

## **Introdução ao conceito de orientação a objetos**

A orientação a objetos (O.O) é um paradigma que pode ser aplicado a programação, entre outras etapas do processo de desenvolvimento de software, conceitualmente falando consiste em representar elementos do mundo real de um determinado contexto do qual se deseja desenvolver um programa/sistema através de estruturas definidas dentro da O.O.

Diferente da programação estruturada que está focada nas ações de um sistema e implementa isso através de procedimentos (procedure) e funções (function) a O.O se preocupa com seus objetos e como esses se relacionam na realização de suas tarefas.

Outra característica muito presente na O.O é a preocupação em agrupar dados e ações que manipulam esses dados em artefatos de software contextualizados de maneira que cada um desses artefatos sejam especializados o que permite um desenvolvimento e manutenção de código mais rápido e racional.

Os tópicos abaixo ilustram esses conceitos e estruturas da O.O.

### **Classes**

É uma representação genérica “de alguma coisa” do mundo real dentro de um sistema/software programado em O.O. Podemos usar o seguinte exemplo, imagine que uma construtora vai fazer um condomínio de casas e para reduzir os custos foi decidido que todas as casas serão iguais, então uma empresa de engenharia civil é contratada para a fazer a planta da casa de onde serão construídas várias casas no terreno que a construtora comprou para fazer o condomínio.

A planta nesse caso é uma classe, uma abstração genérica de onde serão feitas diversas casas, quantas forem necessárias ou requeridas.

Agora vamos passar esse conceito para um sistema (software) bancário, imagine a parte desse sistema que se refere ao gerenciamento de contas, tanto no que diz respeito a abertura, quanto ao uso da conta pelos clientes, criamos então uma classe chamada conta que é uma representação genérica de contas dentro do sistema da instituição financeira, porem essa classe apenas representa as contas no que diz respeito aos dados e ações que essa possui assim como a planta da casa do exemplo anterior e partindo dessa classe conta o banco pode abrir quantas contas forem necessárias e cada cliente poderá manipular suas contas.

## Atributos

Os atributos são um dos elementos que constituem uma classe e representam os dados (características) da classe, vamos voltar ao exemplo da planta (classe) da casa. Quais seriam os seus atributos ou as características que a definem, poderíamos dizer que esses atributos são, número de cômodos, quantidade de portas, quantidade de janelas, a cor das paredes etc. Porém tenha o seguinte conceito em mente, nesse ponto os atributos são apenas “espaços” que representam onde essas informações serão armazenadas, quando tivermos essa casa construída alguns desses atributos podem ter valores diferentes entre si, por exemplo, o atributo cor das paredes pode variar de uma casa para outra, mas todas as casas terão uma cor de parede.

Vamos aplicar esse mesmo conceito na classe conta, poderíamos dizer que essa classe possui os seguintes atributos, número da conta, tipo da conta, saldo, limite, senha, etc. Agora novamente pense da seguinte forma, quantas contas um banco pode possuir, milhares talvez milhões e cada conta é única, mas o que faz cada conta ser única são os valores associados aos seus atributos, alguns até podem se repetir como o saldo por exemplo, mas outros atributos jamais serão idênticos como o número da conta.

## Métodos

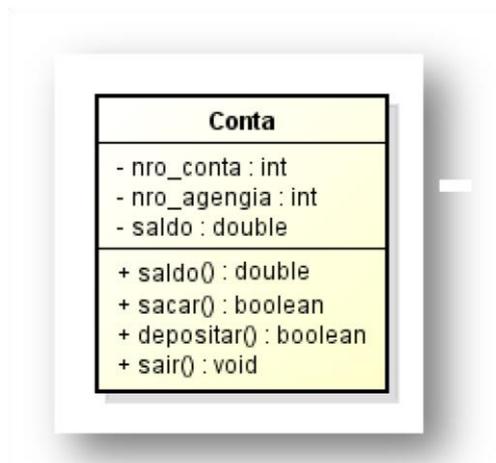
Métodos são outra estrutura da O.O que compõe uma classe e essas estruturas são responsáveis pelas tarefas/ações que a classe executa.

Pulando direto para o nosso exemplo do sistema bancário nossa classe conta poderia ter os seguintes métodos sacar, depositar, ver o saldo, transferir, etc.

Métodos executam tarefas específicas e isso permite maior facilidade na manutenção de um sistema além de permitir o reuso de métodos em outras classes de outros sistemas, uma grande característica da O.O.

Uma outra característica de métodos é que muitas vezes um método pode depender de outro para realizar uma tarefa, nesses casos o método dependente envia uma mensagem solicitando alguma ação de outro, logo métodos se comunicam entre si quando há necessidade.

A imagem abaixo ilustra um diagrama de classe da UML (Unified Modeling Language) que é uma linguagem de modelagem de sistemas orientada a objetos:



No diagrama de classes acima temos a representação gráfica da nossa classe conta, note que o diagrama em questão se divide em três retângulos onde o primeiro representa o nome da classe, o segundo representa os atributos da classe e o terceiro os métodos. Você deve ter notado também que existem símbolos a esquerda tanto dos atributos quanto dos métodos, esses símbolos representam a visibilidade dessas estruturas, observe também que a direita dos atributos e dos métodos temos um sinal de dois pontos seguido por uma palavra que varia de um item para o outro, isso é chamado de tipo, falaremos sobre essas questões mais adiante em nosso curso.

### **Objetos/Instâncias**

A classe seus atributos e métodos não podem ser utilizados diretamente pois são representações abstratas de elementos do mundo real dentro de um sistema computacional, tais representações podem ser de elementos físicos como um cliente ou produto, por exemplo ou podem ser representações de elementos conceituais como um processo de venda ou validação de crédito, por exemplo.

Para utilizar então as classes seus atributos e métodos é necessário criar uma instância de objeto dessa classe, esse termo instância vem do fato de que esses objetos que são criados a partir de classes existem apenas por um período, ou seja, um instante de tempo, salvo para métodos STATIC.

Vamos voltar ao nosso exemplo do sistema bancário, um cliente chega ao caixa eletrônico e insere seu cartão a leitora do caixa (hardware) lê os dados do “chip” do cartão e recolhe dados como número da conta e da agência, por exemplo a partir daí o sistema do caixa eletrônico cria um objeto da classe cliente para aquele cliente recuperando certas informações da base dados do banco referentes a esse cliente e associa todos esses dados aos atributos do objeto que foi criado a partir da classe “Conta”, o cliente por sua vez começa a executar

as ações que deseja, verificar o saldo, realizar um depósito ou um saque, enfim tudo o que desejar acessando assim os métodos do objeto da classe “**Conta**”, ou seja, a classe em si não é executada mas sim um “**objeto**” que foi criado a partir dela. Poderíamos então dizer que a classe funciona como um “**Repositório**”. Pense em um banco de grande porte que possui agência espalhadas por todo o país, teríamos então múltiplos cliente realizando operações bancárias em diversos caixas eletrônicos simultaneamente, cada um desses clientes teria seu próprio objeto de conta instanciando no sistema do caixa eletrônico que está usando, mas a classe em si permanece intacta servindo apenas de “modelo” para cada novo objeto de si solicitado pelo sistema, quando um cliente qualquer encerra o uso da aplicação do caixa eletrônico seu objeto de conta é destruído junto com a sessão, ou seja a instância do objeto deixou de existir e o processo se repete indefinidamente.

## O JAVA

A linguagem de programação JAVA foi criada em 1995 pela empresa Sun Microsystems e sua popularização se deu inicialmente em aplicações web no ano de 2003 a Sun disponibilizou boa parte do código sob os termos da licença GNU (General Public License) e em 2007 toda a linguagem foi disponibilizada sob a mesma licença tornando assim o JAVA Open Source permitindo assim seu uso, estudo e alteração livre a qualquer pessoa com conhecimento para tal.

Características da linguagem:

- Orientada a objetos;
- Fortemente tipada;
- Permite tratamento de exceções;
- Multiplataforma através do uso da máquina virtual do JAVA;
- Gerenciamento da memória principal através de “Garbage Collector”
- Compilada (Bit Codes);
- A JVM usa o conceito de “hot code”;

Em 2010 a Sun foi comprada pela ORACLE por 7,4 bilhões de dólares o JAVA continua sendo Open Source.

Segundo a Oracle o JAVA hoje representa os seguintes números:

- 97% dos Desktops Corporativos executam o Java
- 89% dos Desktops (ou Computadores) nos EUA Executam Java

- 9 Milhões de Desenvolvedores de Java em Todo o Mundo
- A Escolha Nº 1 para os Desenvolvedores
- Plataforma de Desenvolvimento Nº 1
- 3 Bilhões de Telefones Celulares Executam o Java
- 100% dos Blu-ray Disc Players Vêm Equipados com o Java
- 5 bilhões de Placas Java em uso
- 125 milhões de aparelhos de TV executam o Java
- 5 dos 5 Principais Fabricantes de Equipamento Original Utilizam o Java ME

## O ambiente JAVA

Para que possamos executar e/ou desenvolver programas em JAVA é necessário antes ter o ambiente da linguagem devidamente instalado e configurado no sistema operacional que vamos utilizar.

Recursos necessários para desenvolvimento e execução de aplicações desenvolvidas em JAVA:

**JRE** – Java Runtime Environment que significa em português Ambiente de Tempo de Execução JAVA é usado para que o sistema operacional execute programas desenvolvidos em JAVA é composto por várias bibliotecas chamadas de API (Application Programming Interface) necessárias para que as aplicações JAVA “rodem” corretamente no S.O adotado. O JRE deve ser baixado e instalado somente por clientes, ou seja, computadores que vão apenas executar os programas feitos em JAVA.

Seu download pode ser feito em: [https://www.java.com/pt\\_BR/download/](https://www.java.com/pt_BR/download/)

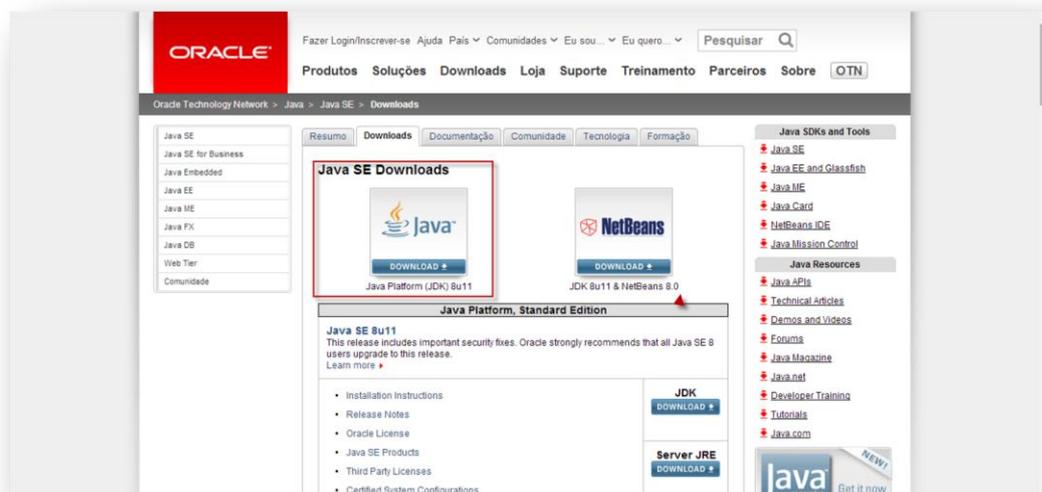
**JVM** – Java Virtual Machine que significa em português Máquina Virtual do Java é a responsável por executar os programas JAVA em vários sistemas operacionais (plataformas) diferentes. A JVM já vem inclusa no JRE, logo é impossível fazer o download da JVM. A JVM é o que possibilita uma das características mais fortes do JAVA a capacidade de “rodar” em múltiplos sistemas operacionais, cada sistema operacional tem sua própria máquina virtual que faz o processo de interpretação dos “códigos JAVA compilados” para o sistema operacional onde a aplicação está sendo executada.

**JDK** – Java Development Kit que significa em português Kit de Desenvolvimento Java que nada mais é do que todo o conjunto de “APIs” do JAVA, ou seja a linguagem propriamente dita. O JDK permite que sejam desenvolvidas aplicações JAVA em qualquer vertente que a linguagem atenda.

Seu download pode ser feito em:

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

**JAVA SE** – O Java Standard Edition é a composição de ferramentas básicas necessárias para iniciar o desenvolvimento de uma aplicação JAVA é composta pelas APIs do JAVA, um compilador JAVA e uma JVM. Geralmente é usada para



A segunda tela nos permite escolher o JDK para a plataforma e arquitetura que desenvolver aplicações desktop stand alone que independem ou não precisam por exemplo de conexões com redes ou servidores.

**JAVA EE** – O Java Enterprise Edition é uma versão mais robusta do JAVA usada para o desenvolvimento de aplicações corporativas de auto desempenho e alta disponibilidade é possível por exemplo aplicar conceitos de mapeamento objeto relacional, arquiteturas de multicamada, arquiteturas distribuídas, web services entre outras.

**Observação:** O que define um ou outro ambiente do JAVA citado acima são os recursos que instalamos e configuramos quando estamos montando nosso ambiente JAVA o escopo desse curso vai abordar o JAVA SE.

## Instalação e configuração do JAVA no Windows

**1º Passo** – Fazer download do JDK no link: <http://www.oracle.com/technetwork/pt/java/javase/downloads/index.html> preste atenção na arquitetura de seu computador (32 bits ou 64 bits); A imagem abaixo ilustra a página inicial do download: utilizamos no me caso foi Windows 64 bits:

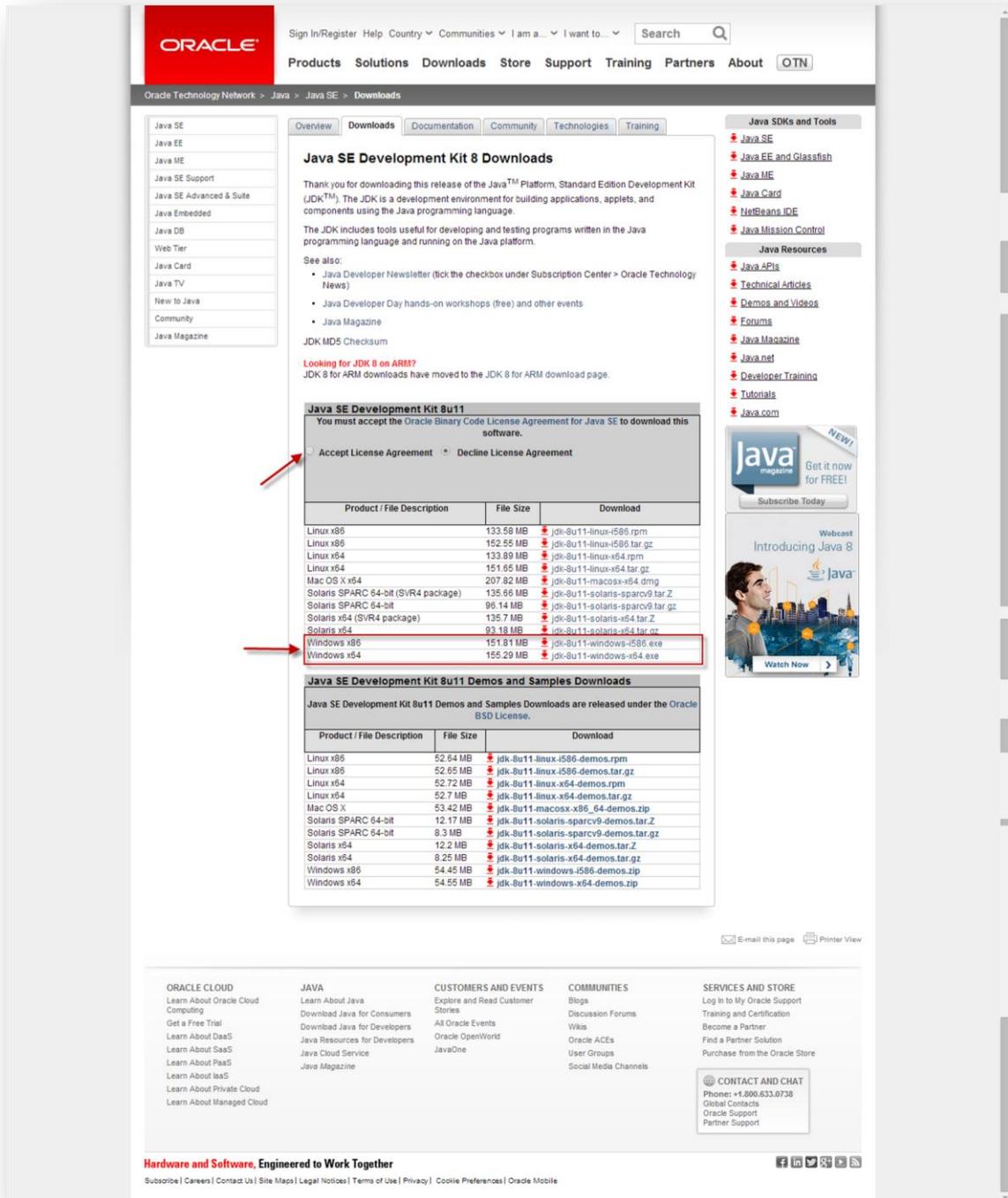
**FUTURE SCHOOL – Cursos de Computação**

Fone: (0XX11) 983422503 [www.osasconamao.com.br](http://www.osasconamao.com.br)



**FUTURE SCHOOL – Cursos de Computação**

Fone: (0XX11) 983422503 [www.osasconamao.com.br](http://www.osasconamao.com.br)



**ORACLE** Sign In/Register Help Country Communities I am a... I want to... Search

Products Solutions Downloads Store Support Training Partners About OTN

Oracle Technology Network > Java > Java SE > Downloads

Overview Downloads Documentation Community Technologies Training

### 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 (tick the checkbox under Subscription Center > Oracle Technology News)
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK MDS Checksum

**Looking for JDK 8 on ARM?**  
JDK 8 for ARM downloads have moved to the JDK 8 for ARM download page.

**Java SE Development Kit 8u11**

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 x86	133.58 MB	<a href="#">jdk-8u11-linux-i586.rpm</a>
Linux x86	152.55 MB	<a href="#">jdk-8u11-linux-i586.tar.gz</a>
Linux x64	133.89 MB	<a href="#">jdk-8u11-linux-x64.rpm</a>
Linux x64	151.65 MB	<a href="#">jdk-8u11-linux-x64.tar.gz</a>
Mac OS X x64	207.82 MB	<a href="#">jdk-8u11-macosx-x64.dmg</a>
Solaris SPARC 64-bit (SVR4 package)	135.66 MB	<a href="#">jdk-8u11-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	96.14 MB	<a href="#">jdk-8u11-solaris-sparcv9.tar.gz</a>
Solaris x64 (SVR4 package)	135.7 MB	<a href="#">jdk-8u11-solaris-x64.tar.Z</a>
Solaris x64	93.18 MB	<a href="#">jdk-8u11-solaris-x64.tar.gz</a>
Windows x86	151.81 MB	<a href="#">jdk-8u11-windows-i586.exe</a>
Windows x64	155.29 MB	<a href="#">jdk-8u11-windows-x64.exe</a>

### Java SE Development Kit 8u11 Demos and Samples Downloads

Java SE Development Kit 8u11 Demos and Samples Downloads are released under the Oracle BSD License.

Product / File Description	File Size	Download
Linux x86	52.64 MB	<a href="#">jdk-8u11-linux-i586-demos.rpm</a>
Linux x86	52.65 MB	<a href="#">jdk-8u11-linux-i586-demos.tar.gz</a>
Linux x64	52.72 MB	<a href="#">jdk-8u11-linux-x64-demos.rpm</a>
Linux x64	52.7 MB	<a href="#">jdk-8u11-linux-x64-demos.tar.gz</a>
Mac OS X	53.42 MB	<a href="#">jdk-8u11-macosx-x86_64-demos.zip</a>
Solaris SPARC 64-bit	12.17 MB	<a href="#">jdk-8u11-solaris-sparcv9-demos.tar.Z</a>
Solaris SPARC 64-bit	8.3 MB	<a href="#">jdk-8u11-solaris-sparcv9-demos.tar.gz</a>
Solaris x64	12.2 MB	<a href="#">jdk-8u11-solaris-x64-demos.tar.Z</a>
Solaris x64	8.25 MB	<a href="#">jdk-8u11-solaris-x64-demos.tar.gz</a>
Windows x86	54.45 MB	<a href="#">jdk-8u11-windows-i586-demos.zip</a>
Windows x64	54.55 MB	<a href="#">jdk-8u11-windows-x64-demos.zip</a>

E-mail this page Printer View

**ORACLE CLOUD**

Learn About Oracle Cloud Computing

Get a Free Trial

Learn About DaaS

Learn About SaaS

Learn About PaaS

Learn About IaaS

Learn About Private Cloud

Learn About Managed Cloud

**JAVA**

Learn About Java

Download Java for Consumers

Download Java for Developers

Java Resources for Developers

Java Cloud Service

Java Magazine

**CUSTOMERS AND EVENTS**

Explore and Read Customer Stories

All Oracle Events

Oracle OpenWorld

JavaOne

**COMMUNITIES**

Blogs

Discussion Forums

Wikis

Oracle ACEs

User Groups

Social Media Channels

**SERVICES AND STORE**

Log In to My Oracle Support

Training and Certification

Become a Partner

Find a Partner Solution

Purchase from the Oracle Store

**CONTACT AND CHAT**

Phone: +1.800.633.0738

Global Contacts

Oracle Support

Partner Support

**Hardware and Software, Engineered to Work Together**

Subscribe | Careers | Contact Us | Site Maps | Legal Notices | Terms of Use | Privacy | Cookie Preferences | Oracle Mobile

**\*NÃO ESQUECE DE ACEITAR A LICENÇA.**

**FUTURE SCHOOL – Cursos de Computação**

Fone: (0XX11) 983422503 [www.osasconamao.com.br](http://www.osasconamao.com.br)

**2º Passo** – O processo de instalação do JDK é como de qualquer outro programa, se realizada através de um instalador, mantenha todas as opções padrão.

### **IDEs para JAVA.**

Uma IDE (Integrated Development Environment) do português Ambiente Integrado de Desenvolvimento nada mais é do que um programa utilizado no apoio ao desenvolvimento de software, esses programas reúnem várias ferramentas que auxiliam e aumentam a produtividade de um(a) programador(a) de computador.

Dada a popularidade do JAVA existem inúmeras IDEs as duas mais conhecidas são o Eclipse que é open source sob a licença EPL (Eclipse Public License) podendo ser usado livremente para qualquer fim, além do JAVA o Eclipse dá suporte para diversas outras linguagens e no que se refere ao JAVA pode ser usado tanto para desenvolver código como interface gráfica, atualmente o Eclipse é o IDE recomendado pela comunidade de desenvolvimento Android devido a sua grande integração com a API JAVA para essa plataforma.

Outra IDE muito conhecida no mundo JAVA é o NetBeans que também é open source e também permite desenvolver em diversas outras linguagens além do JAVA. O NetBeans nasceu proprietário, porém em 1999 a Sun Microsystems o comprou e o tornou livre sob a licença CDDL (Common Development and Distribution License). O NetBeans também fornece um sólido suporte para desenvolvimento JAVA tanto na parte de código como na parte de interfaces gráficas, já vem com o banco de dados Derby embarcado e durante o processo de instalação da IDE podemos optar em instalar de forma integrada o servidor de aplicações GlassFish e o servidor web JAVA Apache Tom Cat, essa será a nossa IDE nesse curso.

### **Adquirindo e instalando o NetBeans para Windows. O**

NetBeans pode ser baixado gratuitamente no link abaixo:

<https://netbeans.org/downloads/>

Além da plataforma e da arquitetura podemos também baixar versões com suporte para mais ou menos linguagens conforme ilustra a imagem abaixo:



Cada coluna disponibiliza uma versão com mais ou menos linguagens de acordo com a necessidade de cada desenvolvedor(a) minha recomendação para esse curso e baixar no mínimo a versão “Java EE”

O processo de instalação não requer nenhum conhecimento específico basta seguir a telas do instalador mantendo as opções padrão.

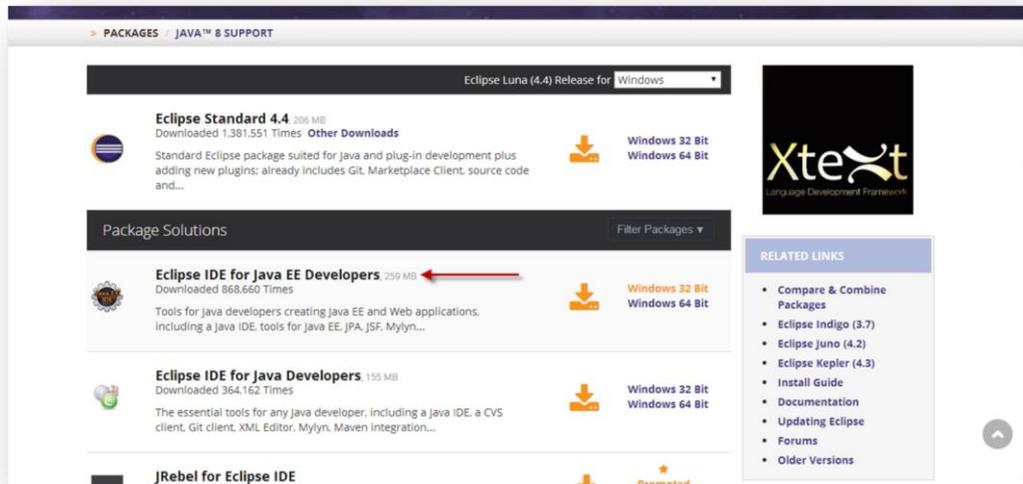
**Nota: Só é possível instalar o NetBeans após a correta instalação e configuração do JAVA demonstrada nos passos anteriores desse material.**

**Adquirindo e instalando o Eclipse para Windows.**

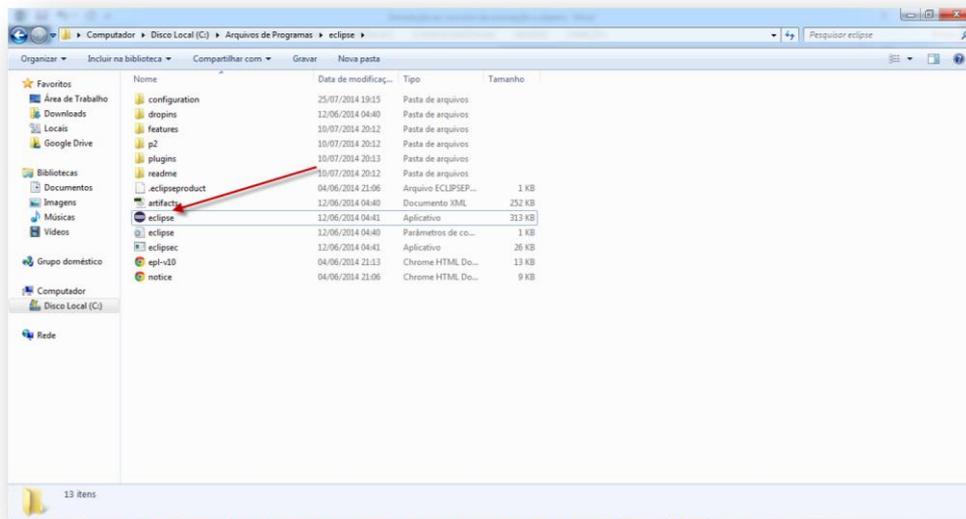
O Eclipse pode ser adquirido no link abaixo:

<https://www.eclipse.org/downloads/>

Diferente do Eclipse aqui você deve fazer o download da versão para seu sistema operacional e arquitetura conforme ilustra a imagem abaixo:



O processo de instalação do eclipse é ainda mais fácil do que o do Eclipse até pelo fato de que não há instalação no Eclipse, após realizar o download basta descompactar o arquivo no local de sua preferência e clicar no ícone conforme ilustra a imagem abaixo:



Nós recomendamos a criação de um atalho na área de trabalho. Agora estamos prontos para iniciar o desenvolvimento de aplicações em JAVA!

**FUTURE SCHOOL – Cursos de Computação**

Fone: (0XX11) 983422503 [www.osasconamao.com.br](http://www.osasconamao.com.br)