

Apache Maven

Apache Maven, ou simplesmente Maven, é uma ferramenta de automação de compilação utilizada primariamente em projetos Java.

O Maven utiliza um arquivo XML (POM) para descrever o projeto de software sendo construído, suas dependências sobre módulos e componentes externos, a ordem de compilação, diretórios e plug-ins necessários. Ele vem com objetivos pré-definidos para realizar certas tarefas bem definidas como compilação de código e seu empacotamento.

O Maven baixa bibliotecas Java e seus plug-ins dinamicamente de um ou mais repositórios, como o Maven 2 Central Repository, e armazena-os em uma área de cache local.^[2] Este cache local de artefatos baixados pode também ser atualizado com artefatos criados por projetos locais. Repositórios públicos podem também ser atualizados.

O Maven é construído utilizando uma arquitetura baseada em plugin, que permite que ele faça uso de qualquer aplicação controlável através da entrada padrão. Teoricamente, isto permitiria qualquer um escrever plugins para fazer interface com ferramentas de construção (compiladores, ferramentas de teste de unidade, etc.) para qualquer outra linguagem.

Projetos Maven são configurados usando um Project Object Model, que é armazenado em um arquivo `pom.xml`. A seguir está um exemplo mínimo:

```
<project>
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.mycompany.app</groupId>
  <artifactId>my-app</artifactId>
  <version>1.0</version>

  <dependencies>

    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>

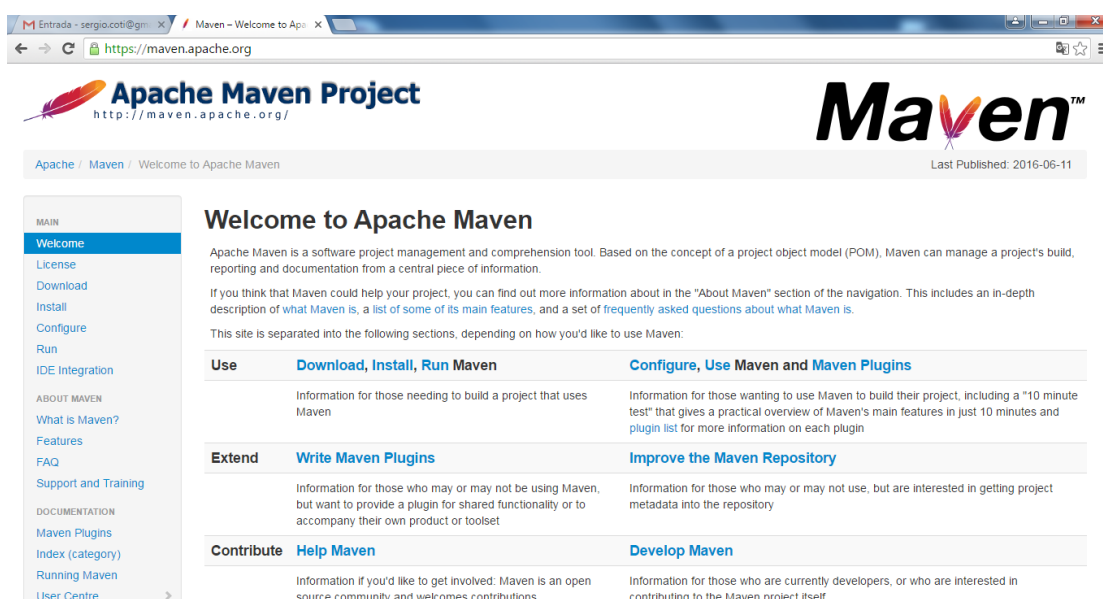
  </dependencies>

</project>
```

Este POM apenas define um identificador único para o projeto (*coordenadas*) e sua dependência do framework JUnit. No entanto, isto já é suficiente para a construção do projeto e execução dos testes de unidade associados ao projeto. O Maven realiza isto adotando a ideia de Convenção sobre configuração, isto é, ele fornece valores padrões para a configuração do projeto

Fazendo download do Maven:

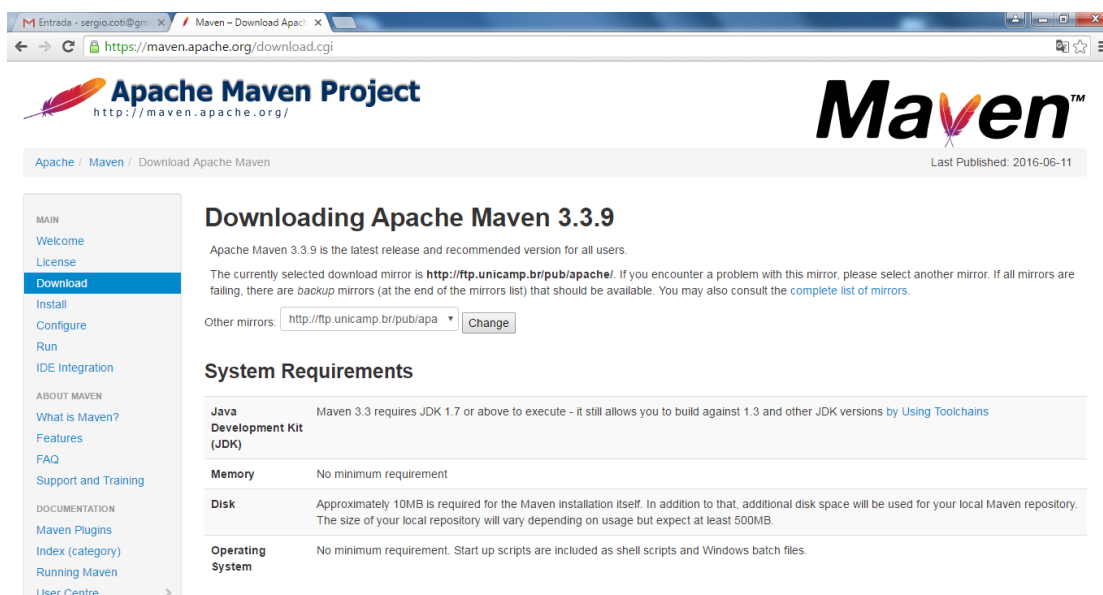
<https://maven.apache.org/>



The screenshot shows the Apache Maven Project homepage. The header includes the Apache Maven Project logo and the text "Maven™". The main content area is titled "Welcome to Apache Maven" and contains a table with links to various sections. The table has three columns: "Use", "Download, Install, Run Maven", and "Configure, Use Maven and Maven Plugins". The "Use" column contains links for "Write Maven Plugins" and "Improve the Maven Repository". The "Download, Install, Run Maven" column contains links for "Download, Install, Run Maven" and "Configure, Use Maven and Maven Plugins". The "Configure, Use Maven and Maven Plugins" column contains links for "Write Maven Plugins" and "Improve the Maven Repository".

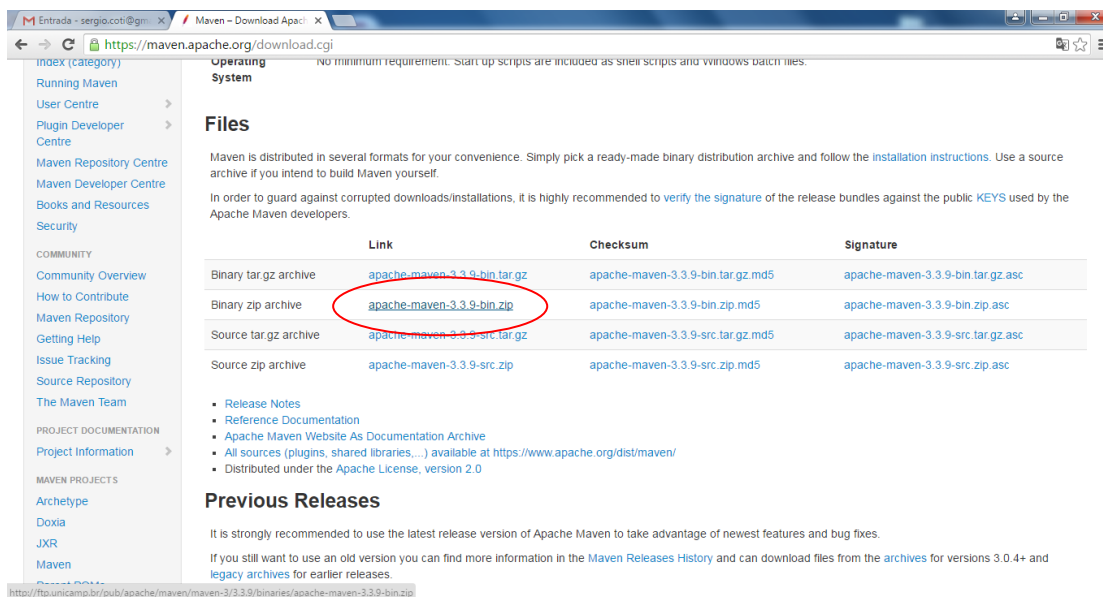
Use	Download, Install, Run Maven	Configure, Use Maven and Maven Plugins
	Information for those needing to build a project that uses Maven	Information for those wanting to use Maven to build their project, including a "10 minute test" that gives a practical overview of Maven's main features in just 10 minutes and plugin list for more information on each plugin
Extend	Write Maven Plugins	Improve the Maven Repository
	Information for those who may or may not be using Maven, but want to provide a plugin for shared functionality or to accompany their own product or toolset	Information for those who may or may not use, but are interested in getting project metadata into the repository
Contribute	Help Maven	Develop Maven
	Information if you'd like to get involved: Maven is an open source community and welcomes contributions.	Information for those who are currently developers, or who are interested in contributing to the Maven project itself

<https://maven.apache.org/download.cgi>

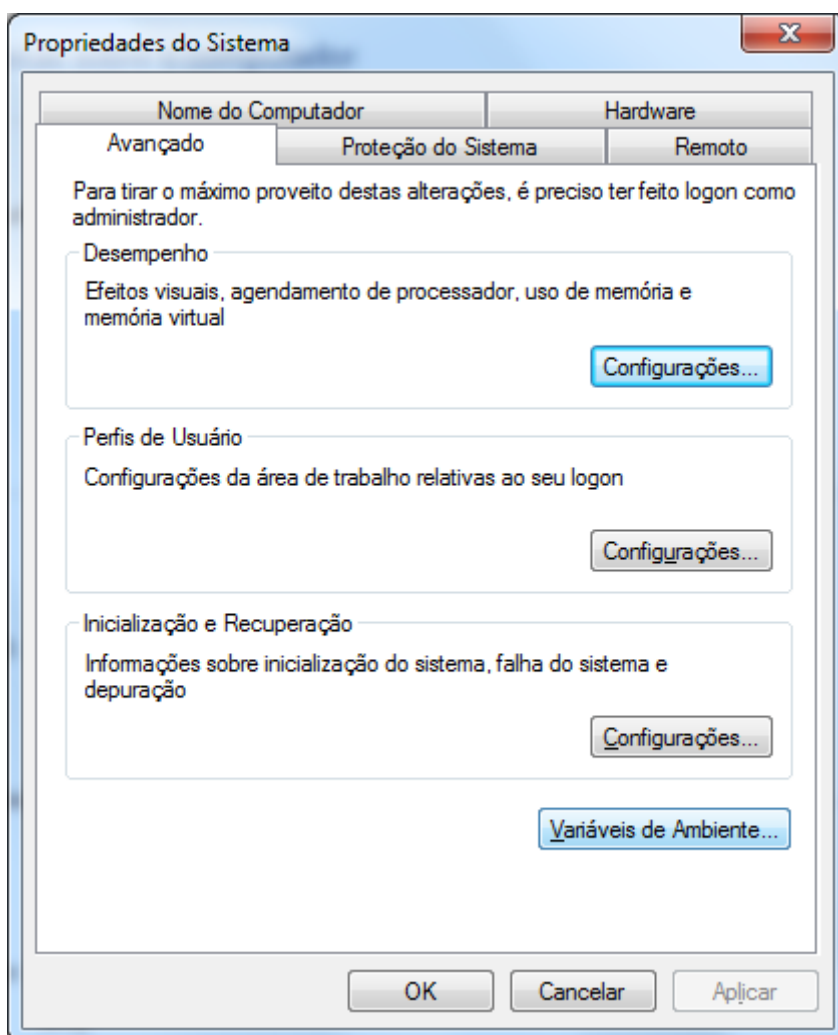


The screenshot shows the Apache Maven Project download page. The header includes the Apache Maven Project logo and the text "Maven™". The main content area is titled "Downloading Apache Maven 3.3.9" and contains a table with system requirements. The table has two columns: "System Requirements" and "Details". The "System Requirements" column contains links for "Java Development Kit (JDK)", "Memory", "Disk", and "Operating System". The "Details" column contains information about the requirements for each system.

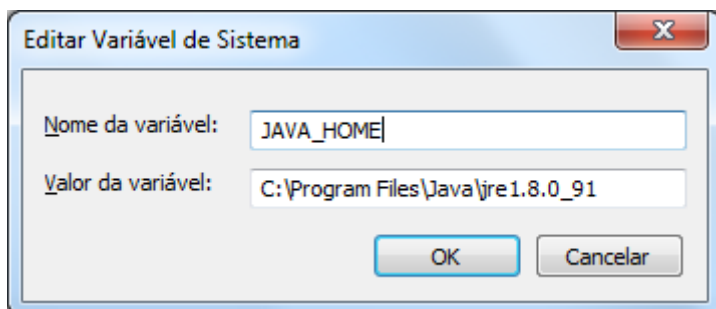
System Requirements	Details
Java Development Kit (JDK)	Maven 3.3 requires JDK 1.7 or above to execute - it still allows you to build against 1.3 and other JDK versions by Using Toolchains
Memory	No minimum requirement
Disk	Approximately 10MB is required for the Maven installation itself. In addition to that, additional disk space will be used for your local Maven repository. The size of your local repository will vary depending on usage but expect at least 500MB.
Operating System	No minimum requirement. Start up scripts are included as shell scripts and Windows batch files.



Configurando as variáveis de ambiente:

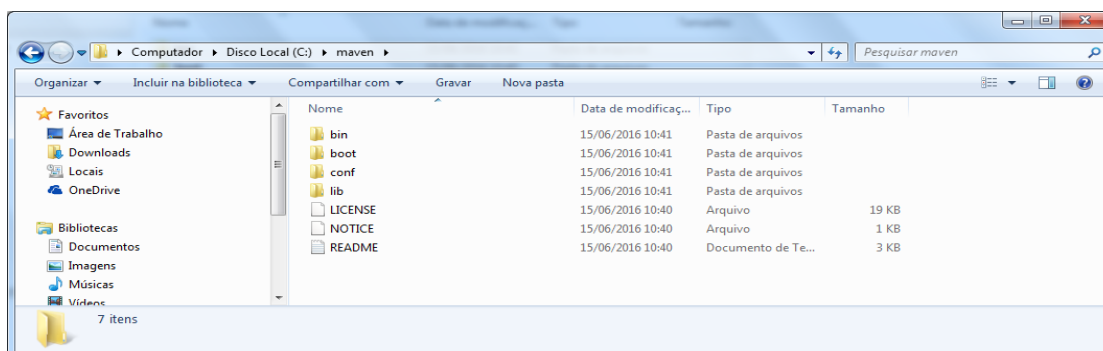


Definindo a variável JAVA_HOME

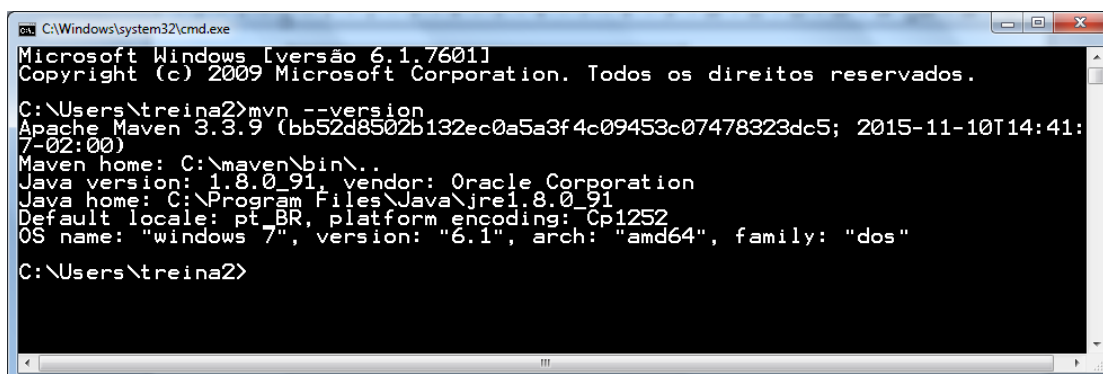
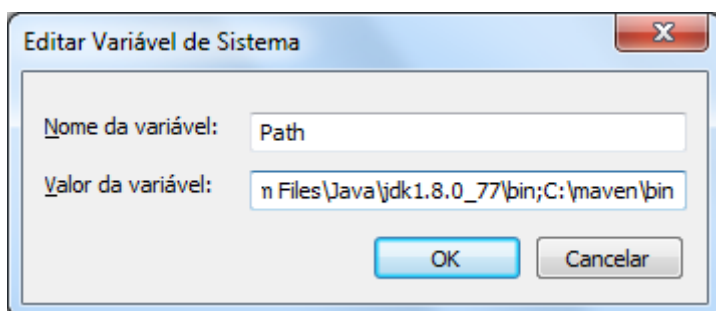


Crie um diretorio **c:\maven**

E extraia para este diretorio o conteudo do MAVEN



Criando a variável de ambiente para o MAVEN:



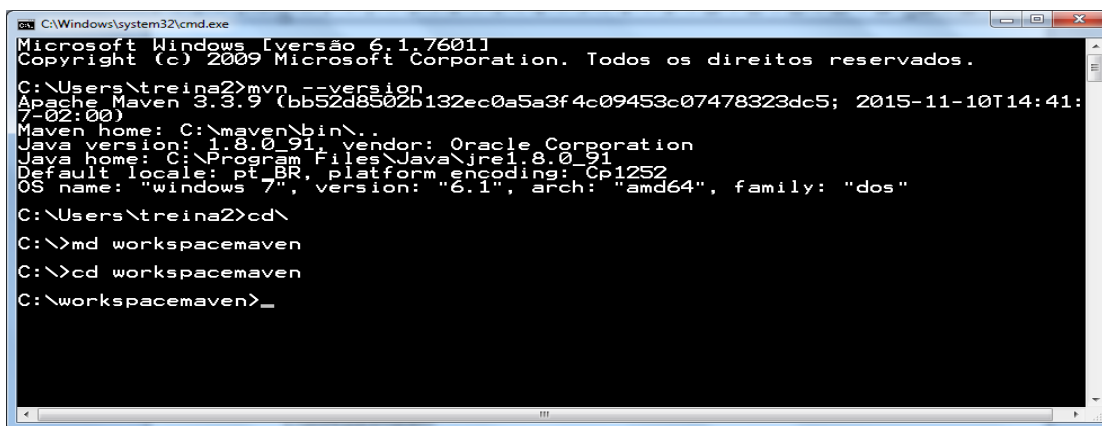
Crie o workspace:

```
C:\Users\treina2>cd\
```

```
C:\>md workspacemaven
```

```
C:\>cd workspacemaven
```

```
C:\workspacemaven>
```



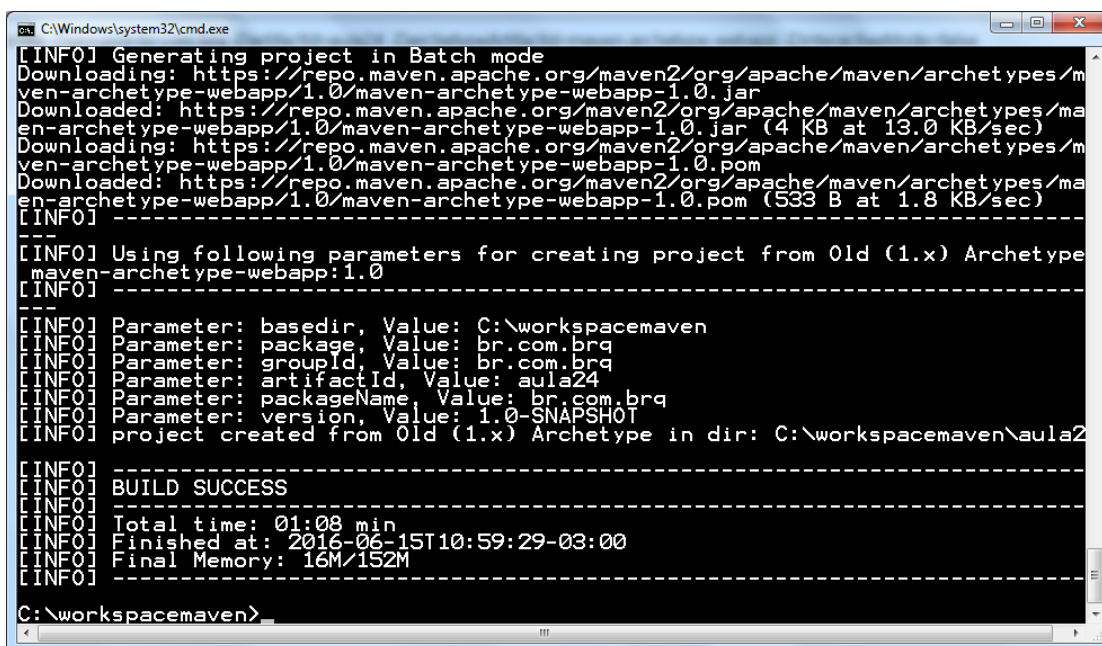
```
Microsoft Windows [Versão 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Users\treina2>mvn --version
Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-10T14:41:
7-02:00)
Maven home: C:\maven\bin\..
Java version: 1.8.0_91, vendor: Oracle Corporation
Java home: C:\Program Files\Java\jre1.8.0_91
Default locale: pt_BR, platform.encoding: Cp1252
OS name: "windows 7", version: "6.1", arch: "amd64", family: "dos"

C:\Users\treina2>cd\
C:\>md workspacemaven
C:\>cd workspacemaven
C:\workspacemaven>
```

Criando o projeto:

```
mvn archetype:generate -DgroupId=br.com.brq -DartifactId=aula24 -
DarchetypeArtifactId=maven-archetype-webapp -DinteractiveMode=false
```

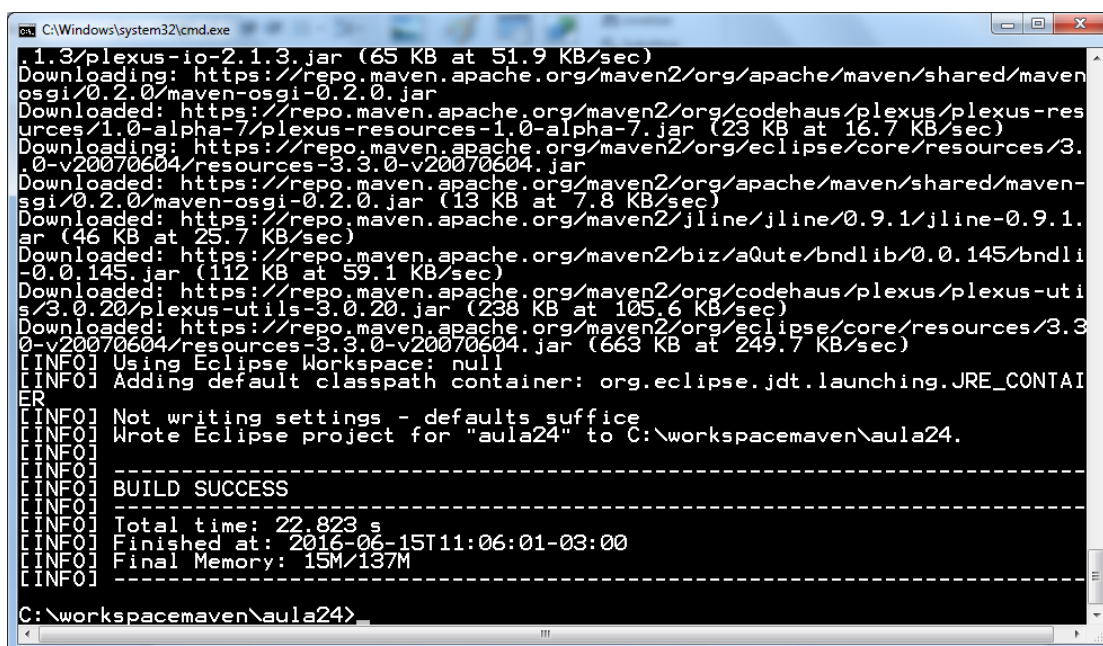


```
[INFO] Generating project in Batch mode
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/m
ven-archetype-webapp/1.0/maven-archetype-webapp-1.0.jar
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/m
ven-archetype-webapp/1.0/maven-archetype-webapp-1.0.jar (4 KB at 13.0 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/m
ven-archetype-webapp/1.0/maven-archetype-webapp-1.0.pom
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/m
ven-archetype-webapp/1.0/maven-archetype-webapp-1.0.pom (533 B at 1.8 KB/sec)
[INFO] -----
[INFO] Using following parameters for creating project from Old (1.x) Archetype
maven-archetype-webapp:1.0
[INFO] -----
[INFO] Parameter: basedir, Value: C:\workspacemaven
[INFO] Parameter: package, Value: br.com.brq
[INFO] Parameter: groupId, Value: br.com.brq
[INFO] Parameter: artifactId, Value: aula24
[INFO] Parameter: packageName, Value: br.com.brq
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: C:\workspacemaven\aula24
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:08 min
[INFO] Finished at: 2016-06-15T10:59:29-03:00
[INFO] Final Memory: 16M/152M
[INFO] -----
C:\workspacemaven>
```

Acessando a pasta do projeto e atualizando as dependencias:

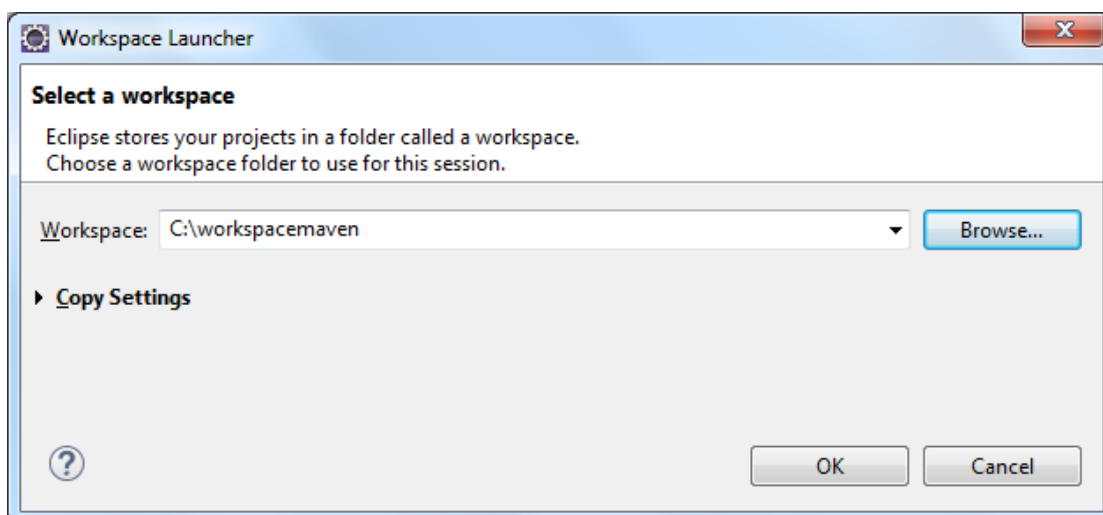
em `c:\workspacemaven\aula24`

`mvn eclipse:eclipse -Dwtversion=2.0`

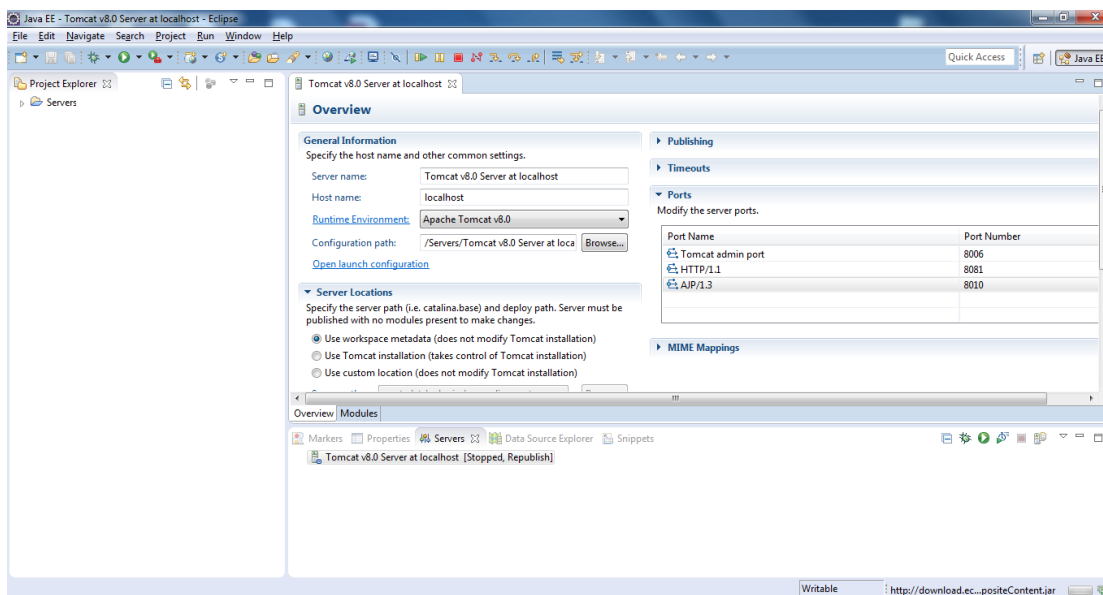


```
C:\Windows\system32\cmd.exe
.1.3/plexus-io-2.1.3.jar (65 KB at 51.9 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-
osgi/0.2.0/maven-osgi-0.2.0.jar
Downloaded: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-res
ources/1.0-alpha-7/plexus-resources-1.0-alpha-7.jar (23 KB at 16.7 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/eclipse/core/resources/3.
0-v20070604/resources-3.3.0-v20070604.jar
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-
osgi/0.2.0/maven-osgi-0.2.0.jar (13 KB at 7.8 KB/sec)
Downloaded: https://repo.maven.apache.org/maven2/jline/0.9.1/jline-0.9.1.
jar (46 KB at 25.7 KB/sec)
Downloaded: https://repo.maven.apache.org/maven2/biz/aQute/bndlib/0.0.145/bndli
-0.0.145.jar (112 KB at 59.1 KB/sec)
Downloaded: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-uti
s/3.0.20/plexus-utils-3.0.20.jar (238 KB at 105.6 KB/sec)
Downloaded: https://repo.maven.apache.org/maven2/org/eclipse/core/resources/3.3
0-v20070604/resources-3.3.0-v20070604.jar (663 KB at 249.7 KB/sec)
[INFO] Using Eclipse Workspace: null
[INFO] Adding default classpath container: org.eclipse.jdt.launching.JRE_CONTAI
ER
[INFO] Not writing settings - defaults suffice
[INFO] Wrote Eclipse project for "aula24" to C:\workspacemaven\aula24.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 22.823 s
[INFO] Finished at: 2016-06-15T11:06:01-03:00
[INFO] Final Memory: 15M/137M
[INFO] -----
C:\workspacemaven\aula24>
```

Abrindo o eclipse no workspace criado:

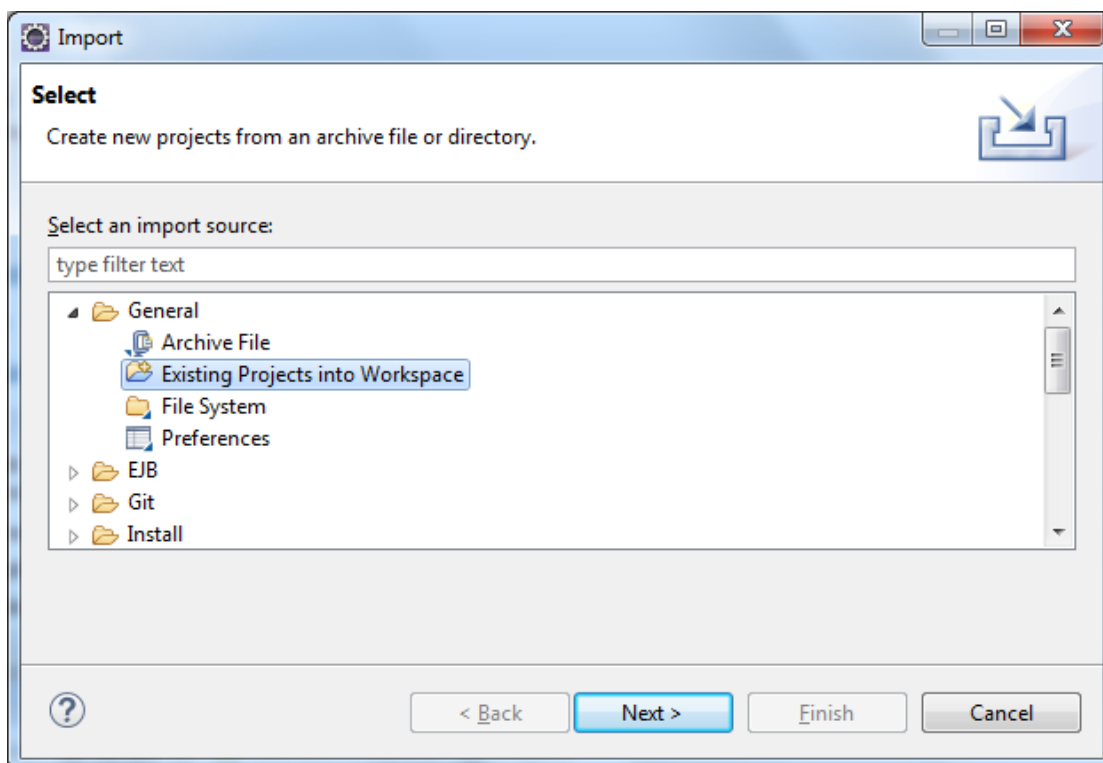


Configurando o servidor Tomcat:

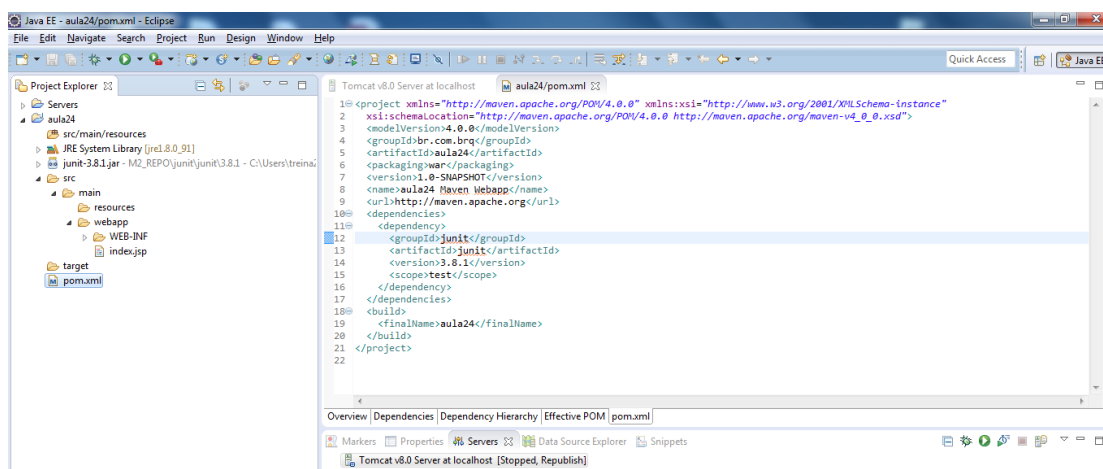
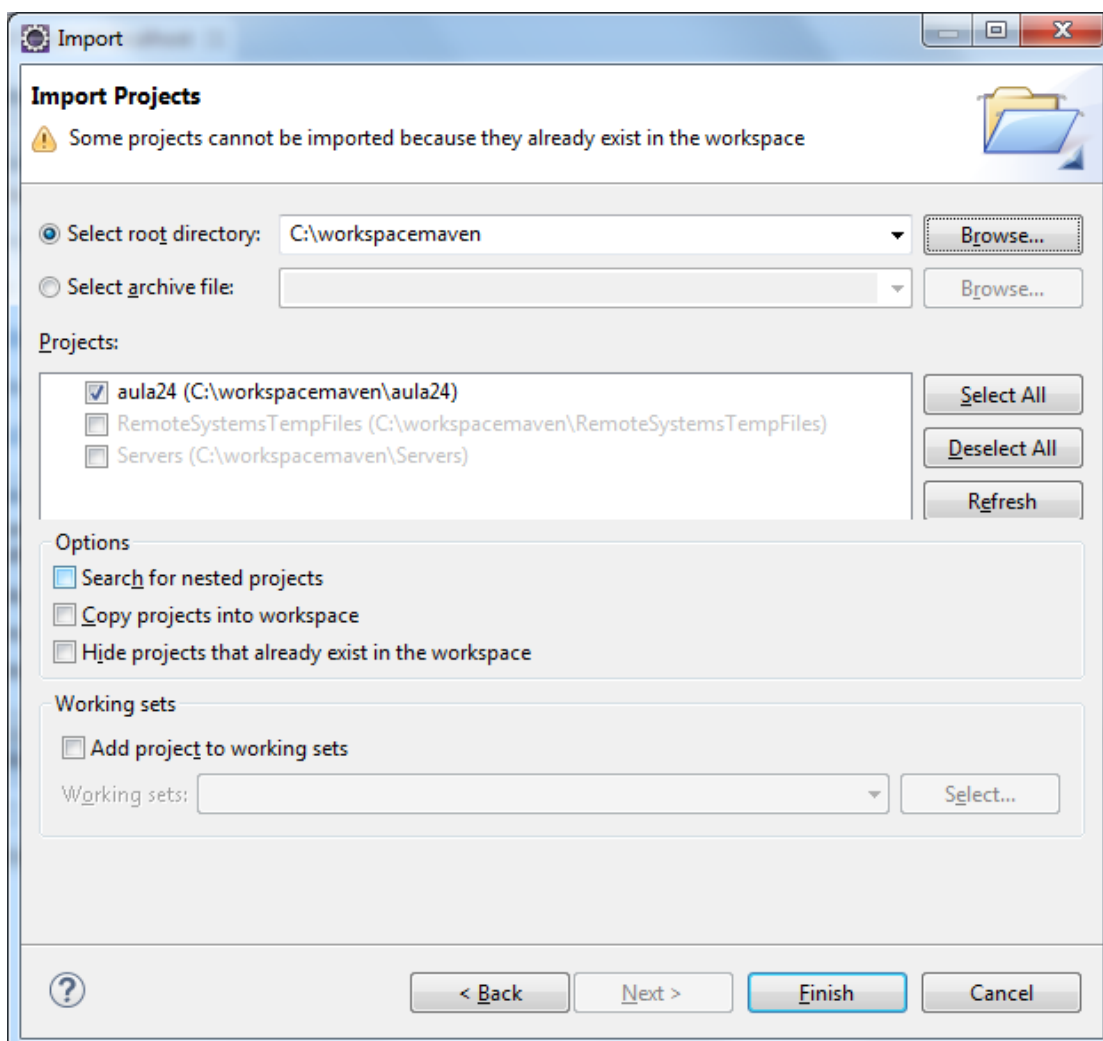


Abrindo o projeto no Eclipse:

- File > Import > Existing Project Into Workspace

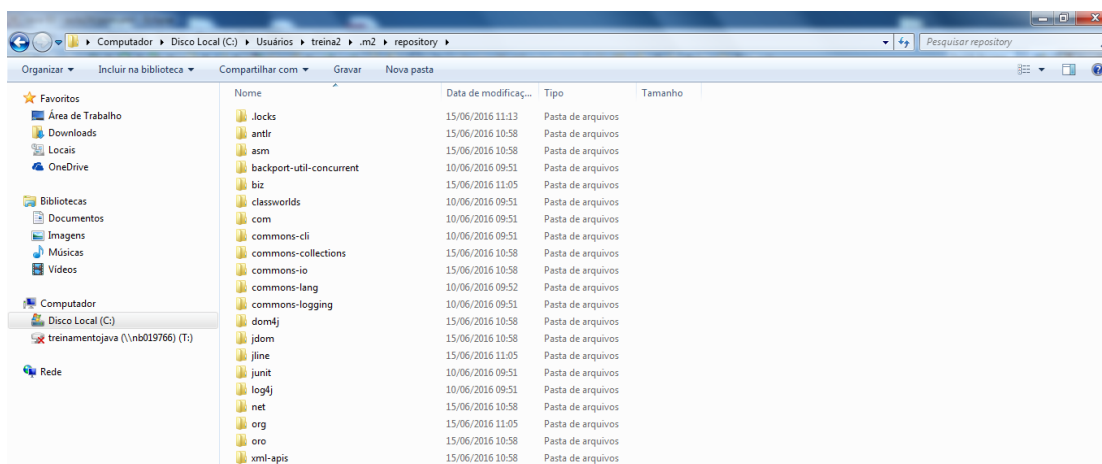


Selecione o projeto criado:

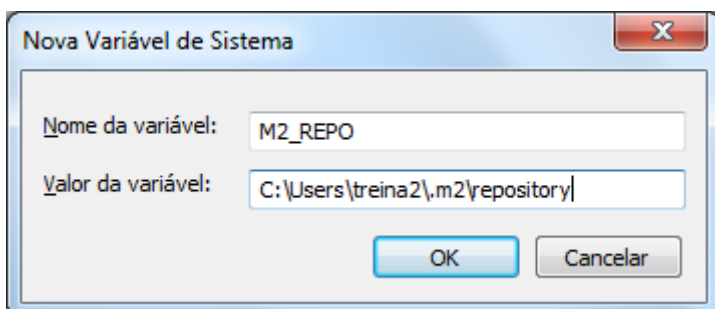


C:\Users\treina2\.m2\repository

Este é o diretório para onde o MAVEN faz o download das dependencias definidas pelo arquivo POM.XML do projeto:

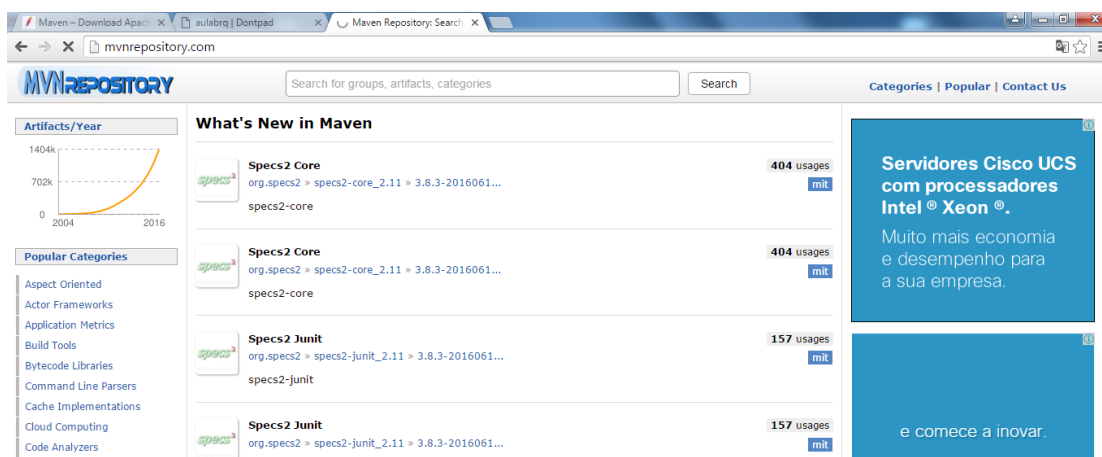


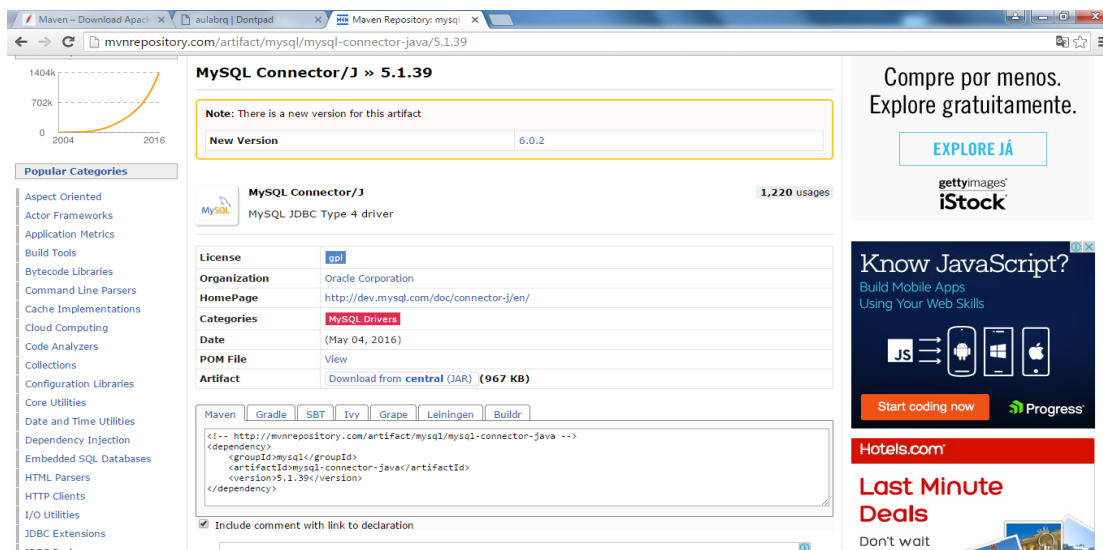
Criando uma variavel de ambiente para o caminho do repositorio do MAVEN:



Obtendo dependencias para o projeto:

<http://mvnrepository.com/>





Obtendo o MySQL Connector:

```
<!-- http://mvnrepository.com/artifact/mysql/mysql-connector-java -->
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>5.1.39</version>
</dependency>
```

Editando o arquivo pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>br.com.brq</groupId>
  <artifactId>aula24</artifactId>
  <packaging>war</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>aula24 Maven Webapp</name>
  <url>http://maven.apache.org</url>
  <dependencies>

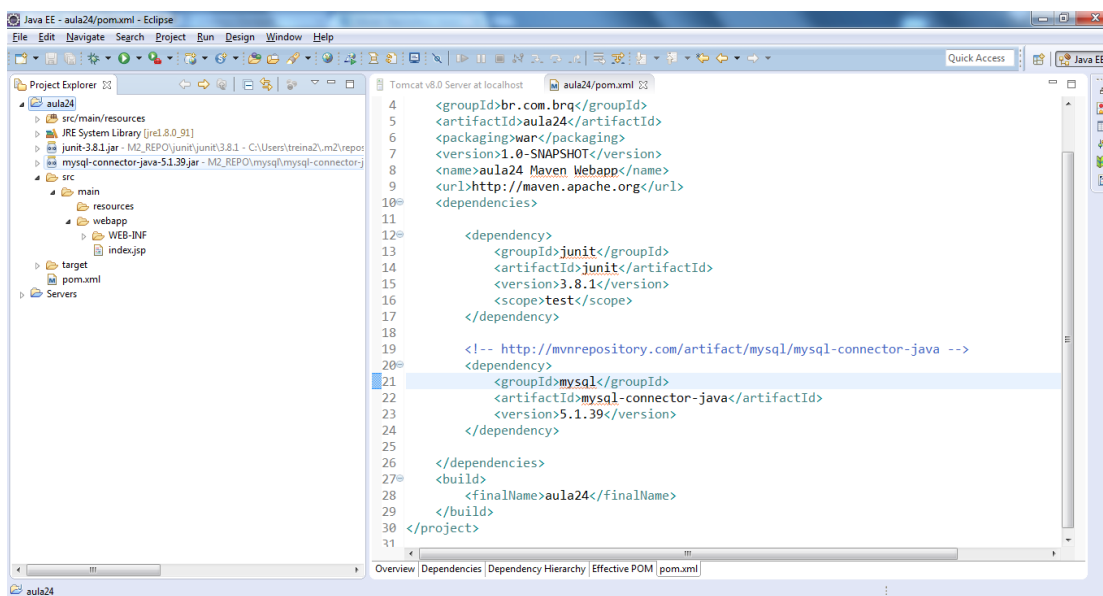
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
```

```
<!-- http://mvnrepository.com/artifact/mysql/
mysql-connector-java -->
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>5.1.39</version>
</dependency>

</dependencies>
<build>
  <finalName>aula24</finalName>
</build>
</project>
```

Atualize as dependencias do projeto:

C:\workspacemaven\aula24>mvn eclipse:eclipse -Dwtversion=2.0



Incluindo dependencias para JSF e Primefaces,
além de servlets, EL e JSTL

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>br.com.brq</groupId>
```

```
<artifactId>aula24</artifactId>
<packaging>war</packaging>
<version>1.0-SNAPSHOT</version>
<name>aula24 Maven Webapp</name>
<url>http://maven.apache.org</url>
<dependencies>

    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>3.8.1</version>
        <scope>test</scope>
    </dependency>

    <!-- http://mvnrepository.com/artifact/mysql/
    mysql-connector-java -->
    <dependency>
        <groupId>mysql</groupId>
        <artifactId>mysql-connector-java</artifactId>
        <version>5.1.39</version>
    </dependency>

    <!-- PrimeFaces -->
    <dependency>
        <groupId>org.primefaces</groupId>
        <artifactId>primefaces</artifactId>
        <version>5.1</version>
    </dependency>

    <!-- JSF 2 -->
    <dependency>
        <groupId>com.sun.faces</groupId>
        <artifactId>jsf-api</artifactId>
        <version>2.1.11</version>
    </dependency>
    <dependency>
        <groupId>com.sun.faces</groupId>
        <artifactId>jsf-impl</artifactId>
        <version>2.1.11</version>
    </dependency>

    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>jstl</artifactId>
        <version>1.2</version>
    </dependency>

    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>servlet-api</artifactId>
```

```

        <version>2.5</version>
    </dependency>

    <dependency>
        <groupId>javax.servlet.jsp</groupId>
        <artifactId>jsp-api</artifactId>
        <version>2.1</version>
    </dependency>

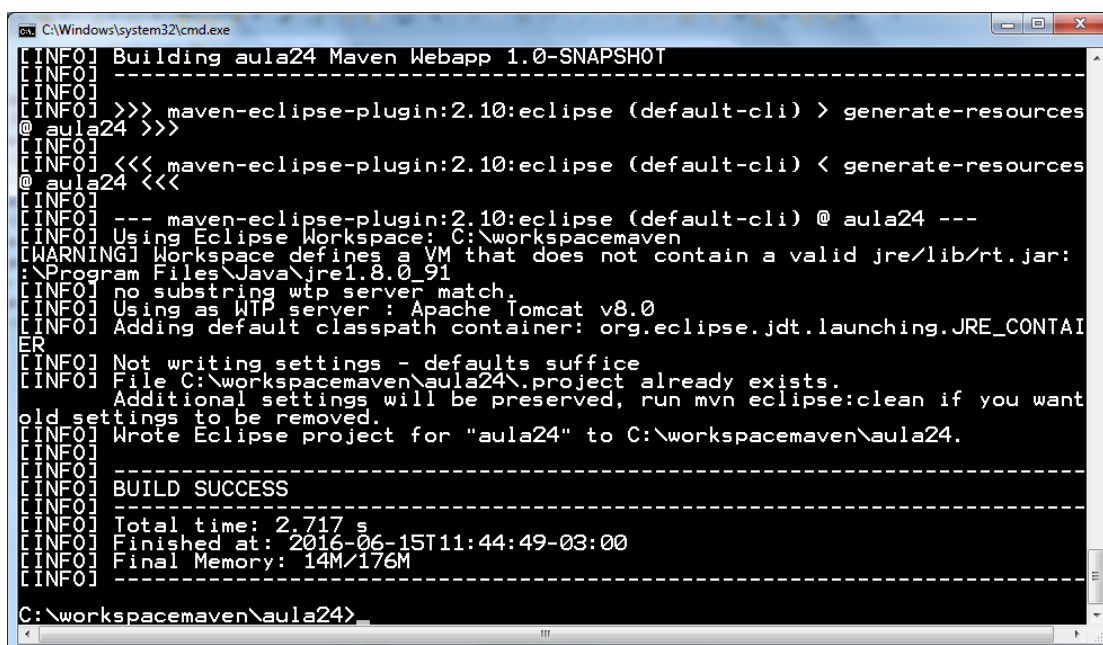
    <!-- EL -->
    <dependency>
        <groupId>org.glassfish.web</groupId>
        <artifactId>el-impl</artifactId>
        <version>2.2</version>
    </dependency>

</dependencies>
<build>
    <finalName>aula24</finalName>
</build>
</project>

```

Atualizando as dependencias:

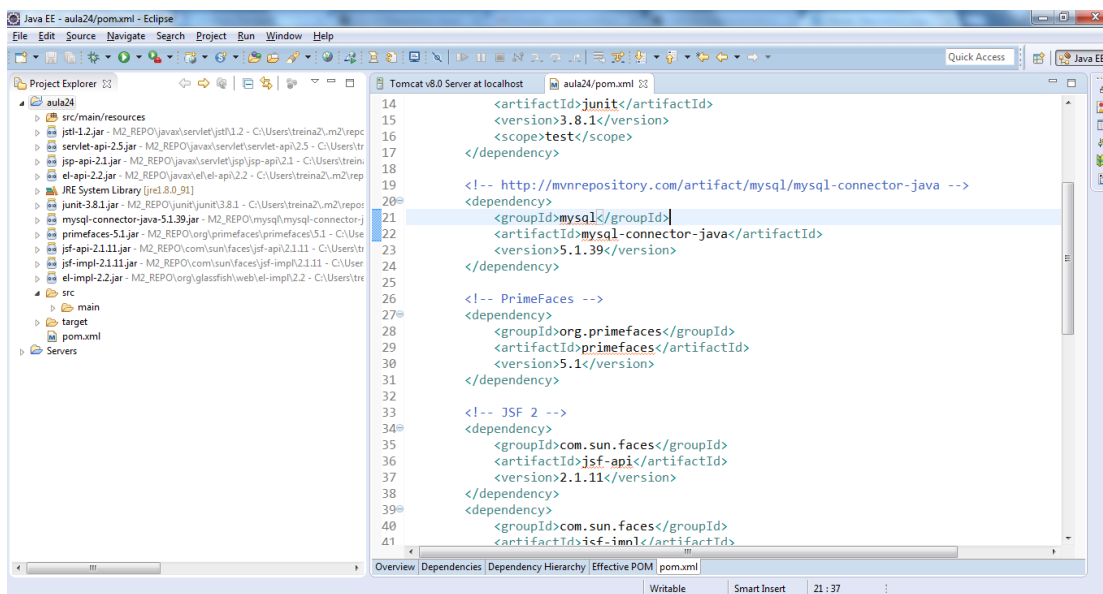
C:\workspacemaven\aula24>mvn eclipse:eclipse -Dwtversion=2.0



```

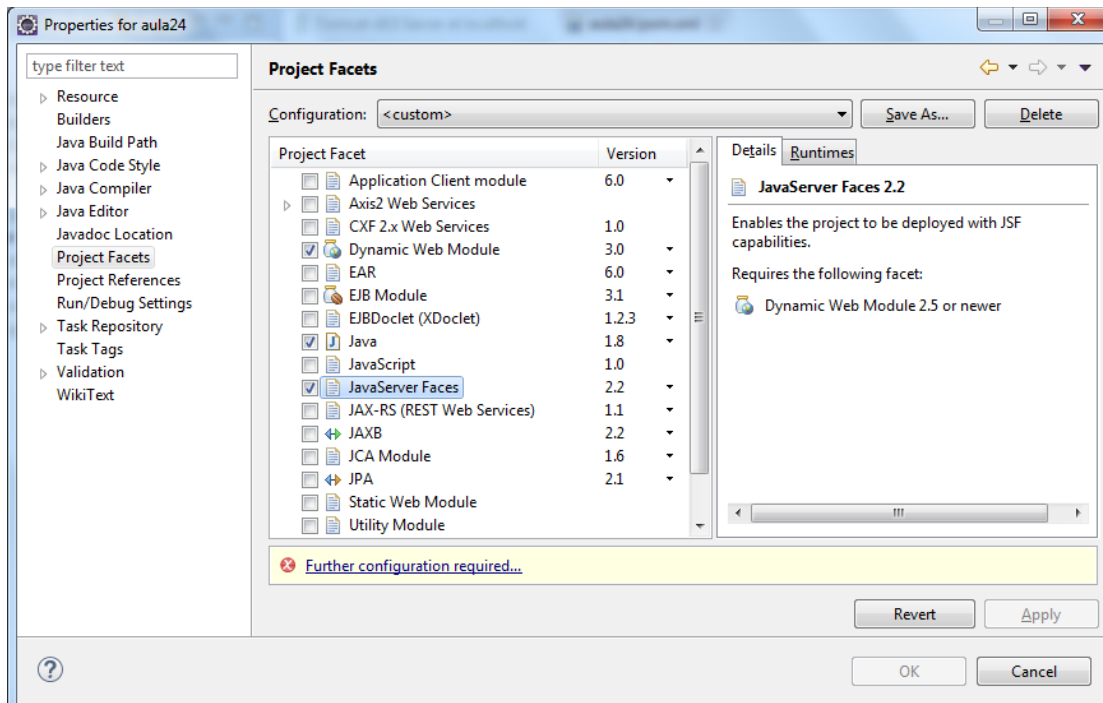
C:\Windows\system32\cmd.exe
[INFO] Building aula24 Maven Webapp 1.0-SNAPSHOT
[INFO]
[INFO]
[INFO] >>> maven-eclipse-plugin:2.10:eclipse (default-cli) > generate-resources
@ aula24 >>>
[INFO]
[INFO] <<< maven-eclipse-plugin:2.10:eclipse (default-cli) < generate-resources
@ aula24 <<<
[INFO]
[INFO] --- maven-eclipse-plugin:2.10:eclipse (default-cli) @ aula24 ---
[INFO] Using Eclipse Workspace: C:\workspacemaven
[WARNING] Workspace defines a VM that does not contain a valid jre/lib/rt.jar:
:\Program Files\Java\jre1.8.0_91
[INFO] no substring wtp server match.
[INFO] Using as WTP server : Apache Tomcat v8.0
[INFO] Adding default classpath container: org.eclipse.jdt.launching.JRE_CONTAI
ER
[INFO] Not writing settings - defaults suffice
[INFO] File C:\workspacemaven\aula24\.project already exists.
Additional settings will be preserved, run mvn eclipse:clean if you want
old settings to be removed.
[INFO] Wrote Eclipse project for "aula24" to C:\workspacemaven\aula24.
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 2.717 s
[INFO] Finished at: 2016-06-15T11:44:49-03:00
[INFO] Final Memory: 14M/176M
[INFO]
C:\workspacemaven\aula24>

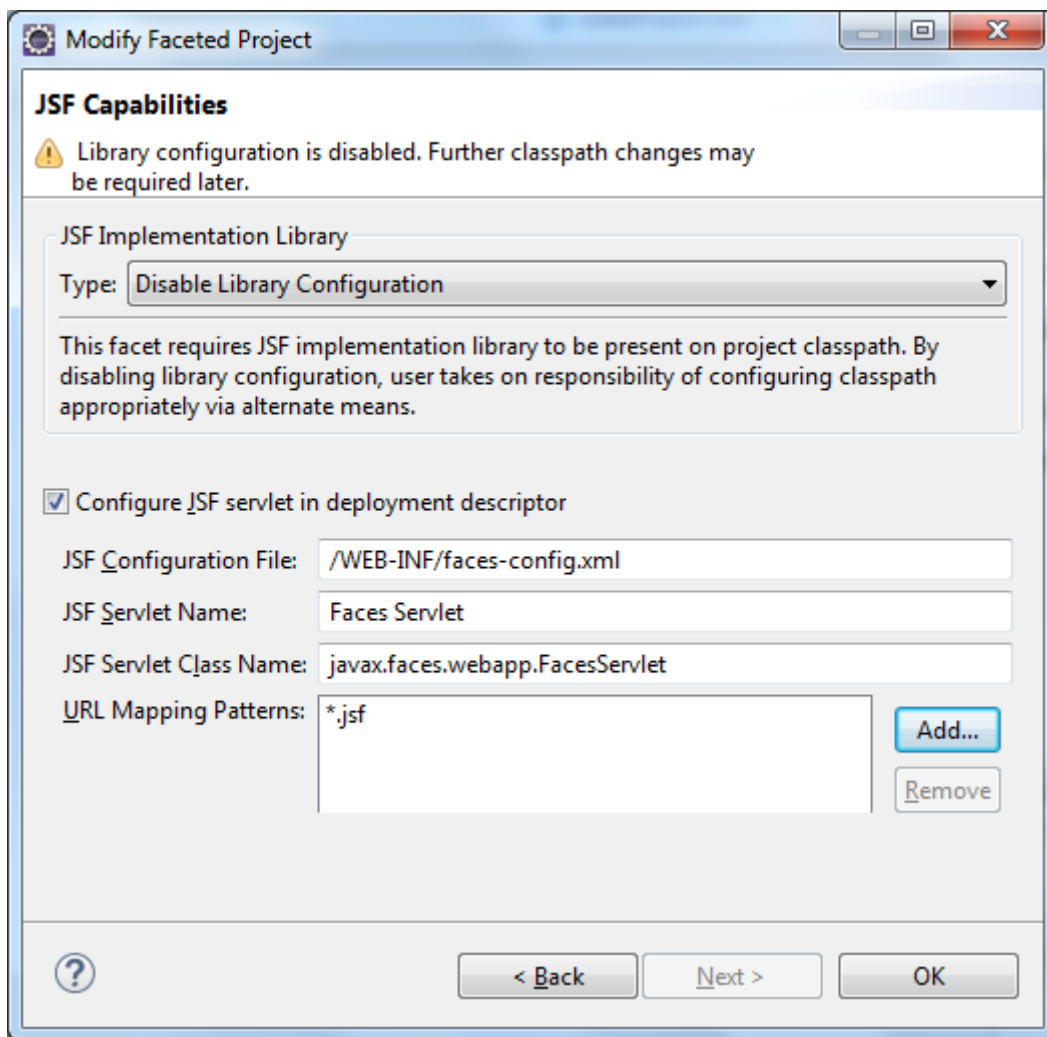
```



Configurando o JSF no projeto:

- Build Path / Configure BuildPath





Criando a página inicial:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:ui="http://java.sun.com/jsf/faceslets"
    xmlns:h="http://java.sun.com/jsf/html"
    xmlns:f="http://java.sun.com/jsf/core"
    xmlns:p="http://primefaces.org/ui">

    <h:head>
    </h:head>
    <h:body>

        <p:panel header="Painel de Produto">

            <h:form id="form1">
                Nome:
                <p:inputText />
            <br /></h:form>
```



```
                <p:commandButton value="Cadastrar" />
            </h:form>

        </p:panel>

    </h:body>

</html>
```

Entidade simples:

```
package entity;
```

```
public class Produto {

    private Integer id;
    private String nome;

    public Produto(Integer id, String nome) {
        super();
        this.id = id;
        this.nome = nome;
    }

    public Produto() {
        super();
    }

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }
}
```

ManagedBean:

```
package manager;
```

```
import javax.faces.application.FacesMessage;
import javax.faces.bean.ManagedBean;
import javax.faces.bean.RequestScoped;
import javax.faces.context.FacesContext;
```

```
import entity.Produto;

@ManagedBean(name="pBean")
@RequestScoped
public class ProdutoBean {

    private Produto produto;

    public ProdutoBean() {
        produto = new Produto();
        produto.setNome("Camisa");
    }

    public Produto getProduto() {
        return produto;
    }

    public void setProduto(Produto produto) {
        this.produto = produto;
    }

    public void cadastrar(){
        FacesContext fc = FacesContext.getCurrentInstance();

        fc.addMessage("form1", new FacesMessage
            ("Produto " + produto.getNome()));
    }
}
```

Classe de acesso ao banco de dados:

```
package persistence;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

public class Dao {

    Connection con;
    PreparedStatement stmt;
    ResultSet rs;

    private final String URL = "jdbc:mysql://localhost:3306/aula24";
    private final String USER = "root";
    private final String PASS = "brqbrq";

    public void open()throws Exception{
        Class.forName("com.mysql.jdbc.Driver");
        con = DriverManager.getConnection(URL, USER, PASS);
    }
}
```

```
        public void close()throws Exception{
            con.close();
        }
    }
}
```

Classe de persistencia:

```
package persistence;

import java.util.ArrayList;
import java.util.List;

import entity.Produto;

public class ProdutoDao extends Dao {

    public void cadastrar(Produto produto)throws Exception{
        open();

        stmt = con.prepareStatement("INSERT INTO produto
                                   VALUES(NULL, ?)");
        stmt.setString(1, produto.getNome());
        stmt.execute();

        close();
    }

    public List<Produto> buscarNome(String nome)throws Exception{
        open();
        List<Produto> lista = new ArrayList<Produto>();

        stmt = con.prepareStatement("SELECT * FROM produto
                                   WHERE nome LIKE ?");
        stmt.setString(1, nome);
        rs = stmt.executeQuery();

        while(rs.next()){
            Produto p = new Produto(rs.getInt("idProduto"),
                                    rs.getString("nome"));
            lista.add(p);
        }

        close();
        return lista;
    }
}
```

Voltando ao ManagedBean:

```
package manager;

import java.util.ArrayList;
import java.util.List;
```

```
import javax.faces.application.FacesMessage;
import javax.faces.bean.ManagedBean;
import javax.faces.bean.RequestScoped;
import javax.faces.context.FacesContext;

import persistence.ProdutoDao;

import entity.Produto;

@ManagedBean(name="pBean")
@RequestScoped
public class ProdutoBean {

    private Produto produto;
    private List<Produto> lista;

    public ProdutoBean() {
        produto = new Produto();
        lista = new ArrayList<Produto>();
    }

    public List<Produto> getLista() {
        return lista;
    }

    public void setLista(List<Produto> lista) {
        this.lista = lista;
    }

    public Produto getProduto() {
        return produto;
    }

    public void setProduto(Produto produto) {
        this.produto = produto;
    }

    public void cadastrar(){
        FacesContext fc = FacesContext.getCurrentInstance();

        try{
            new ProdutoDao().cadastrar(produto);
            produto = new Produto();
        }catch(Exception e){
            e.printStackTrace();
        }

        fc.addMessage("form1", new FacesMessage("Produto
            cadastrado com sucesso"));
    }
}
```

```
public void consultar(){
    try{
        lista = new ProdutoDao().buscarNome(produto.getNome()
                                                + "%");
    }catch(Exception e){
        e.printStackTrace();
    }
}
```

Finalizando a página:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:p="http://primefaces.org/ui">

    <h:head>
    </h:head>

    <h:body>

        <p:panel header="Painel de Produto">

            <h:form id="form1">
                Nome:
                <p:inputText value="#{pBean.produto.nome}" />
                <br /><br />
                <p:commandButton value="Cadastrar"
                                actionListener="#{pBean.cadastrar}"
                                update="@form" />
                <p:growl/></p:growl>
            </h:form>
            <br />
            <a href="consultar.jsf">Consultar Produto</a>
        </p:panel>

    </h:body>

</html>
```

Página de consulta:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core">
```

```
xmlns:p="http://primefaces.org/ui">

<h:head>
</h:head>

<h:body>

    <h:form id="form1">

        <p:panel header="CONSULTAR PRODUTO">

            <h:outputText value="Nome: " />
            <p:inputText value="#{pBean.produto.nome}" />
            <p:commandButton value="Buscar produto"
                actionListener="#{pBean.consultar}" update="@form" />

            <p:separator />

            <p:dataTable value="#{pBean.lista}" var="prod">
                <p:column headerText="IDPRODUTO">
                    <h:outputText value="#{prod.id}" />
                </p:column>

                <p:column headerText="NOME">
                    <h:outputText value="#{prod.nome}" />
                </p:column>
            </p:dataTable>

        </p:panel>
        <a href="painel.jsf">Voltar</a>
    </h:form>

</h:body>

</html>
```

Finalizando o ManagedBean

```
package manager;

import java.util.ArrayList;
import java.util.List;

import javax.faces.application.FacesMessage;
import javax.faces.bean.ManagedBean;
import javax.faces.bean.RequestScoped;
import javax.faces.context.FacesContext;

import persistence.ProdutoDao;

import entity.Produto;

@ManagedBean(name="pBean")
@RequestScoped
```

```
public class ProdutoBean {

    private Produto produto;
    private List<Produto> lista;

    public ProdutoBean() {
        produto = new Produto();
        lista = new ArrayList<Produto>();
    }

    public List<Produto> getLista() {
        return lista;
    }

    public void setLista(List<Produto> lista) {
        this.lista = lista;
    }

    public Produto getProduto() {
        return produto;
    }

    public void setProduto(Produto produto) {
        this.produto = produto;
    }

    public void cadastrar(){
        FacesContext fc = FacesContext.getCurrentInstance();

        try{

            new ProdutoDao().cadastrar(produto);
            produto = new Produto();

        }catch(Exception e){
            e.printStackTrace();
        }

        fc.addMessage("form1", new FacesMessage
            ("Produto cadastrado com sucesso"));
    }

    public void consultar(){
        try{
            lista = new ProdutoDao().buscarNome
                (produto.getNome() + "%");
        }catch(Exception e){
            e.printStackTrace();
        }
    }
}
```