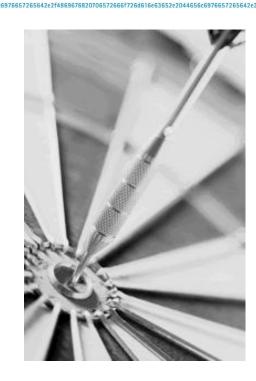


Module 2: Introduction to Java

Module Objective

- No final deste módulo, você poderá:
 - Descrever os principais conceitos que suportam a tecnologia Java
 - Explicar como o Java obtém independência de plataforma
 - Discuta as diferentes ferramentas e bibliotecas disponíveis como parte do Standard Java Development KIT (J2SE JDK)
 - Instale e configure o software, ferramentas e bibliotecas necessários para iniciar o Java
 - Escreva, compile e execute aplicativos Java simples



Brief History of Java

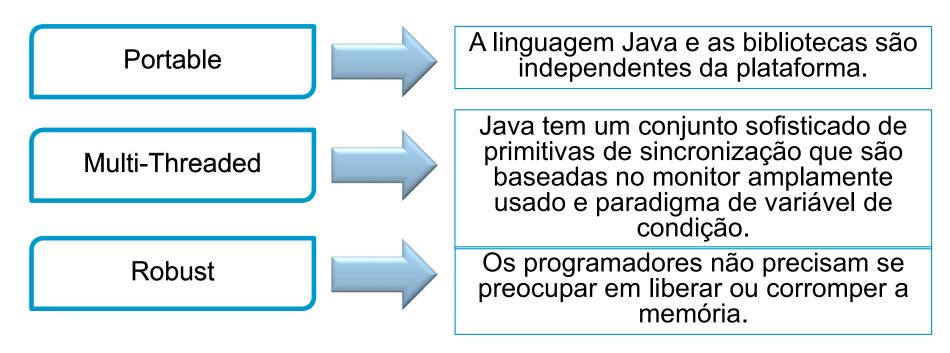
Java foi originalmente chamado de "Oak"



- Java era para ser uma linguagem de programação específica para dispositivos e vários pequenos dispositivos
- A especificação original:
 - Decodificadores direcionados para televisão a cabo que permitiam aos usuários controle e interatividade ao usar o serviço
 - A especificação Java pertence à Sun Microsystems
- A Internet recém-emergente provou ser uma plataforma muito melhor para a especificação:
 - A maneira como a Internet estava sendo usada tinha a mesma interatividade planejada para os clientes de televisão a cabo

The Java Programming Language

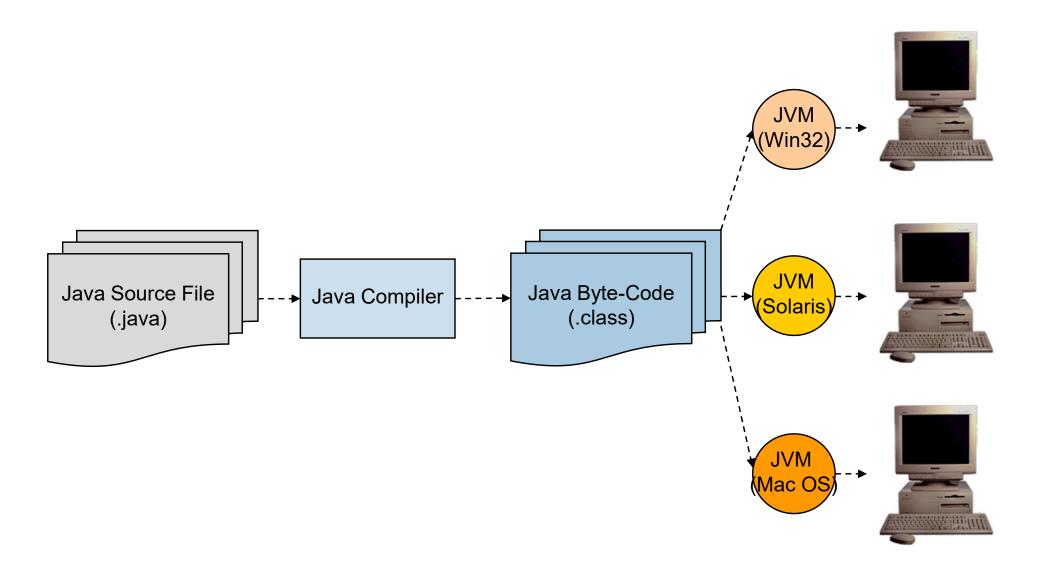
- A linguagem de programação Java é uma linguagem totalmente orientada a objetos.
- A edição padrão fornece bibliotecas e APIs pré-criadas que fornecem recursos úteis imediatamente.
- Java é:



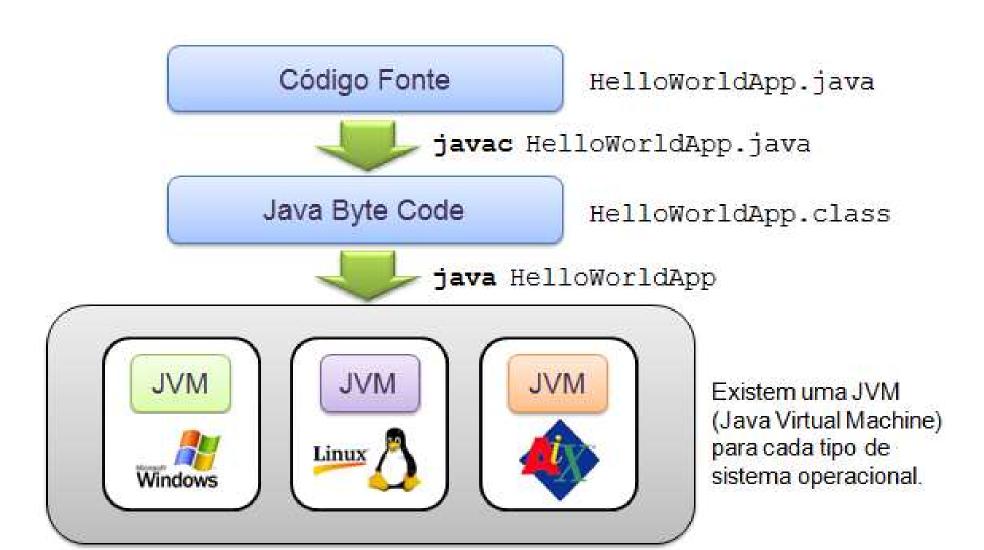
The Java Technology Platform

- Os aplicativos Java são executados em um ambiente de máquina virtual que:
 - Isola a plataforma;
 - Alcança portabilidade e desempenho;
 - Fornece segurança;
- O código-fonte Java é gravado como arquivos de texto simples (.java) que são compilados como códigos de bytes independentes da plataforma (.class)
- Os códigos de bytes Java são interpretados e executados pela máquina virtual que passa as instruções para a plataforma real;

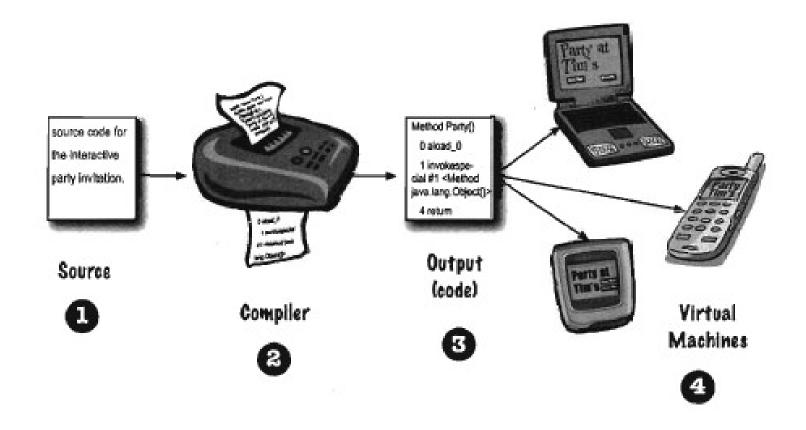
The Java Technology Platform



Java Virtual Machine



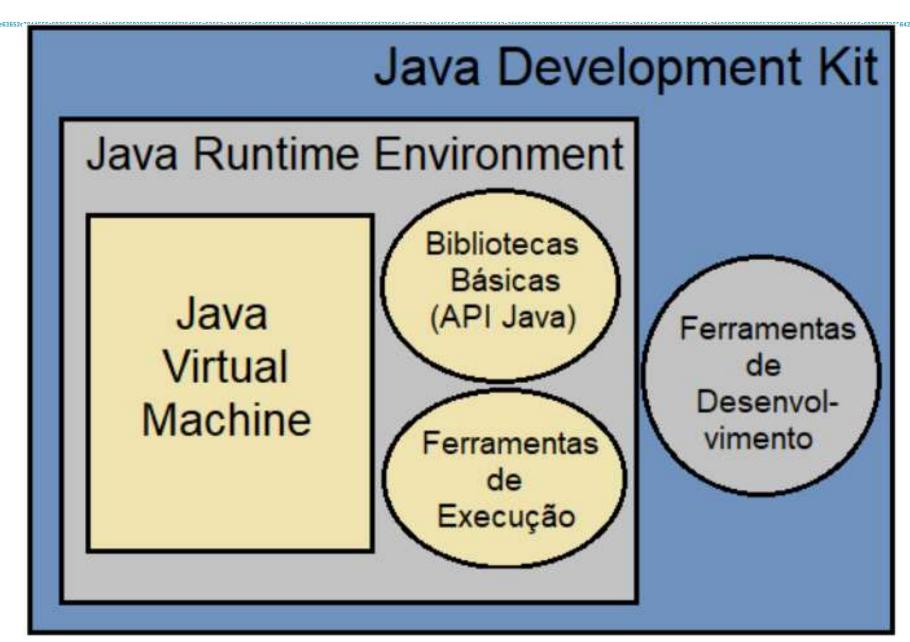
Java Virtual Machine



Java Development Kit

- Java Development Kit (JDK):
 - É um conjunto de software, ferramentas e bibliotecas que precisam ser instaladas para começar a escrever e compilar aplicativos Java
 - Instala a máquina virtual (ambiente de tempo de execução) necessária para executar aplicativos Java na plataforma
 - A máquina virtual pode ser baixada separadamente do JDK como um download do JRE (Java Runtime Environment)

JDK



Activity

- Instale o Java Development Kit;
- 2. Instale um IDE (Eclipse);
- 3. Crie o programa HelloWorld e compile-o usando javac;
- 4. Crie e configure um projeto dentro do Eclipse importando os arquivos de trabalho do SEF;
- 5. Compile e execute um aplicativo Java simples;



Installing the JDK

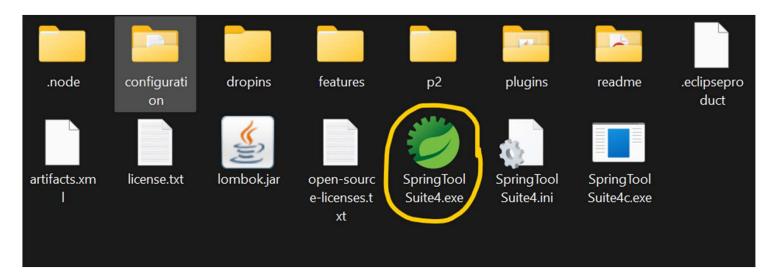
- O Contrato de Licença do JDK usado neste curso encontra-se na seção Apêndice deste módulo
- Se o instalador não estiver disponível localmente, acesse http://java.sun.com e faça o download do instalador do J2SE8
 - Certifique-se de baixar a versão 'JDK' apropriada para sua plataforma;
- Quando o instalador estiver disponível, execute o instalador e siga os procedimentos solicitados.
 - Basta escolher as opções padrão para qualquer opção de instalação;

Integrated Development Environment

- Os arquivos de origem Java estão no formato de texto e podem ser gravados usando qualquer editor de texto;
- Para maior produtividade, baixe e instale um Ambiente de Desenvolvimento Integrado (IDE). Um IDE:
 - Fornece um conjunto de ferramentas que auxiliam no desenvolvimento, teste e depuração de aplicativos Java;
 - Popular IDEs are:
 - NetBeans (from http://java.sun.com).
 - Eclipse (http://eclipse.org).

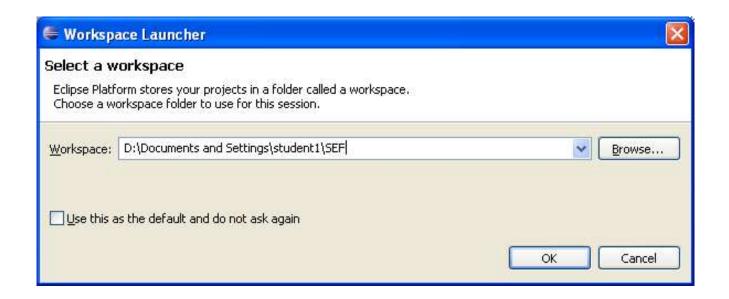
- O Eclipse usado neste curso é um OSS aprovado pela Accenture. O Contrato de Licença encontra-se na seção Apêndice deste módulo;
- https://spring.io/tools/ and download the Eclipse IDE.

Depois de extraído, vá para o diretório, localize o executável
 'SpringToolSuite4.exe' e crie um atalho para ele;



 Coloque o atalho em um local conveniente e clique duas vezes no atalho para iniciar o Eclipse IDE;

 Após a tela inicial, o IDE solicitará que você crie uma área de trabalho. Nomeie o espaço de trabalho "SEF".



O "workspace" é o diretório onde a IDE salva as configurações, projetos e outros metadados relacionados ao seu trabalho.

É um local central onde o STS guarda todas as informações sobre os projetos e suas configurações, permitindo que você trabalhe com vários projetos simultaneamente e que a IDE lembre quais projetos você está trabalhando.

- Se for a primeira vez que você usa o eclipse, a janela "Bem-vindo" deve fornecer algumas opções e tutoriais para iniciantes;
- Feche a janela por enquanto.
 Isso pode ser acessado a qualquer momento sob o 'Help->Welcome' menu.

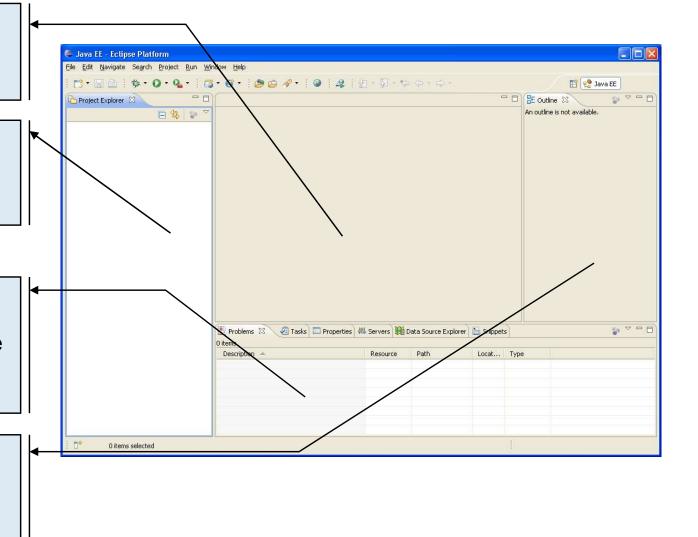


A janela do editor é onde o componente selecionado pode ser visualizado e editado

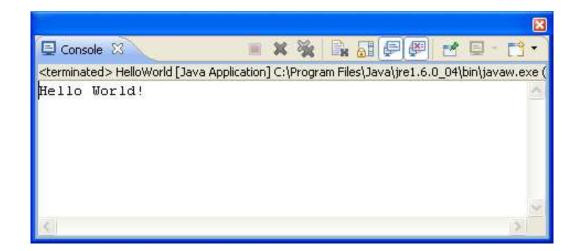
O project explorer permite navegar pelo conteúdo dos vários projetos que você criou na área de trabalho

As guias Informações na parte inferior fornecem informações adicionais sobre o aplicativo / componente selecionado

A visualização de estrutura de tópicos permite examinar os vários elementos de um componente selecionado no explorador de projeto



- O arquivo HelloWorld é um executável Java;
- Execute o arquivo clicando com o botão direito do mouse no file->Run As->Java application;
- A guia do console abaixo do editor principal deve mostrar os resultados do aplicativo;

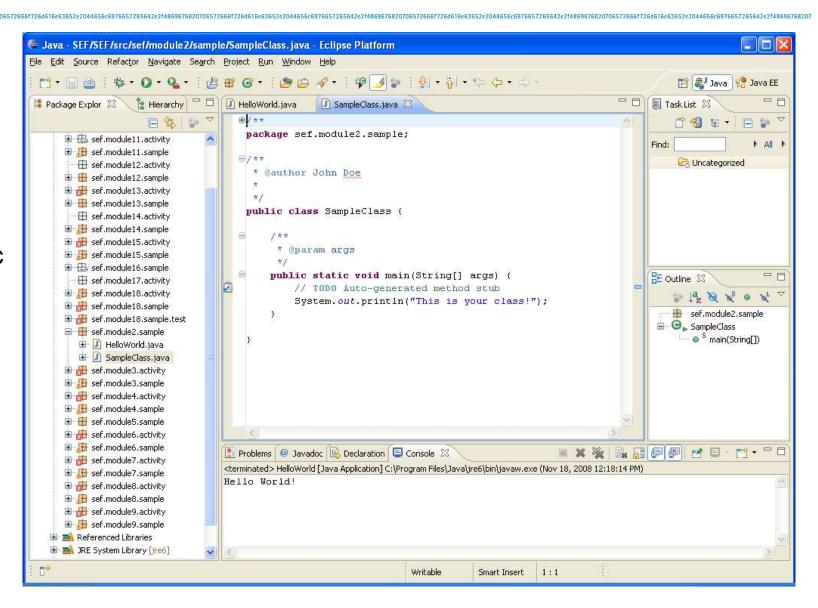


O que é uma classe Java?

- Classes são blocos de construção fundamentais de um programa Java;
- A letra inicial no nome de uma classe deve ser maiúscula;
- O método main () em uma classe serve como um ponto de entrada padrão.
 - Quando uma classe é executada com o interpretador Java, o sistema de tempo de execução inicia chamando o método main () da classe.

Criando uma nova classe Java

- Right-click on the package sef.module2.sa mple and select New->Class
- Dê um nome paraa classe 'SampleClass'
- Adicione 'public static void main(String arg[]) in the method stubs
- Click 'Generate Comments'
- Click Finish when done



Criando uma nova classe Java

Complete o main method digitando:

```
public static void main(String[] args) {
        System.out.println("This is your class!");
}
```

Compile e execute a nova classe e veja os resultados!

Comunidade JAVA na Accenture

Java Automation Blueprint | JAVA COMMUNITY



JAVA COMMUNITY







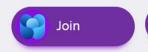
HOME

JAVA NEWS

JAVA ESSENTIALS

JAVA EXCELLENCE

ABOUT JAVA COMMUNITY







WELCOME TO

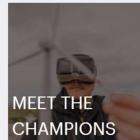
JAVA (Your) COMMUNITY

Welcome to the home of the Java Community in Accenture. In this site you will find all the information relevant for doing Java work in Accenture: from collaboration capabilities provided by the Stream to developer resources. We do wish you make this your own home and contribute to make Java greater than ever! Sincerely yours, the Java Community Leads and Champions.













Navegação

Ctrl + Shift + R	Dialog de pesquisa para recursos, ex: arquivos de texto, classes java, arquivos html, etc.
Ctrl + Shift + T	Dialog de pesquisa por tipo
Ctrl + E	Dialog de pesquisa para selecionar algum arquivo atualmente aberto
Ctrl + F8	Atalho para trocar de pespectiva

Executandoprogramas

Ctrl + F11	Executar
Alt + Shift + X, J	Executa classe selecionada como Aplicação Java

Pesquisa no editor

F3	Abre o editor para o elemento selecionado (TIPO) ou navega para a declaração da variavel selecionada.
Ctrl + .	Vai para o próximo problema / error
Ctrl + ,	Vai para o problem / error anterior
F4 na variável	Exibe a Hierarquia do tipo
Ctrl + K	Procura pelo texto selecionado ou se nada estiver selecionado procura pela ultima pesquisa do Find dialog.
Ctrl + Shift + G	No editor java, pesquisa por referencias no workspace
Ctrl + Shift + P	Seleciona o { ou } do seu codigo java. O cursor precisa estar entre as "{}" (Chaves).

Navegação entre editores

Alt + ←	Vai para o editor anterior. O cursor é colocado no local onde esta antes de abrir o novo editor.
Alt + →	Similar ao Alt + ← mas abre o próximo editor
Ctrl + Q	Vai para o editor e posição do ultimo editor que foi feito alteções.
Ctrl + PageUp	Vai para o editor anterior aberto
Ctrl + PageDown	Vai para o próximo editor aberto

Ctrl + 1	Correção rápida; resultado depende da posição do cursor.
Ctrl + Space	completa o código que você esta digitando.
Ctrl + T	Exibe a hierarquia da classe java ou metodo.
Ctrl + O	Exibe todos os metodos da classe atual, precione Ctrl + O novamente para exibir a hierarquia dos metodos.
Ctrl + M	Maximiza o editor ativo
Ctrl + Shift + F	Formata o código fonte.
Ctrl + I	Corrige a identação, Ex. formata tabs/espaços no código
Ctrl + F	Abre o dialogo de pesquisa
Ctrl + Shift + O	Organiza os imports; adiciona as instruções de imports faltando and remove os não usados.

Copiar e mover linhas

Ctrl + Alt + ↓ ou Ctrl + Alt + ↑	Copia a linha ou seleção em que o cursor está localizado.
Alt + Up ou Alt + Down	Move a linha ou seleção para cima ou para baixo

Delete

Ctrl + D	Deleta linha ou seleção
Ctrl + Shift + DEL	Deleta a linha do lado direito do cursor
Ctrl + DEL	Deleta o próximo elemento
Ctrl + BACKSPACE	Deleta o elemento anterior

Shift + Enter	Adiciona uma linha em branco abaixo da linha atual e move o cursor para a nova linha. A diferença entre o ENTER e esse é que a linha atual não é alterada, independentimente da posição do cursor.
Ctrl+Shift+Enter	Mesmo que o Shift + Enter mas para cima.

Codificação

Shift + F2	Exibe o Javadoc para o tipo / class / metodo selecionado
Alt+Shift + N	Atalho para o menu para criar novos objetos
Alt + Shift + Z	Criar um block com try e catch no código selecionado

Questions and Comments

 What questions or comments do you have?



