

FIA/P GRADUAÇÃO

SISTEMAS DE INFORMAÇÃO

MICROSERVICE AND WEB ENGINEERING

PROF. Dr. Emerson R. Abraham

Agenda

Aula anterior:

- Introdução Java EE e conceitos básicos WEB

Aula de Hoje:

- Tomcat, o que é?
- Configuração e execução
- Exercícios

Objetivo

- Entender o funcionamento do Apache Tomcat e ser capaz de configurar e realizar deploys de aplicações.

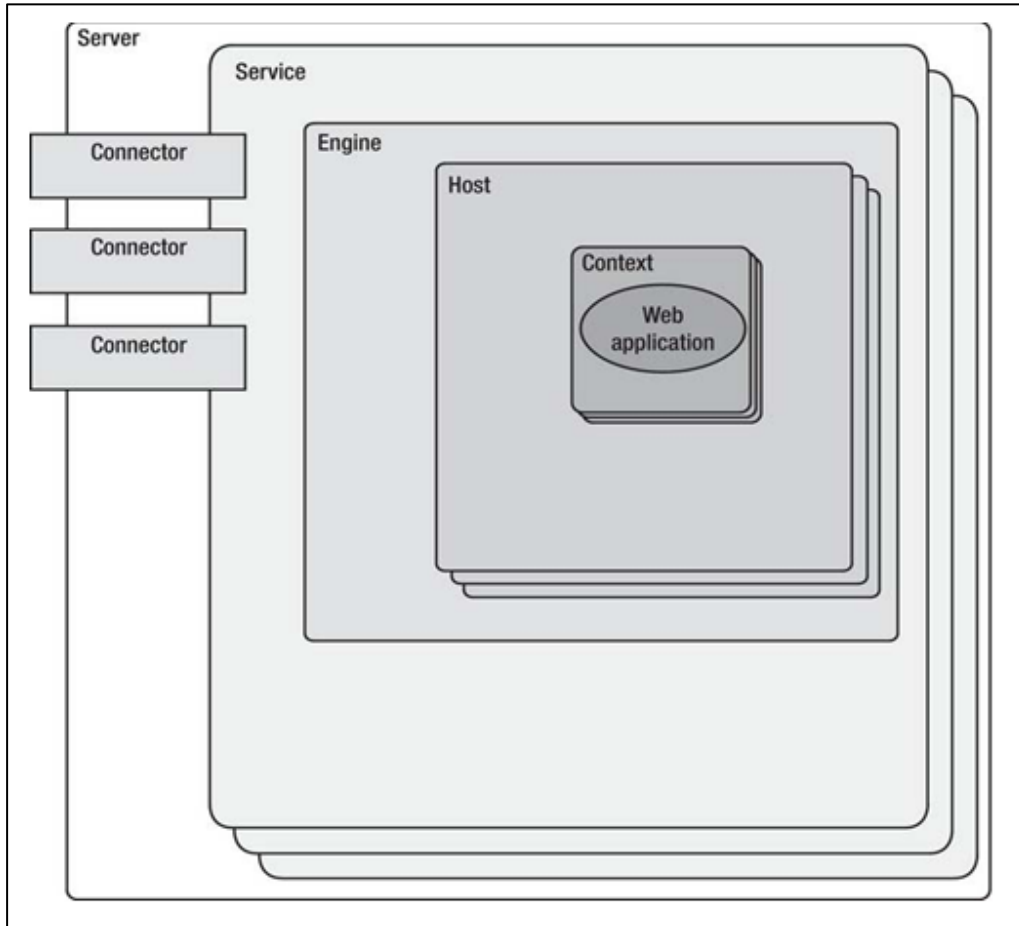


- O Apache Tomcat® é um **servidor de aplicações** Java para web
- O software é uma implementação de código aberto das tecnologias Java Servlet, JavaServer Pages, Java Expression Language e Java WebSocket.
- Middleware (container) que faz instância da servlet (Singleton)

Observações:

- Não compreende Java Standard Tag Library (JSTL), sendo necessário configuração da biblioteca .jar (taglibs)
- Não implementa Enterprise JavaBeans (EJB). Para aplicações que utilizam essa especificação, pode-se utilizar JBoss, GlassFish, Gerônimo, etc.
- Existe uma especificação denominada TomEE (Java Enterprise Edition do Tomcat). Tomcat com EJB adicionado e integrado, pronto para usar.

Apache Tomcat, o que é?

















- Servidor Web Java
- Criado para executar aplicações Web
- Executa Java Servlets e JavaServer Pages (JSP)



Apache Tomcat, o que é?

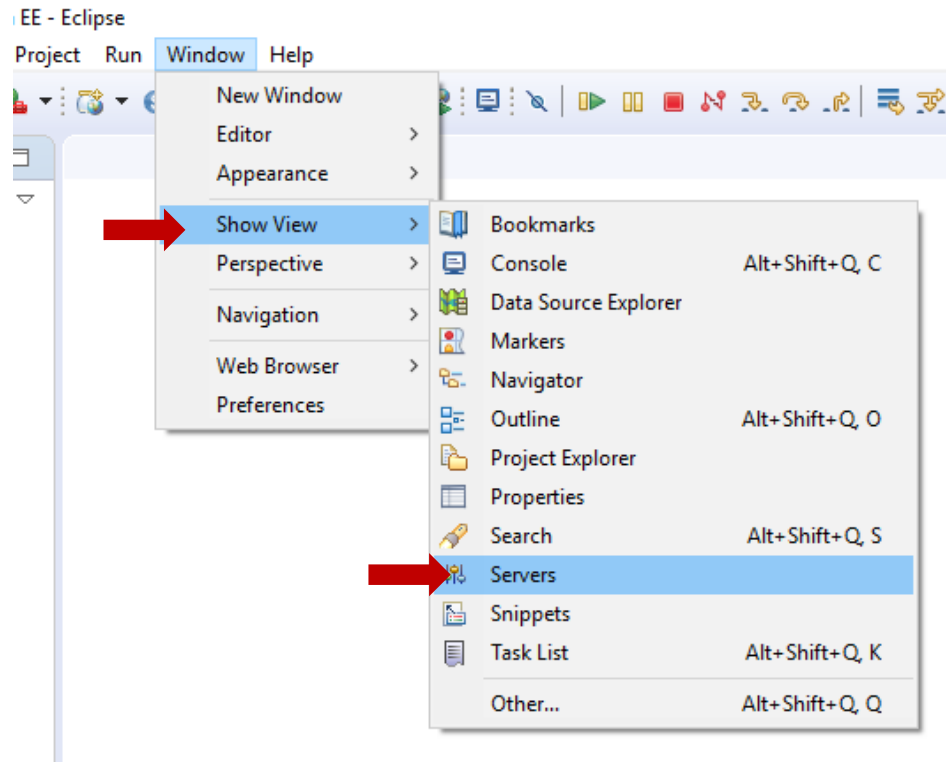
- **SERVER:** Representa todo o *Catalina Servlet Engine* e é usado como um elemento de nível mais alto para uma simples *instância do Tomcat*.
- **SERVICE:** Contém uma coleção de um ou mais *Connectors* que compartilham um única Engine.
- **CONNECTOR:** Define as classes que fazem o atual *gerenciamento das requisições e respostas* para uma chamada de uma aplicação cliente.
- **ENGINE:** *Gerencia todas as requisições* recebidas por todos os Connectors definidos.
- **HOST:** *Define os hosts virtuais* que são contidos em cada instância do Cataline Engine. Cada Host pode ser um pai de uma ou mais aplicações web.
- **CONTEXT:** É o elemento mais comumente usado numa instância do Tomcat. Cada elemento Context *representa uma aplicação web* individual que está rodando dentro de um <Host> definido.

Estrutura do Apache Tomcat

	bin	←	Arquivos para o funcionamento do Tomcat
	BUILDING.txt		
	conf	←	Configurações
	CONTRIBUTING.md		
	lib	←	Bibliotecas
	LICENSE		
	logs	←	Arquivos de logs
	NOTICE		
	README.md		
	RELEASE-NOTES		
	RUNNING.txt		
	temp	←	Arquivos temporários
	webapps	←	Arquivos fonte da aplicação
	work	←	Arquivos necessários durante o tempo de execução

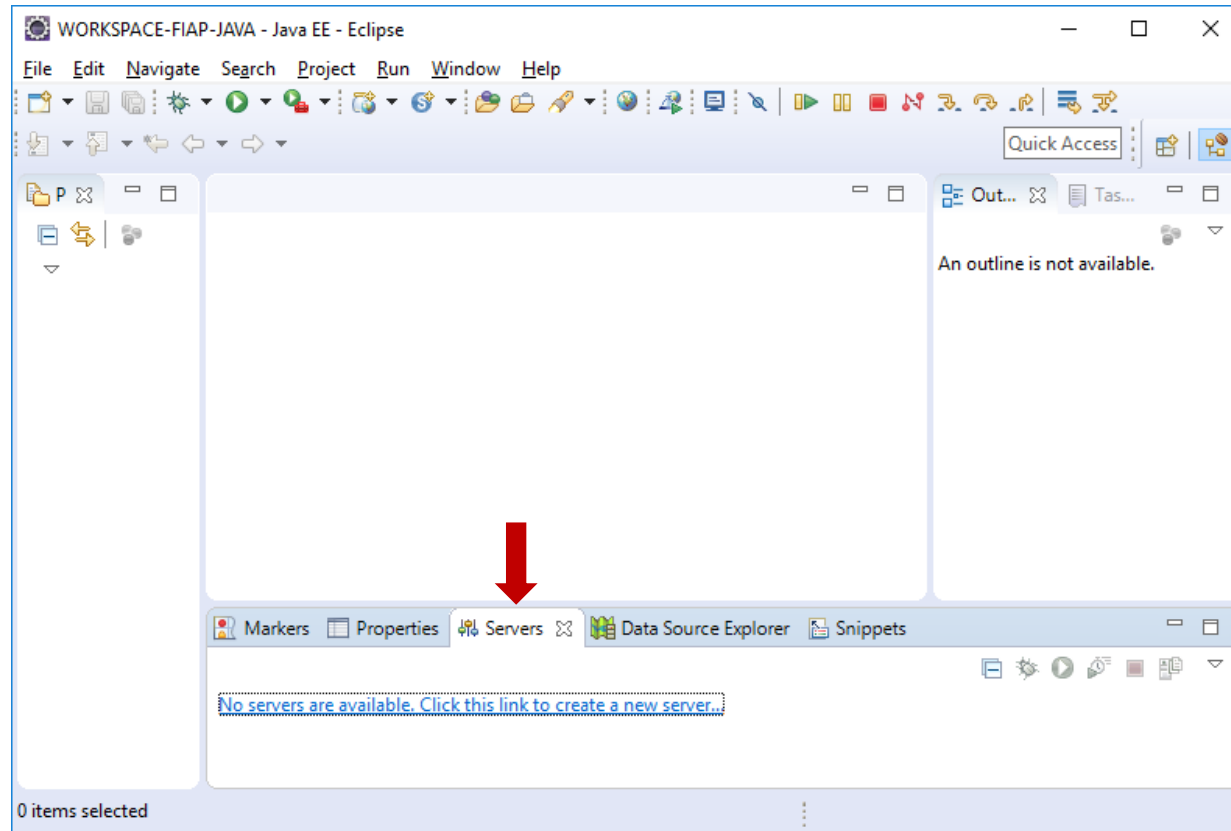
Configuração do Eclipse + Tomcat

- Deixando visível a aba de servidores



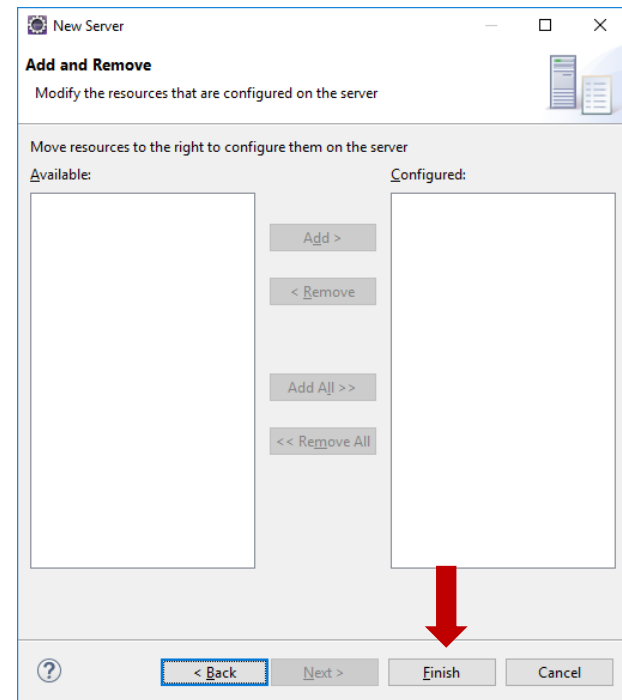
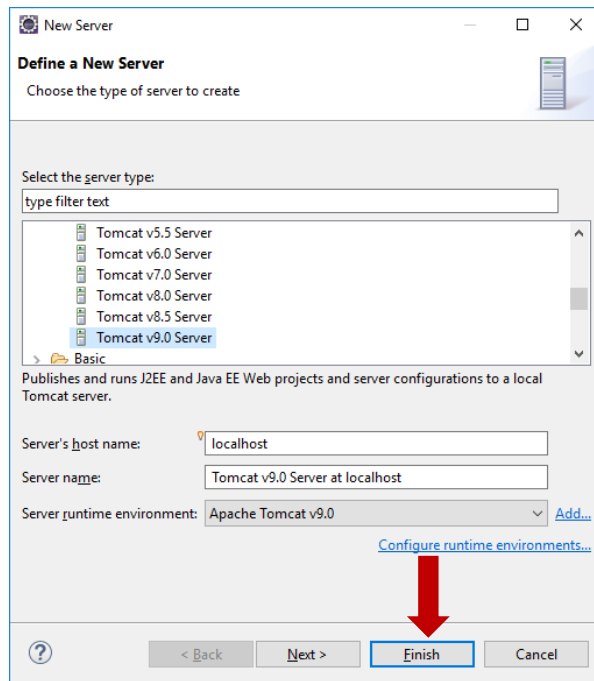
Configuração do Eclipse + Tomcat

- Adicionando o Tomcat



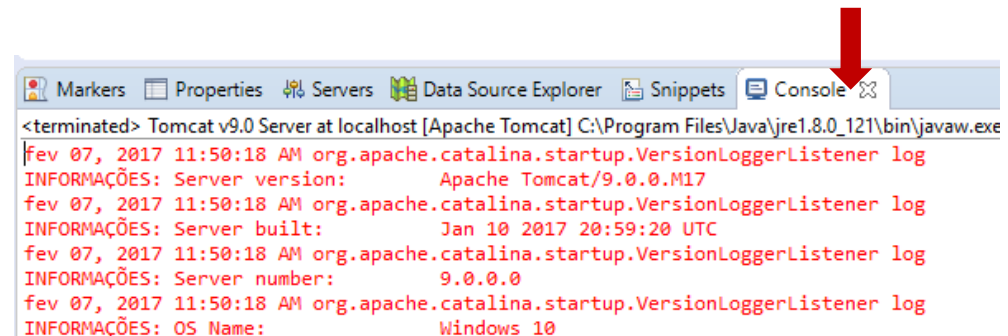
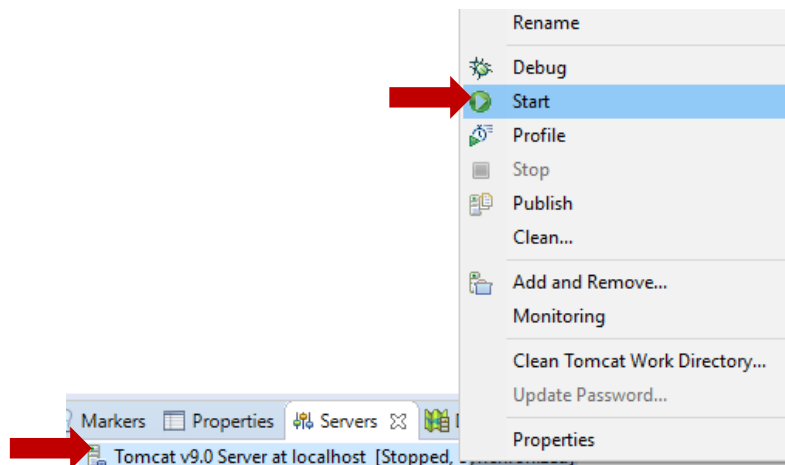
Configuração do Eclipse + Tomcat

- Adicionando o Tomcat
 - Selecione a pasta Apache e o Servidor V9.0



Configuração do Eclipse + Tomcat

- Iniciando o servidor e consultando o log





Exercício

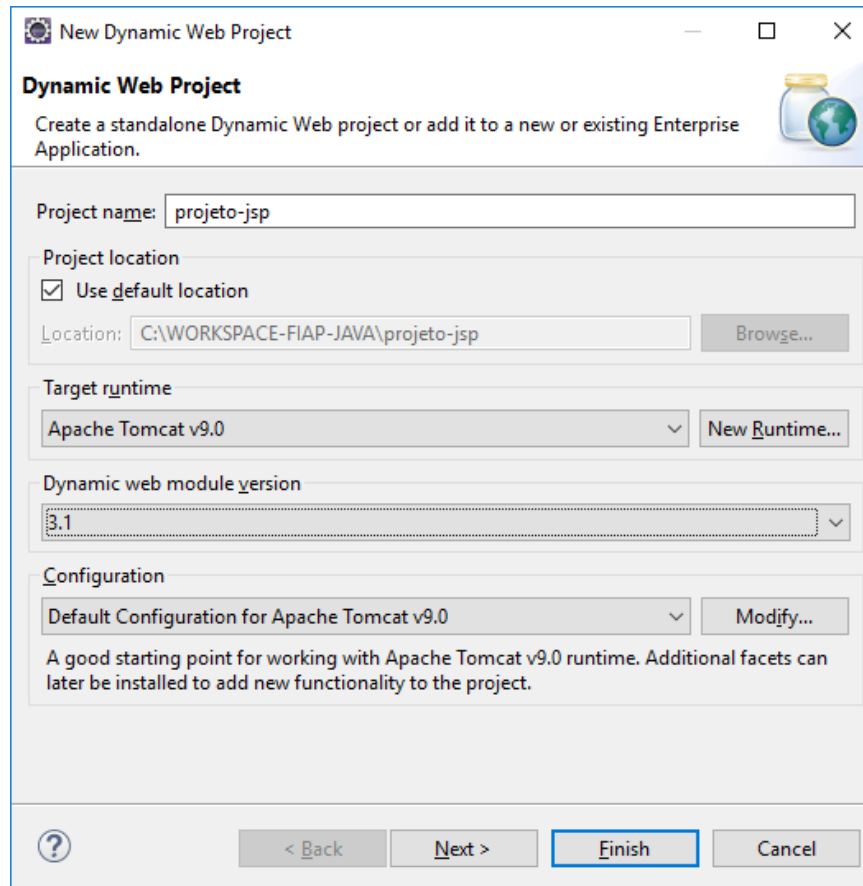
Criação da Aplicação Web

Criação Aplicação Web

- **Criando as aplicações Web - Dinâmica**
 - Criar um aplicativo web (Dynamic Web Project)
nome: **projeto-jsp**
 - Em cada projeto criar um arquivo **index.jsp** com o conteúdo que irá identificar o projeto.
 - Adicionar o projeto no servidor de desenvolvimento (Apache Tomcat).
 - Executar o servidor e os projetos para validar a construção.

Criação da aplicação web

- Criando o Projeto
 - Menu File > New > Dynamic Web Project



Criação da aplicação web

- Criando o Projeto
 - Menu File > New > Dynamic Web Project

New Dynamic Web Project

Dynamic Web Project
Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.

Project name:

Project location
☒ Use default location
Location:

Target runtime

Dynamic web module version

Configuration

A good starting point for working with Apache Tomcat v9.0 runtime. Additional facets can later be installed to add new functionality to the project.

New Dynamic Web Project

Web Module
Configure web module settings.

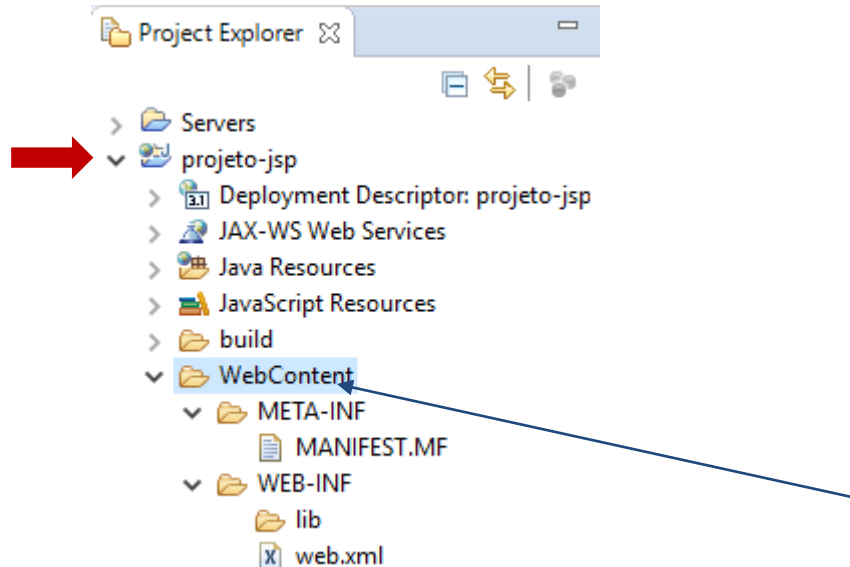
Context root:

Content directory:

☒ Generate web.xml deployment descriptor

Criação da aplicação web

- Projeto

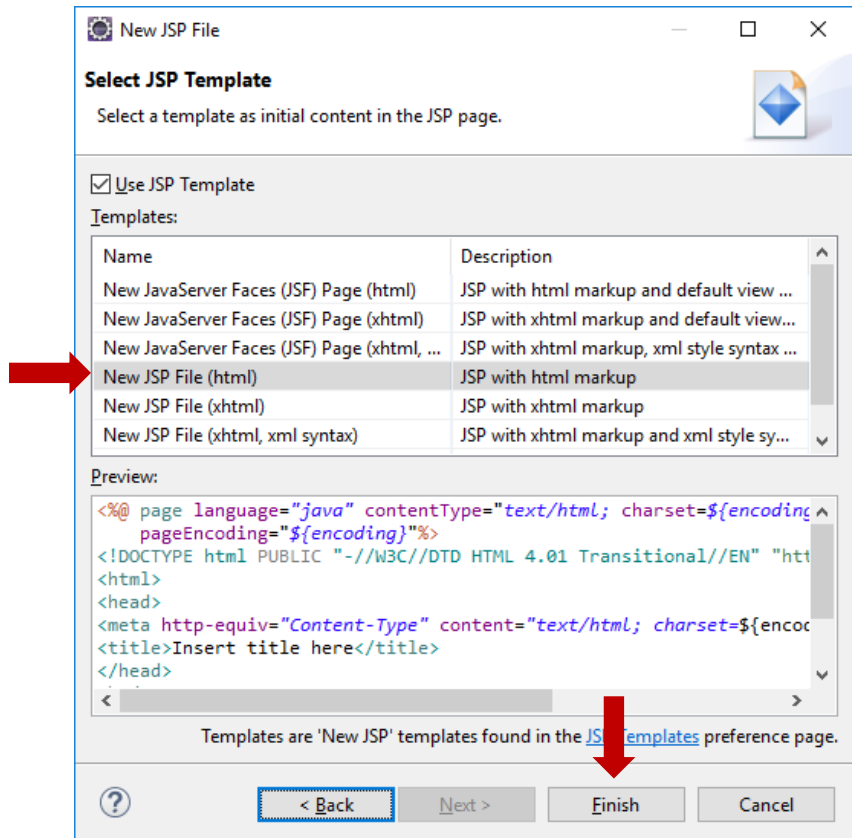
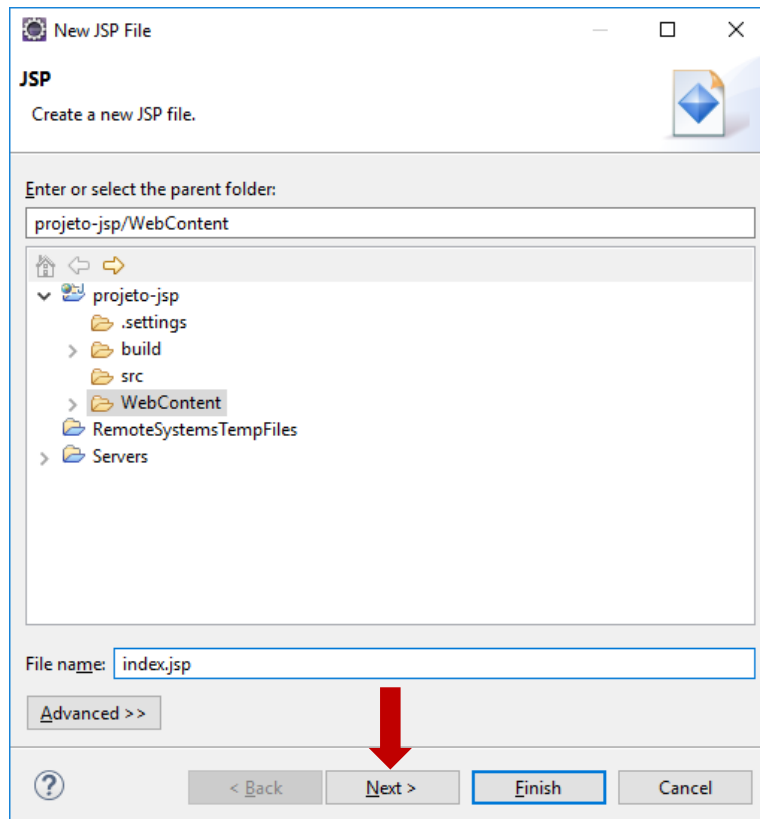


A pasta **WebContent** é responsável por armazenar o conteúdo visível (views) da aplicação web.

Exemplos: imagens, css, js, html e também **JSP**.

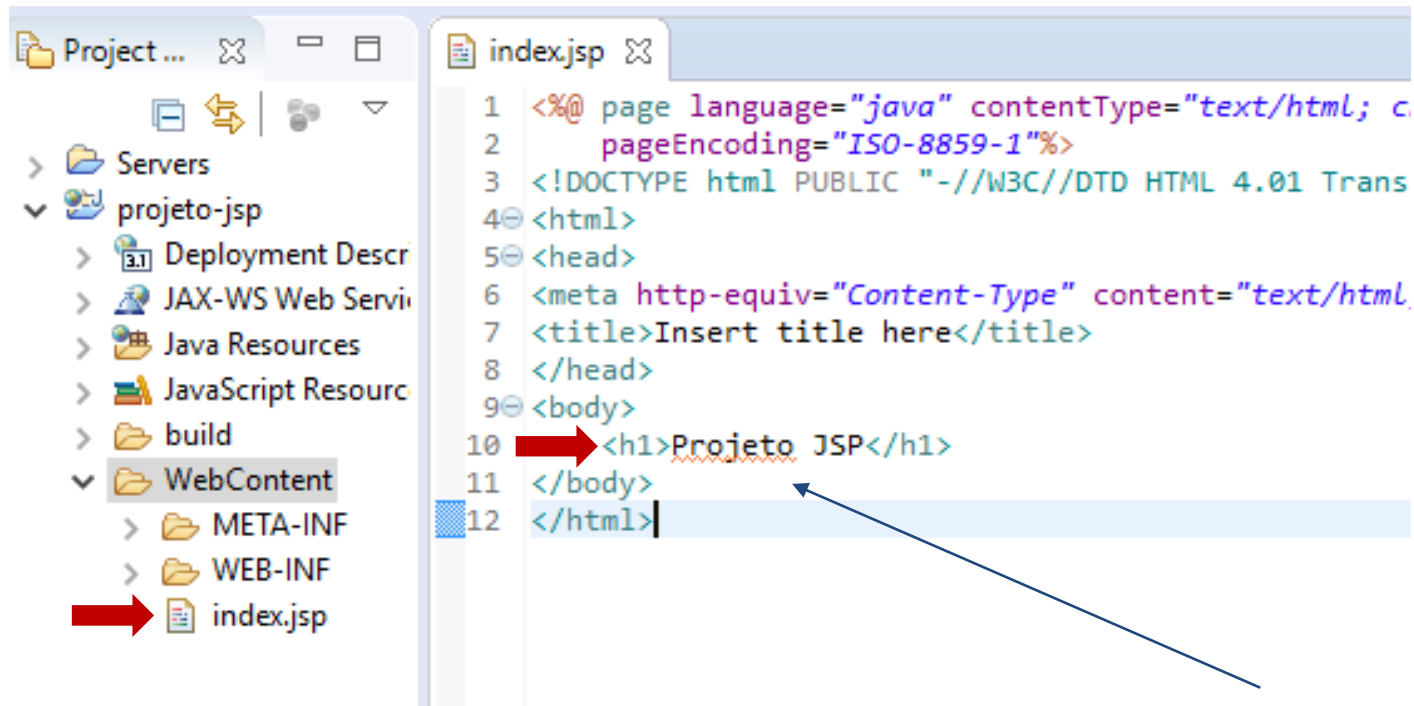
Criação da aplicação web

- Criando as páginas JSP
 - botão direito na pasta WebContent > New > JSP File



Criação da aplicação web

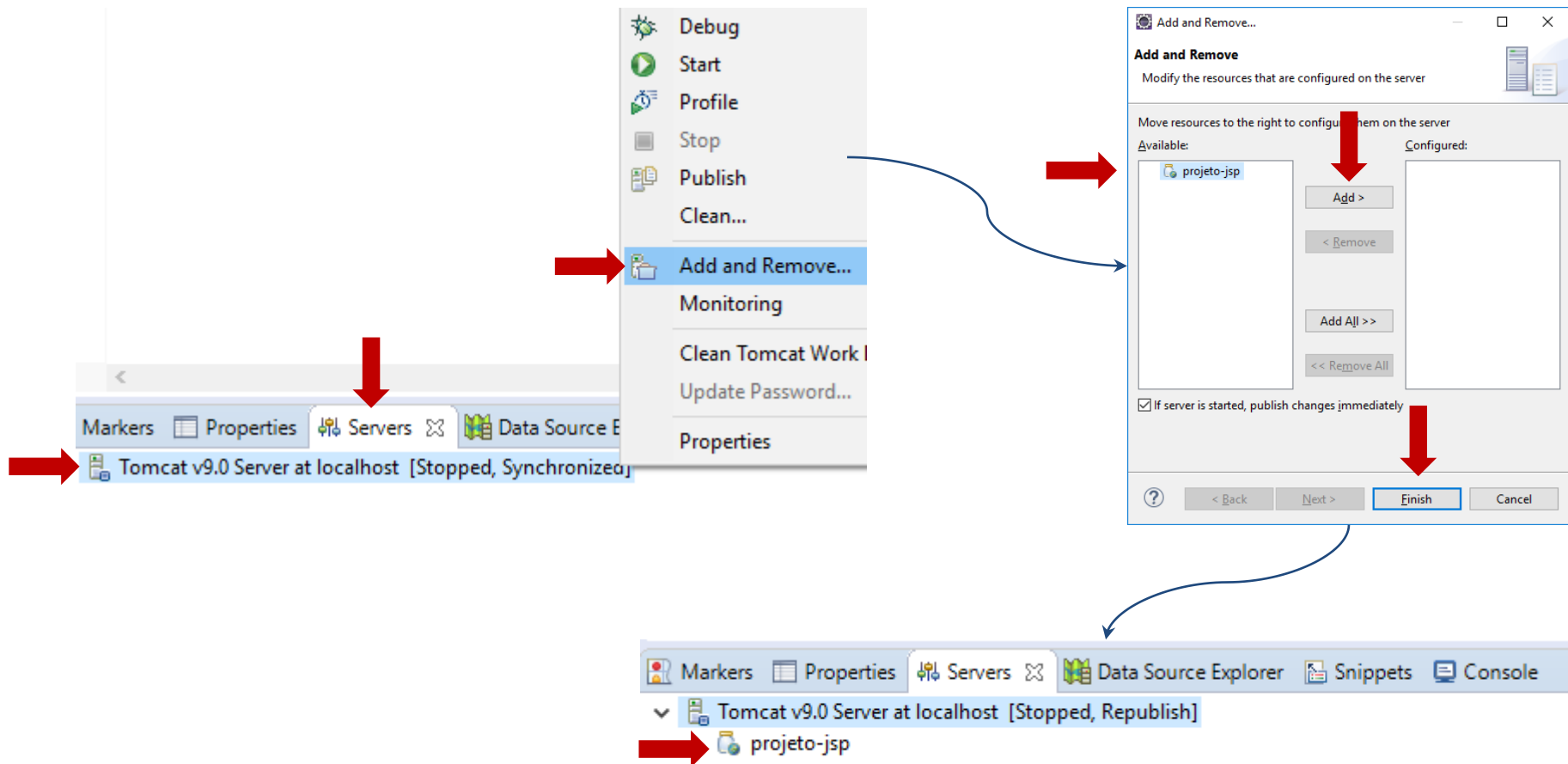
- Criando as páginas JSP



o arquivo **index.jsp** será carregado como padrão da aplicação. Assim vamos inserir um texto para identificar que essa é a primeira página do projeto-jsp.

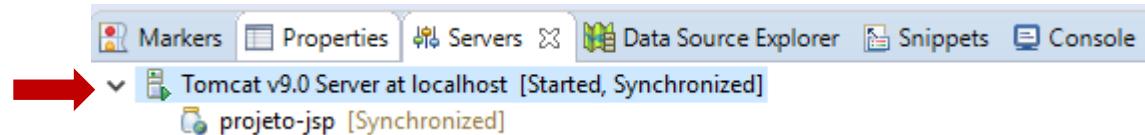
Criação da aplicação web

- Adicionando o projeto ao servidor



Criação da aplicação web

- Executando o servidor
 - Botão direito > Start



- Executando a aplicação
 - `http://localhost:8080/projeto-jsp`





Referências

Apache Tomcat disponível em: <http://tomcat.apache.org/>

Profº. Pedro Ivo Correia

Profº. Lucas Furlaneto

Copyright © 2021

Todos direitos reservados. Reprodução ou divulgação total / parcial deste documento é proibido sem o consentimento formal.