

# Laboratório de Programação de Web Sites Dinâmicos

HTML e páginas dinâmicas

Tassio Sirqueira – 2019/02

# | Servlets | Exibindo Informações na Web

- ❑ A única linguagem que os navegadores conseguem interpretar para a exibição de conteúdo é o HTML.
  - ❑ HTML (**H**yper**T**ext **M**arkup **L**anguage), que significa Linguagem de Marcação de Hipertexto
- ❑ A partir de declarações HTML podemos definir a estrutura de um documento.
  - ❑ Títulos
  - ❑ Cabeçalho
  - ❑ Barra de navegação
  - ❑ Rodapé
  - ❑ Conteúdo...



# | Servlets | Exibindo Informações na Web | Exemplo

## ❏ Exemplo:

- ❏ Suponha que você decide responder com o seguinte texto a solicitação de um navegador.



Mirror Fashion.

Bem-vindo à Mirror Fashion, sua loja de roupas e acessórios.

Confira nossas promoções.

Receba informações sobre nossos lançamentos por email.

Navegue por todos nossos produtos em catálogo.

Compre sem sair de casa.

# | Servlets | Exibindo Informações na Web | Exemplo



Mirror Fashion.

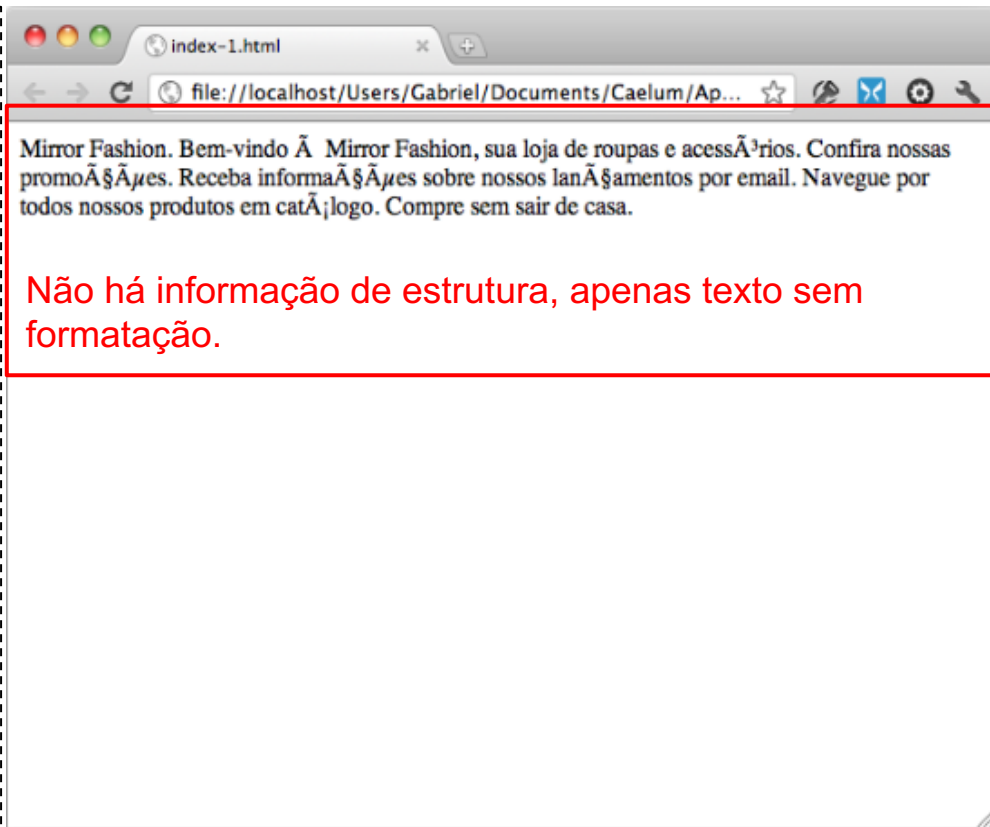
Bem-vindo à Mirror Fashion, sua loja de roupas e acessórios.

Confira nossas promoções.

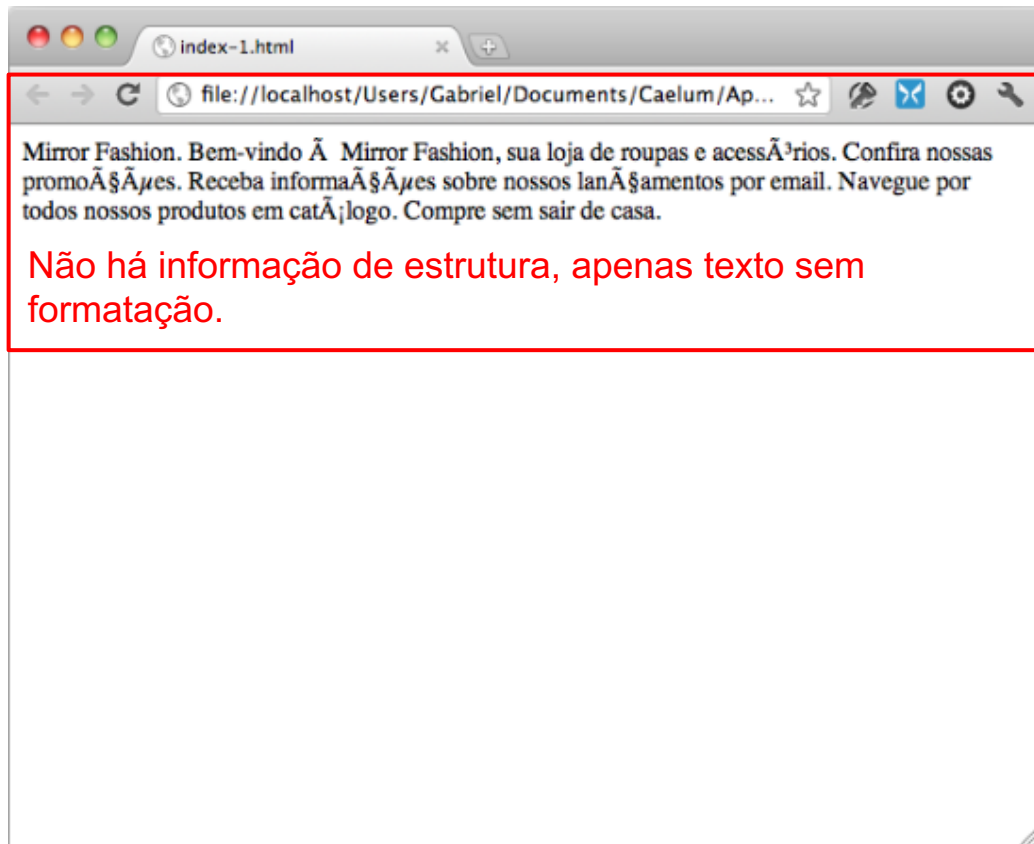
Receba informações sobre nossos lançamentos por email.

Navegue por todos nossos produtos em catálogo.

Compre sem sair de casa.



# | Servlets | Exibindo Informações na Web | Exemplo



- ❑ Os acentos não são exibidos corretamente.
- ❑ Não há quebra de linha.
- ❑ Não há marcação de título.
- ❑ Não há estrutura para lista.

**O HTML permite que a estrutura do documento seja definida!**

# | Servlets | Exibindo Informações na Web



**<!DOCTYPE html>**

```
<html>
  <head>
    <title>Mirror Fashion</title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1>Mirror Fashion.</h1>
    <h2>Bem-vindo à Mirror Fashion, sua loja de roupas
e acessórios.</h2>
    <ul>
      <li>Confira nossas promoções.</li>
      <li>Receba informações sobre nossos lançamentos
por email.</li>
      <li>Navegue por todos nossos produtos em
catálogo.</li>
      <li>Compre sem sair de casa.</li>
    </ul>
  </body>
</html>
```

Temos uma diretiva especial para indicar ao navegador a versão do HTML que deve ser utilizada.

A diretiva **<!DOCTYPE html>** indica que deve ser utilizada a versão corrente, nesse caso a versão HTML5.

# | Servlets | Exibindo Informações na Web



```
<!DOCTYPE html>
```

```
<html>
  <head>
    <title>Mirror Fashion</title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1>Mirror Fashion.</h1>
    <h2>Bem-vindo à Mirror Fashion, sua loja de roupas
e acessórios.</h2>
    <ul>
      <li>Confira nossas promoções.</li>
      <li>Receba informações sobre nossos lançamentos
por email.</li>
      <li>Navegue por todos nossos produtos em
catálogo.</li>
      <li>Compre sem sair de casa.</li>
    </ul>
  </body>
</html>
```

Todo o documento é definido por marcações, chamadas de **tags**.

Toda tag tem o formato é iniciada por **<tag>** e encerrada por **</tag>**.

As páginas HTML possuem uma tag “pai” que deve estar presente em todo o documento, que é a tag **<html>**.

# | Servlets | Exibindo Informações na Web



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Mirror Fashion</title>
```

```
<meta charset="utf-8">
```

```
</head>
```

```
<body>
```

```
<h1>Mirror Fashion.</h1>
```

```
<h2>Bem-vindo à Mirror Fashion, sua loja de roupas e acessórios</h2>
```

```
<ul>
```

```
<li>Confira nossas promoções.</li>
```

```
<li>Receba informações sobre nossos lançamentos por email.</li>
```

```
<li>Navegue por todos nossos produtos em catálogo.</li>
```

```
<li>Compre sem sair de casa.</li>
```

```
</ul>
```

```
</body>
```

```
</html>
```

A estrutura de uma página HTML é dividida em **cabeçalho (head)** e **corpo (body)**.

O cabeçalho de uma página é definido pela tag **<head>** e encerrada por **</head>**. Dentro dessa tag temos apenas metadados e declarações sobre o documento não visíveis na página.

O corpo de uma página é definido pela tag **<body>** e encerrada por **</body>**. Aqui estão todos os elementos exibidos na página.



# | Servlets | Exibindo Informações na Web



```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>Mirror Fashion</title>
```

```
    <meta charset="utf-8">
```

```
  </head>
```

```
  <body>
```

```
    <h1>Mirror Fashion.</h1>
```

```
    <h2>Bem-vindo à Mirror Fashion, sua loja de roupas e acessórios.</h2>
```

```
    <ul>
```

```
      <li>Confira nossas promoções.</li>
```

```
      <li>Receba informações sobre nossos lançamentos por email.</li>
```

```
      <li>Navegue por todos nossos produtos em catálogo.</li>
```

```
      <li>Compre sem sair de casa.</li>
```

```
    </ul>
```

```
  </body>
```

```
</html>
```

Dentro do cabeçalho, temos a tag **<title>** que define o título da página, exibido na barra de títulos ou na aba em que o documento está aberto.

# | Servlets | Exibindo Informações na Web



```
<!DOCTYPE html>
<html>
  <head>
    <title>Mirror Fashion</title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1>Mirror Fashion.</h1>
    <h2>Bem-vindo à Mirror Fashion, sua loja de roupa</h2>
    <ul>
      <li>Confira nossas promoções.</li>
      <li>Receba informações sobre nossos lançamentos</li>
      <li>Navegue por todos nossos produtos em catálogo</li>
      <li>Compre sem sair de casa.</li>
    </ul>
  </body>
</html>
```

No cabeçalho está definida a tag **<meta>** é utilizada para adicionar informações importantes que não são utilizadas na página.

Exemplos são: autor, data de modificação, descrição, palavras chave...

Para adicionar informações em algumas tags, utilizamos o conceito de **atributos**. Nesse caso declaramos o atributo **charset** com valor UTF-8.

Essa declaração define para o navegador o conjunto de caracteres utilizado na página, e corrige os acentos.

# | Servlets | Exibindo Informações na Web



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Mirror Fashion</title>
```

```
<meta charset="utf-8">
```

```
</head>
```

```
<body>
```

```
<h1>Mirror Fashion.</h2>
```

```
<h2>Bem-vindo à Mirror Fashion, sua loja de roupas e acessórios.</h2>
```

```
<ul>
```

```
<li>Confira nossas
```

```
<li>Receba informaço
```

```
<li>Navegue por todos
```

```
<li>Compre sem sair d
```

```
</ul>
```

```
</body>
```

```
</html>
```

No corpo da página temos várias tags para marcar estrutura.

Entre elas temos os títulos, sub-títulos, títulos de seção...

O HTML possui 5 tags para indicar títulos e subtítulos em diversos níveis: **<h1>**, **<h2>**, **<h3>**, **<h4>** e **<h5>**.

O h1 é o título principal e o h5 o menor nível de subtítulo.

# | Servlets | Exibindo Informações na Web



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Mirror Fashion<
```

```
<meta charset="utf-8">
```

```
</head>
```

```
<body>
```

```
<h1>Mirror Fashion
```

```
<h2>Bem-vindo Mirror
```

```
<ul>
```

```
<li>Confira nossas promoções.</li>
```

```
<li>Receba informações sobre nossos lançamentos por email.</li>
```

```
<li>Navegue por todos nossos produtos em catálogo.</li>
```

```
<li>Compre sem sair de casa.</li>
```

```
</ul>
```

```
</body>
```

```
</html>
```

Para exibir listas estruturadas temos tags particularmente úteis no HTML.

O **<ul>** permite a declaração de listas não ordenadas.

Cada item da lista pe definido pela tag **<li>**.

# | Servlets | Exibindo Informações na Web | Exemplo



`<title> </title>`

Observe que os acentos  
estão preservados  
devido ao  
`<meta charset="utf-8">`

**Mirror Fashion.**

`<h1> </h1>`

**Bem-vindo à Mirror Fashion, sua loja de roupas e acessórios.**

`<h1> </h1>`

- Confira nossas promoções. `<i> </i>`
- Receba informações sobre nossos lançamentos por email. `<i> </i>`
- Navegue por todos nossos produtos em catálogo. `<i> </i>`
- Compre sem sair de casa. `<i> </i>`

`<ul> </ul>`

`<body> </body>`

# | Servlets | Outras tags importantes!



## ❑ Parágrafos

```
<p>Um parágrafo de texto.</p>
```

```
<p>Outro parágrafo de texto.</p>
```

## ❑ Ênfase

```
<p>Compre suas roupas e acessórios na <strong>Mirror Fashion</strong>.
```

## ❑ Links

```
<p> Visite<a href="http://www.site.com.br">o site</a>.</p>
```

## ❑ Imagens

```

```

## ❑ Alguma área na estrutura da página

```
<div> ... </div>
```



# | Servlets | Outras tags importantes! | Formulários

## ☐ Caixas de texto

```
Nome: <input type="text" name="fname">  
E-mail: <input type="email" name="email_usuario" />  
Receber newsletter: <input type="checkbox" name="news" />  
E-mail público <input type="checkbox" name="publico" />
```

## ☐ Botões

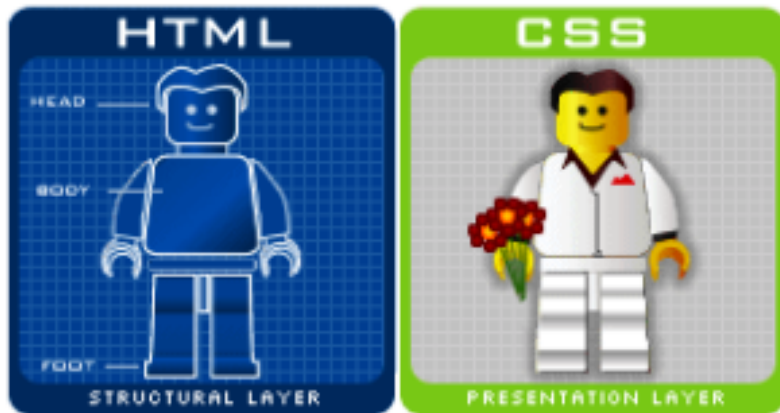
```
<button name="button">Click me</button>
```

## ☐ Formulários

```
<form action="/nomes" method="post">  
  Nome: <input type="text" name="nome">  
  <input type="submit" value="Submit">  
</form>
```

# | Servlets | E como mudar a aparência da minha página?

- ❑ A finalidade da linguagem HTML é definir a estrutura do documento.
- ❑ Para definir a aparência ou o “estilo” de um documento utilizamos o CSS.
  - ❑ Cascading Style Sheets (CSS) - Folhas de Estilo em Cascata.





# | Servlets | E como mudar a aparência da minha página?



- ❑ Uma outra forma de estilizar um documento, e mais recomendada, é usando um “seletor” para alterar todos os elementos com determinada característica em uma página.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Sobre a Mirror Fashion</title>
    <style>
      p {
        background-color: yellow;
        color: blue;
      }
    </style>
  </head>
  <body>
    <p> O conteúdo desta tag será exibido em azul com fundo amarelo! </p>
  </body>
</html>
```

O seletor **nome-da-tag** aplica as regras de estilo para todos os elementos daquela tag na página.

# | Servlets | E como mudar a aparência da minha página?



❏ Uma outra forma de definir o que deve ser exibido é através de classes.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Sobre a Mirror Fashion</title>
    <style>
      .destaque {
        background-color: yellow;
        color: blue;
      }
    </style>
  </head>
  <body>
    <p class="destaque"> O conteúdo desta tag será exibido em azul com fundo amarelo! </p>
    <p> O conteúdo desta tag será preto. </p>
  </body>
</html>
```

O seletor **.classe** aplica as regras de estilo para todos os elementos com determinada classe na página.

Para definir uma classe em um elemento utilize o atributo **class**.

# | Servlets | E como mudar a aparência da minha página?



❏ Arquivos de estilo podem ainda ser importados de outros arquivos.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Sobre a Mirror Fashion</title>
    <link rel="stylesheet" href="estilos.css">
  </head>
  <body>
    <p class="destaque"> O conteúdo desta tag será exibido
em azul com fundo amarelo! </p>
    <p> O conteúdo desta tag será preto. </p>
  </body>
</html>
```

A tag **link** importa o arquivo cuja URI está definida pelo atributo **href**.

```
/* conteúdo do arquivo estilos.css */
.destaque {
  background-color: yellow;
  color: blue;
}
```

# | Servlets | Outras propriedades de estilo importantes!



## ❑ Dimensões

```
height: 300px;  
width: 300px;
```

## ❑ Espaçamento

```
padding: 10px 20px 15px 5px;
```

## ❑ Borda

```
border: 1px 2px 1px 2px;
```

## ❑ Alinhar à esquerda ou direita

```
float: right;
```

## ❑ Centralizar horizontalmente.

```
margin: 0 auto;
```

Cores, fontes, posicionamento, transformações...

# | **Servlets** | E como adicionar comportamentos à minha página?

- ❑ A finalidade da linguagem HTML é definir a estrutura do documento, e do CSS é definir a aparência do mesmo.
- ❑ Para adicionar comportamento à página, podemos utilizar JavaScript.
  - ❑ Apesar do nome, a linguagem JavaScript tem pouco a ver com o Java.
- ❑ Suportada por diferentes navegadores.
- ❑ Inicialmente utilizada apenas para scripts simples.
- ❑ Hoje em dia é uma linguagem poderosa e completa.
  - ❑ Diversas aplicações além da web.



# | Servlets | E como adicionar comportamentos à minha página?

- ❑ Imperativa e Estruturada
  - ❑ if, while, do...while no estilo C.
- ❑ Baseada em objetos
  - ❑ Objetos JavaScript são arrays associativos.
  - ❑ Podem ser extendidos por meio de protótipos.
- ❑ Funcional
  - ❑ Funções são tipos de “primeira classe”
    - ❑ Podem ser passados como argumentos, serem atribuídos a variáveis ou retornados como qualquer outro objeto.



# | Servlets | E como adicionar comportamentos à minha página?

```
var cars = [{type:"Volvo", year:2016}, {type:"Saab", year:2001},{type:"BMW", year:2010}]

function displayCars() {
  document.getElementById("demo").innerHTML =
  cars[0].type + " " + cars[0].year + "<br>" +
  cars[1].type + " " + cars[1].year + "<br>" +
  cars[2].type + " " + cars[2].year;
}

function funcao() {
  cars.sort(function(a, b){return a.year - b.year});
  displayCars();
}

funcao();
```



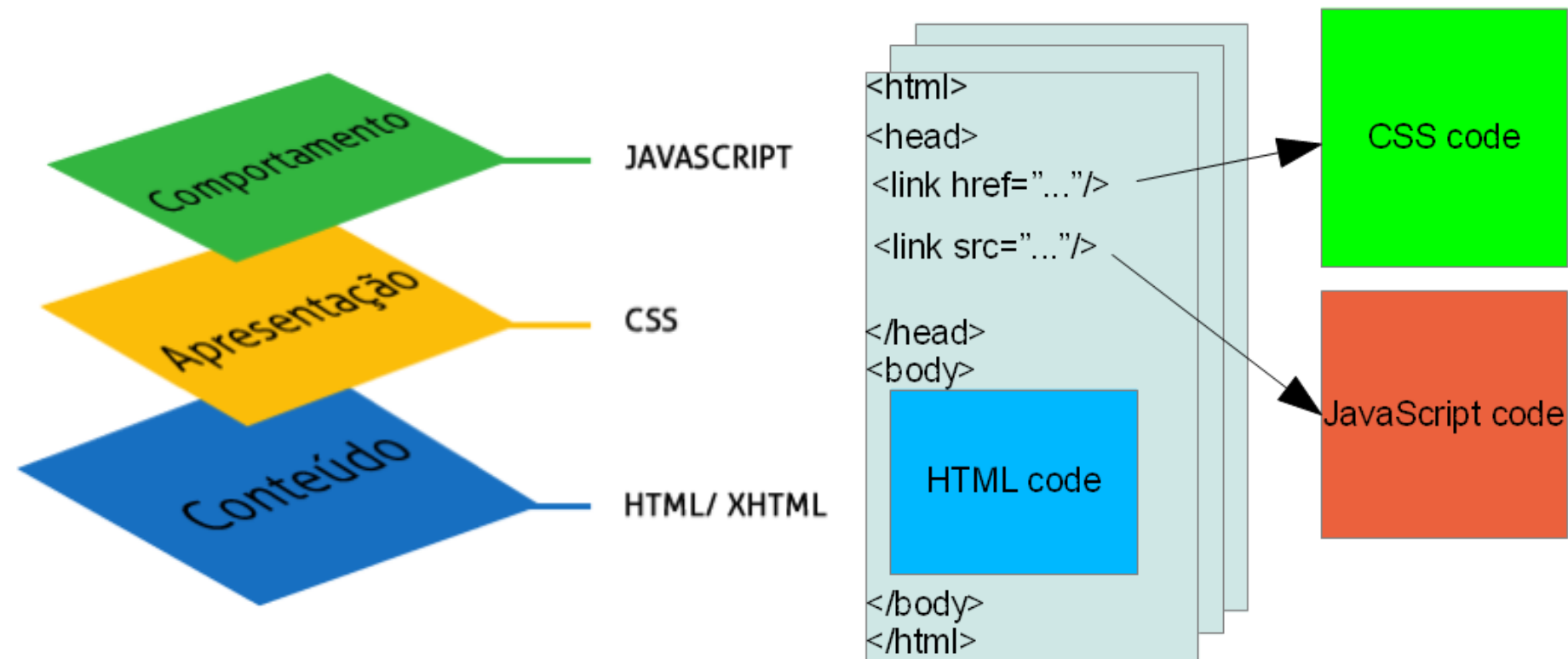
# | **Servlets** | E como adicionar comportamentos à minha página?

- ❑ JavaScript é poderoso porque permite que código seja executado no lado cliente.
  - ❑ Seu código pode rodar dentro do browser, na máquina do cliente.
- ❑ No entanto existem várias necessidades reais de aplicações web que não podem ser resolvidas dessa forma.
  - ❑ Salvamento de dados em um banco de dados.
  - ❑ Regras de negócio e validações.
  - ❑ Autenticação e autorização.
- ❑ O código JavaScript deve
  - ❑ Oferecer uma página mais dinâmica e agradável para navegação.
  - ❑ Melhorar a experiência do usuário por meio de interfaces e dinâmicas validações.





# | Servlets | E como adicionar comportamentos à minha página?



# | Prática | Disponibilizando uma página Web Estática no Glassfish

❑ Do que eu preciso para começar?

❑ **JDK** = Java Development Kit

❑ <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

**Sugiro fazer download do netbeans separado.**

**Para usuários de Debian / Ubuntu / etc.. sugiro a instalação via PPA.**

**Oracle** Menu

Oracle Technology Network > Java > Java SE > Downloads

**Java SE Downloads**

Overview Downloads Documentation

Java Platform (JDK) 8u144

NetBeans with JDK 8

**Java SE Development Kit 8 Downloads**

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter: From your Oracle account, select **Subscriptions**, expand **Technology**, and subscribe to **Java**.
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK 8u144 checksum

**Java SE Development Kit 8u144**

You must accept the **Oracle Binary Code License Agreement for Java SE** to download this software.

☒ Accept License Agreement ☐ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.89 MB	<a href="#">jdk-8u144-linux-arm32-vfp-hflt.tar.gz</a>
Linux ARM 64 Hard Float ABI	74.63 MB	<a href="#">jdk-8u144-linux-arm64-vfp-hflt.tar.gz</a>
Linux x86	164.65 MB	<a href="#">jdk-8u144-linux-i586.rpm</a>
Linux x86	179.44 MB	<a href="#">jdk-8u144-linux-i586.tar.gz</a>
Linux x64	162.1 MB	<a href="#">jdk-8u144-linux-x64.rpm</a>
Linux x64	176.92 MB	<a href="#">jdk-8u144-linux-x64.tar.gz</a>
Mac OS X	226.6 MB	<a href="#">jdk-8u144-macosx-x64.dmg</a>
Solaris SPARC 64-bit	139.87 MB	<a href="#">jdk-8u144-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	99.18 MB	<a href="#">jdk-8u144-solaris-sparcv9.tar.gz</a>
Solaris x64	140.51 MB	<a href="#">jdk-8u144-solaris-x64.tar.Z</a>
Solaris x64	96.99 MB	<a href="#">jdk-8u144-solaris-x64.tar.gz</a>
Windows x86	190.94 MB	<a href="#">jdk-8u144-windows-i586.exe</a>

# | Prática | Disponibilizando uma página Web Estática no Glassfish

❑ Do que eu preciso para começar?

❑ IDE Netbeans

NetBeans IDE 8.2 Download

8.1 | 8.2 | Development | Archive

Email address (optional):

Subscribe to newsletters: ☒ Monthly ☐ Weekly

☒ NetBeans can contact me at this address

IDE Language: English Platform: Linux (x86/x64)

Note: Greyed out technologies are not supported for this platform.

**NetBeans IDE Download Bundles**

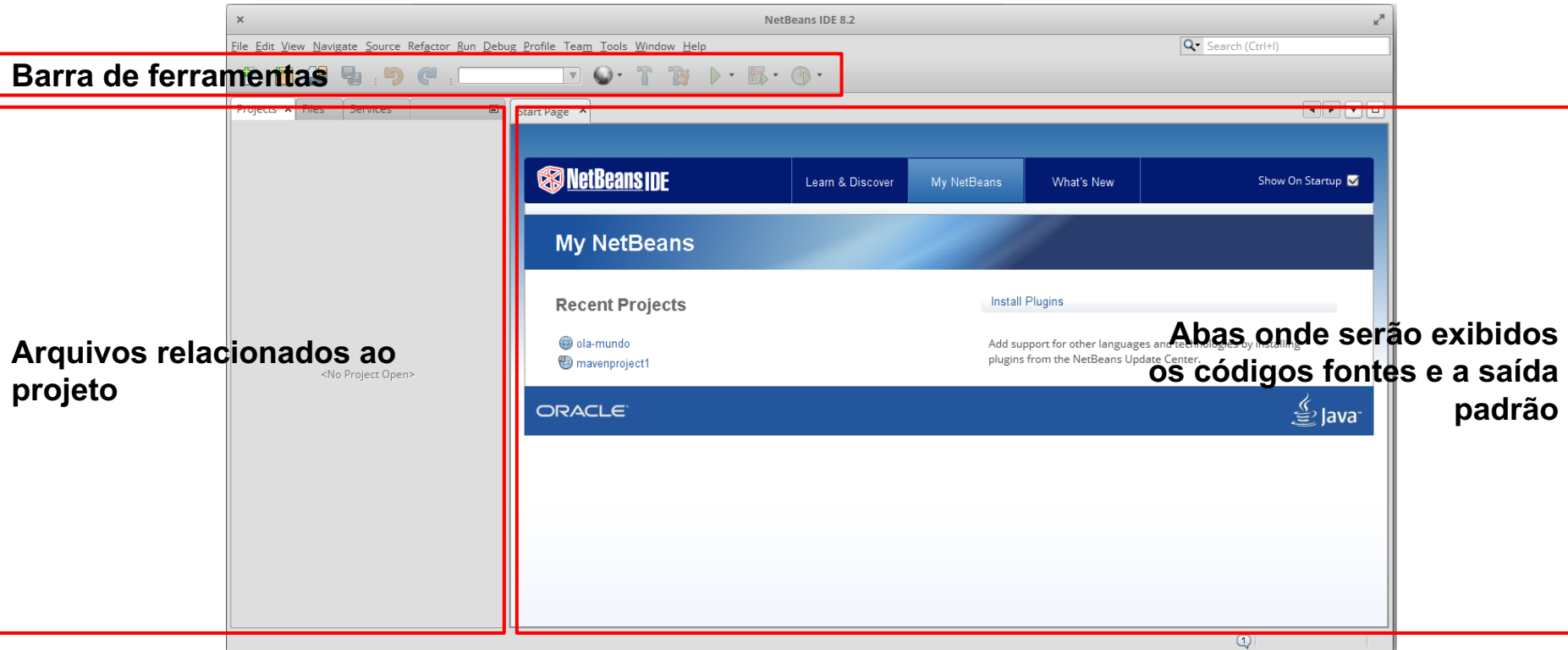
Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME						•
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						—
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

Download Download Download x86 Download x86 Download x86 Download x64 Download x64 Download x64 Download

Free, 94 MB Free, 196 MB Free, 116 - 119 MB Free, 116 - 119 MB Free, 115 - 117 MB Free, 214 MB

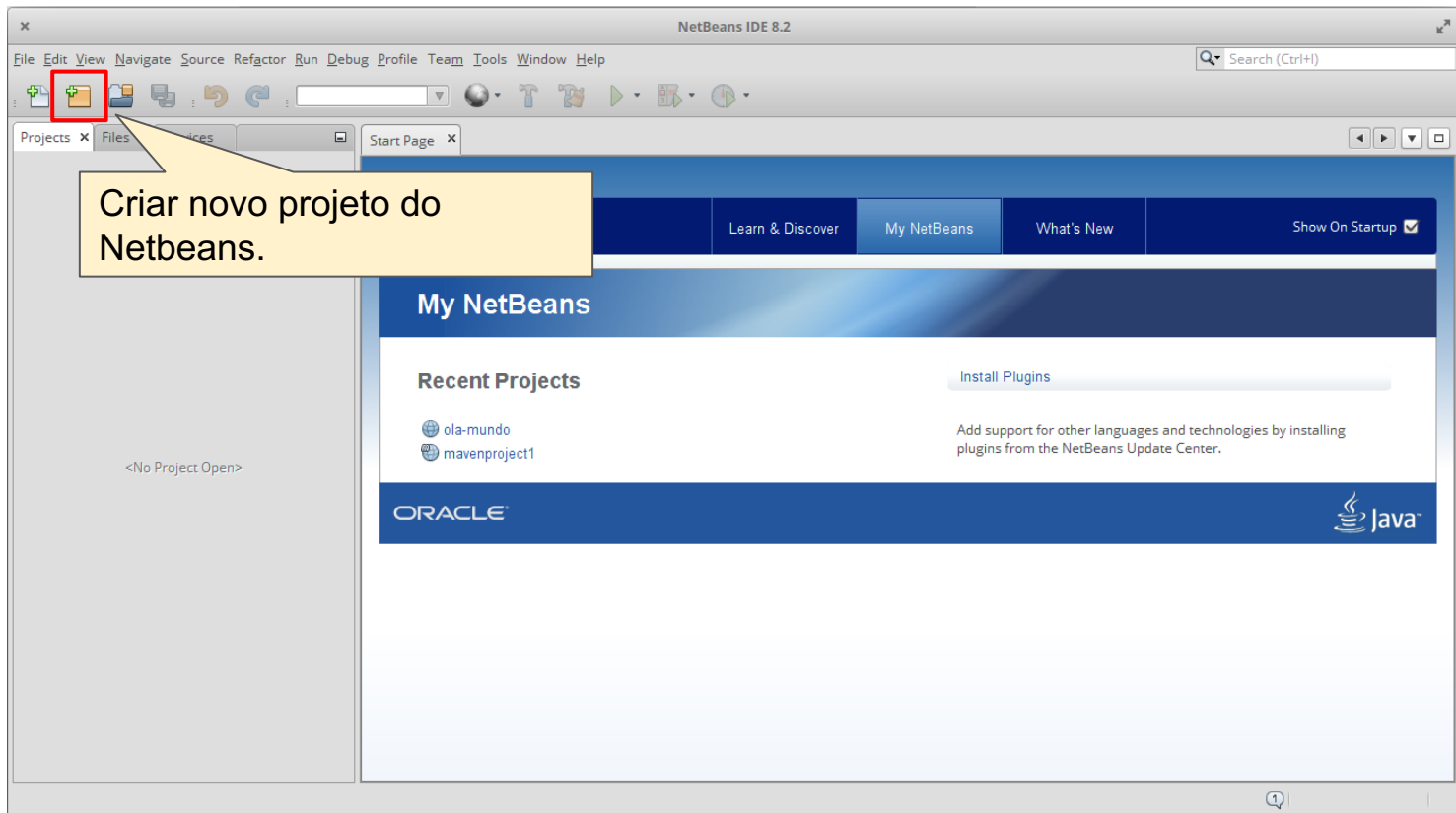
# | Prática | Disponibilizando uma página Web Estática no Glassfish

- ❑ Instale o JDK e o NetBeans, durante a instalação não desmarque a instalação conjunta do Glassfish.



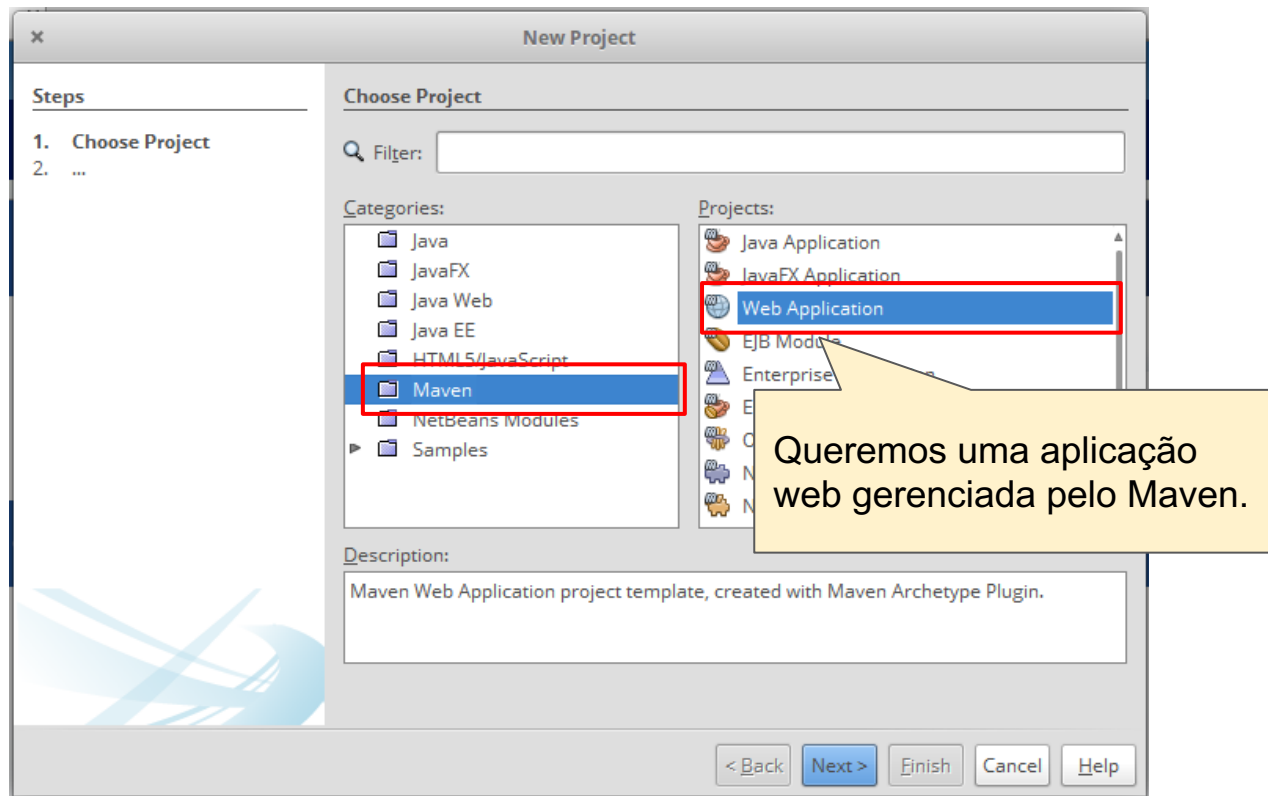
# | Prática | Disponibilizando uma página Web Estática no Glassfish

- ❑ Crie novo projeto a partir do ícone indicado.



# | Prática | Disponibilizando uma página Web Estática no Glassfish

- ❑ Escolha o tipo de projeto.



# | Prática | Disponibilizando uma página Web Estática no Glassfish

- ❑ Apache Maven ( Maven) é uma ferramenta de automação de compilação.
  - a. Permite o gerenciamento de dependências facilitado, sem necessidade de baixar bibliotecas e suas dependências diretamente.
  - b. Permite a configuração da compilação.
  - c. Facilita o gerenciamento de configuração do projeto.
  - d. Extensível por meio de uma variedade de plugins.
  - e. Deixa o projeto independente da IDE.
- ❑ Se você criar um projeto sem Maven todo o código funcionará normalmente.
  - a. Porém gerenciado pelo Netbeans.



# | Prática | Disponibilizando uma página Web Estática no Glassfish

## ❑ Parâmetros do projeto

The screenshot shows the 'New Web Application' dialog box. The 'Steps' pane on the left indicates the current step is '2. Name and Location'. The 'Name and Location' section contains several fields:

- Project Name:** exemplo (highlighted with a red box and labeled 'Nome do projeto')
- Project Location:** /home/welington/Workspace/professor-netbeans (highlighted with a red box and labeled 'Pasta onde será criado'). A 'Browse...' button is next to it.
- Project Folder:** /home/welington/Workspace/professor-netbeans/exemplo
- Artifact Id:** exemplo
- Group Id:** com.mycompany (highlighted with a red box and labeled 'Nome da organização')
- Version:** 1.0-SNAPSHOT (highlighted with a red box and labeled 'Versão')
- Package Name:** com.mycompany.exemplo (Optional) (highlighted with a red box and labeled 'Pacote base')

At the bottom of the dialog are buttons for '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'.

Em Java, pacotes e group ids são por padrão o DNS reverso da organização.

Ex:

- ❑ google.com
  - ❑ Com.google
- ❑ globoesporte.g1.com.br
  - ❑ Br.com.g1.globoesporte
- ❑ www.cesjf.br
  - ❑ br.cesjf



# | Prática | Disponibilizando uma página Web Estática no Glassfish

## ❑ Configuração do Servidor de Aplicação (AS)

The screenshot shows the 'New Web Application' wizard in Glassfish. The 'Settings' step is selected in the 'Steps' list on the left. The 'Server' dropdown is set to 'GlassFish Server 4.1.1' and the 'Java EE Version' dropdown is set to 'Java EE 7 Web'. The 'Finish' button at the bottom right is highlighted with a red box.

**Steps**

1. Choose Project
2. Name and Location
3. Settings

**Settings**

Server: GlassFish Server 4.1.1

Java EE Version: Java EE 7 Web

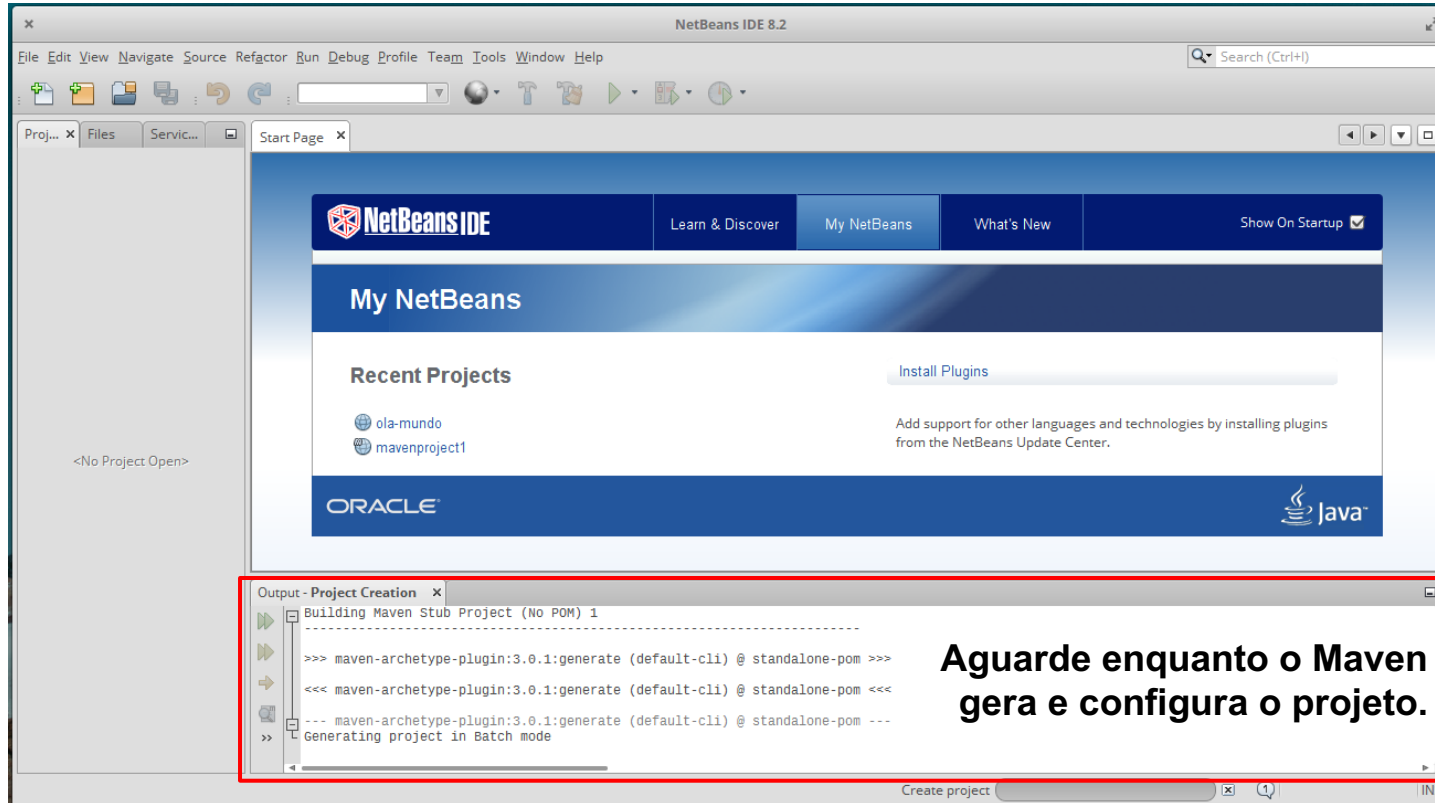
< Back Next > **Finish** Cancel Help

**Selezione o AS Glassfish 4**

**Selezione a versão 7 do Java EE**

# | Prática | Disponibilizando uma página Web Estática no Glassfish

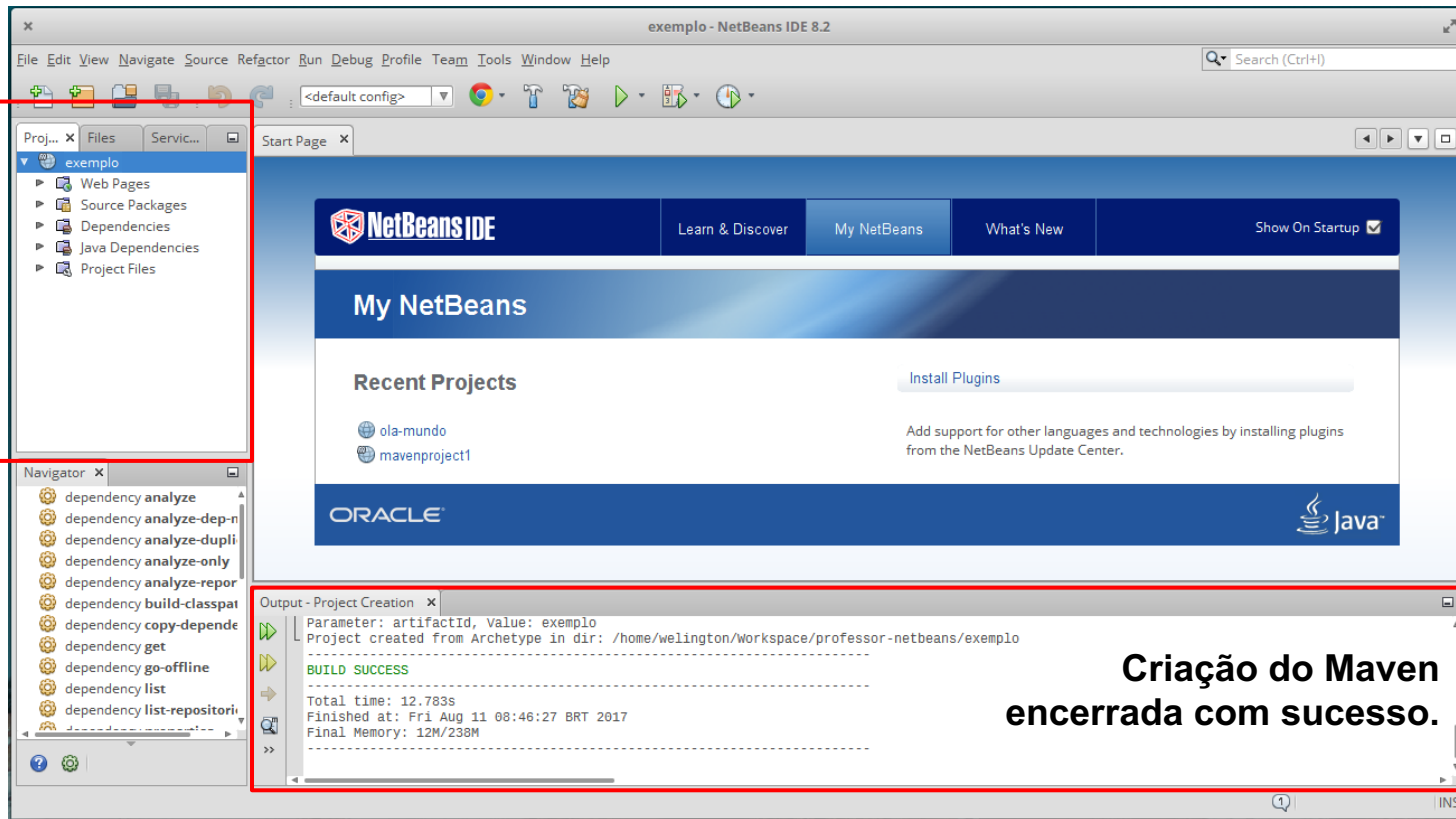
## ❏ Parâmetros do projeto



# | Prática | Disponibilizando uma página Web Estática no Glassfish

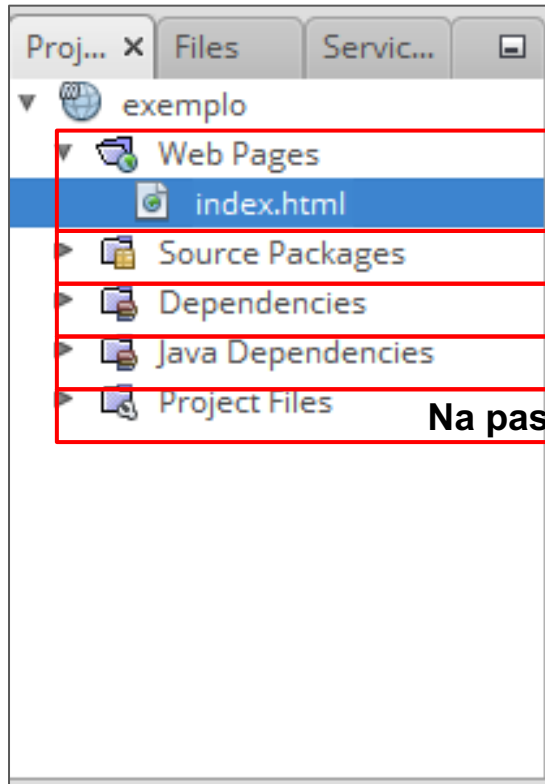
## ❏ Parâmetros do projeto

**Projeto criado em Projetos.**



# | Prática | Disponibilizando uma página Web Estática no Glassfish

📁 Navegando pelo Projeto



**Na pasta Web Pages disponibilizamos todos os recursos web, Como páginas, css, javascript, imagens, etc.**

**Na pasta Source Packages implementaremos nossas classes Java.**

**Na pasta dependencies teremos acesso às dependências do Maven**

**Na pasta Java dependencies teremos acesso às dependências do Java**

**Na pasta Project Files teremos acesso aos arquivos de configuração do projeto.**

# | Prática | Disponibilizando uma página Web Estática no Glassfish

- ❑ Rodando o projeto.

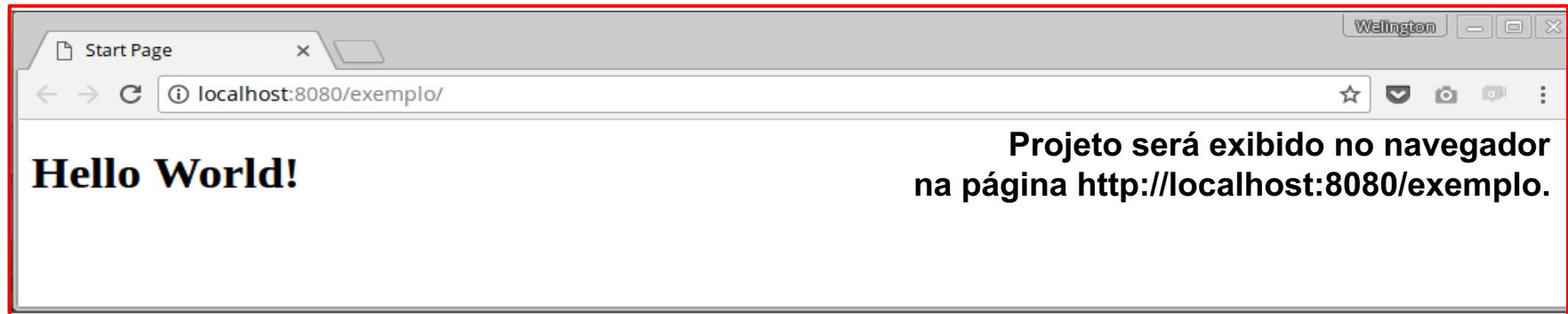


- ❑ Verificando a saída do glassfish e da execução do projeto pelo maven.

```
Output x
Run (exemplo) x  Java DB Database Process x  GlassFish Server 4.1.1 x
Info: visiting unvisited references
Info: visiting unvisited references
Info: Loading application [exemplo] at [/exemplo]
Info: exemplo was successfully deployed in 1,272 milliseconds.
Warning: Instance could not be initialized. Class=interface org.glassfish.grizzly.http.server.AddOn, name=http-listener-1, realClassName=org.gla
Info: Created HTTP listener http-listener-1 on host/port 0.0.0.0:8080
Info: Grizzly Framework 2.3.23 started in: 1ms - bound to [/0.0.0.0:8080]
```

**Endereço em que a aplicação foi disponibilizada. (/exemplo)**

**Porta em que a aplicação foi disponibilizada. (8080)**



**Projeto será exibido no navegador  
na página http://localhost:8080/exemplo.**

# | HTTP | Referências adicionais

1. <https://www.caelum.com.br/apostila-html-css-javascript/>
2. <https://developer.mozilla.org/pt-BR/docs/Web/HTML>
3. <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>
4. <https://developer.mozilla.org/pt-BR/docs/Web/CSS>