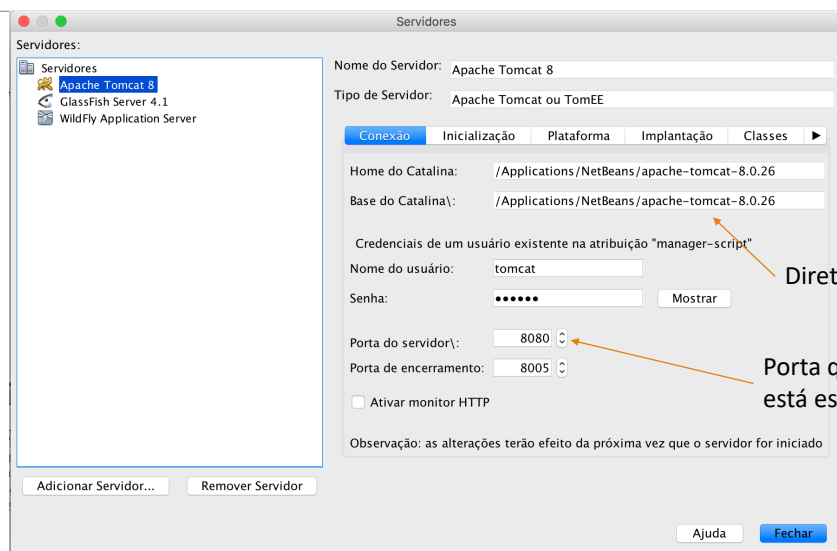


Prof. Dr. Razer A N R Montaña

JAVA WEB

35

## Tomcat no Netbeans.



Prof. Dr. Razer A N R Montaña

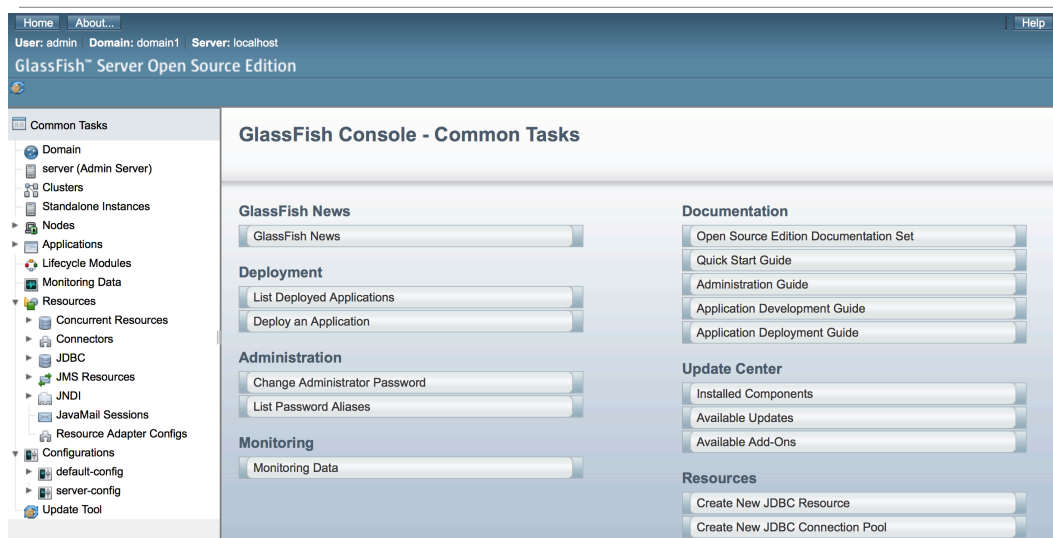
JAVA WEB

36

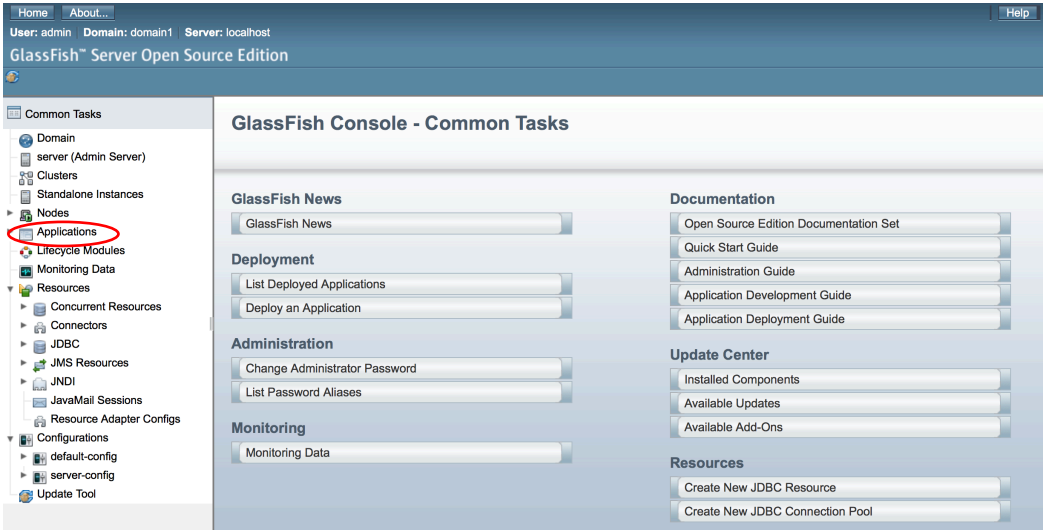
# Glassfish

- Servidor de Aplicações Java EE
  - Implementa toda a especificação Java EE
  - Serve aplicações Java EE completas
- Responde a aplicação na porta 8080 por default  
`http://localhost:8080/app`  
`http://www.servidor.com.br:8080/app`
- Responde a gerência na porta 4848 por default  
`http://localhost:4848`  
`http://www.servidor.com.br:4848`

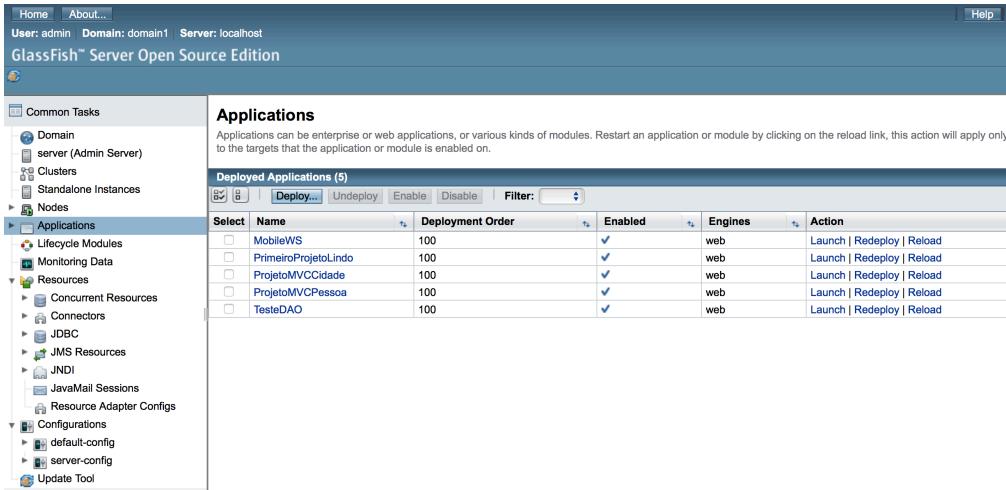
## Glassfish: `http://localhost:4848`



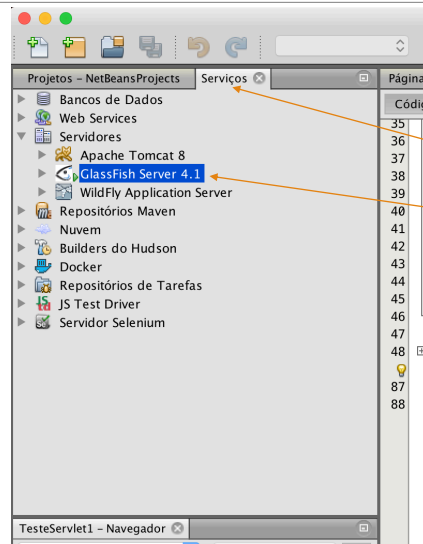
# Glassfish



# Glassfish.

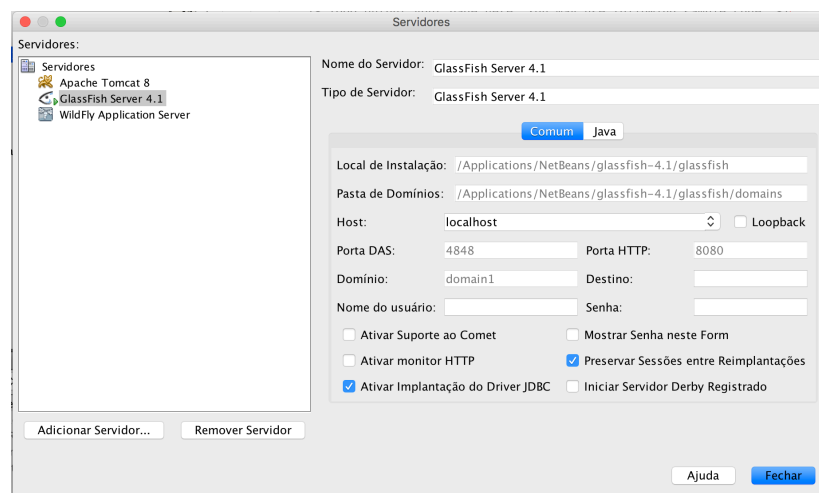


## Glassfish no Netbeans



1. Aba Serviços
2. Botão Direito
3. Propriedades

## Glassfish no Netbeans.





## Exercícios..

---

1. Escreva uma JSP que mostra seu nome 100 vezes (usando laço)
2. Escreva uma JSP que mostra a tabuada do 1 ao 9, usando laços e tabela HTML
3. Escreva uma Servlet que mostra a data e hora atuais
4. Escreva uma Servlet que mostra a tabuada do 1 ao 9, usando laços e tabela HTML