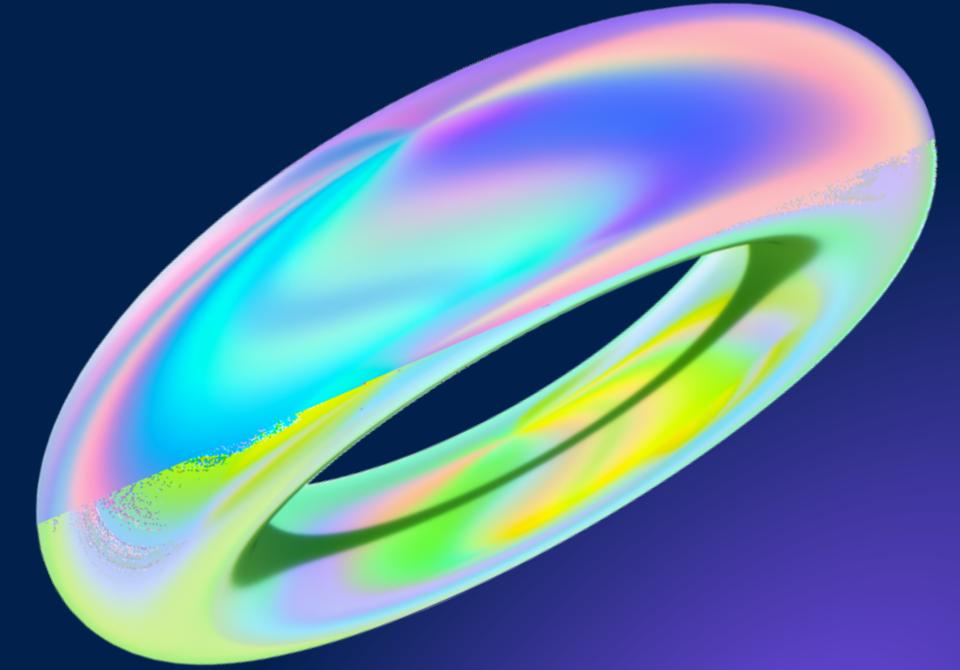
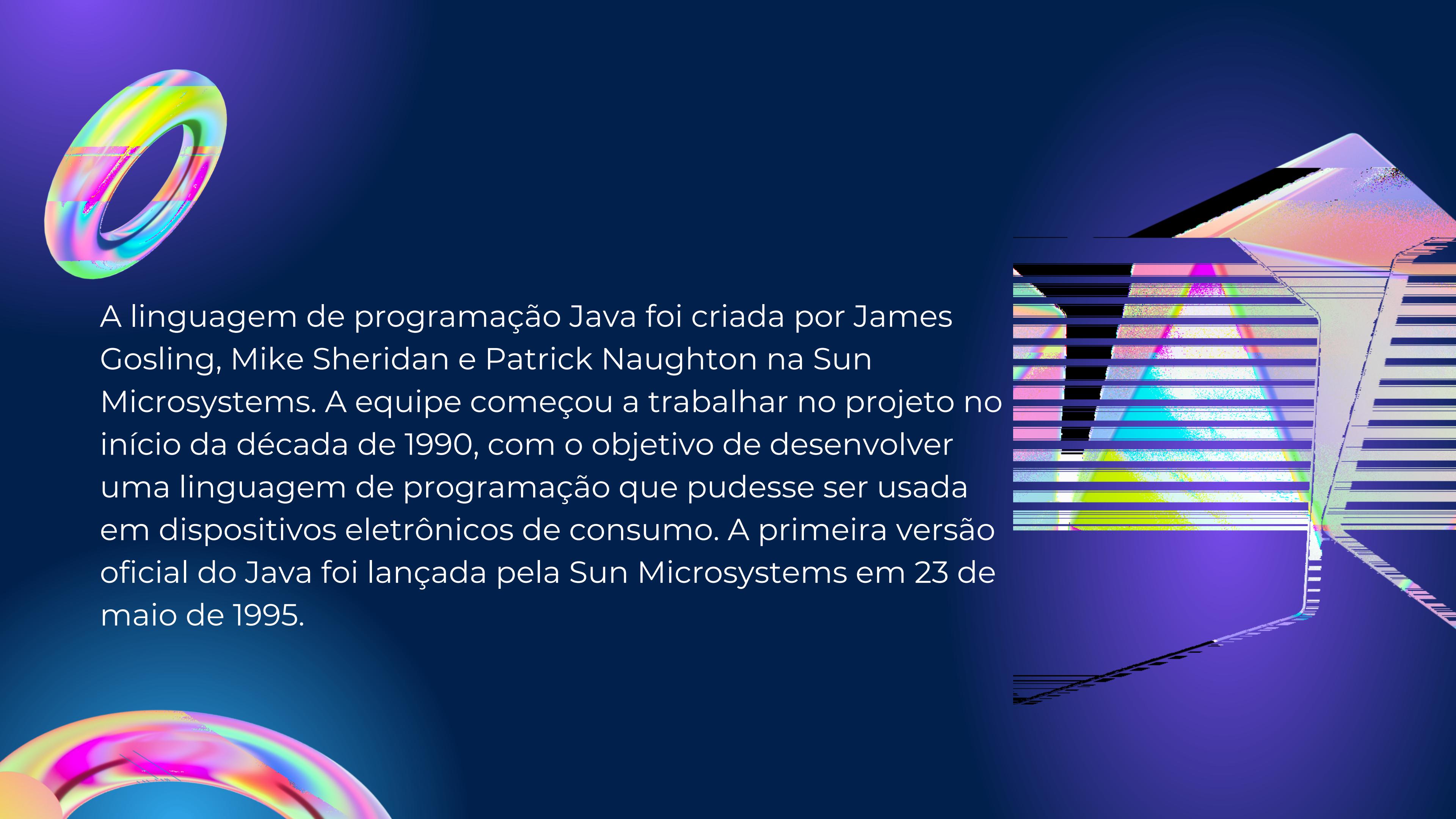


ESTUDO DO JAVA



COMPILE



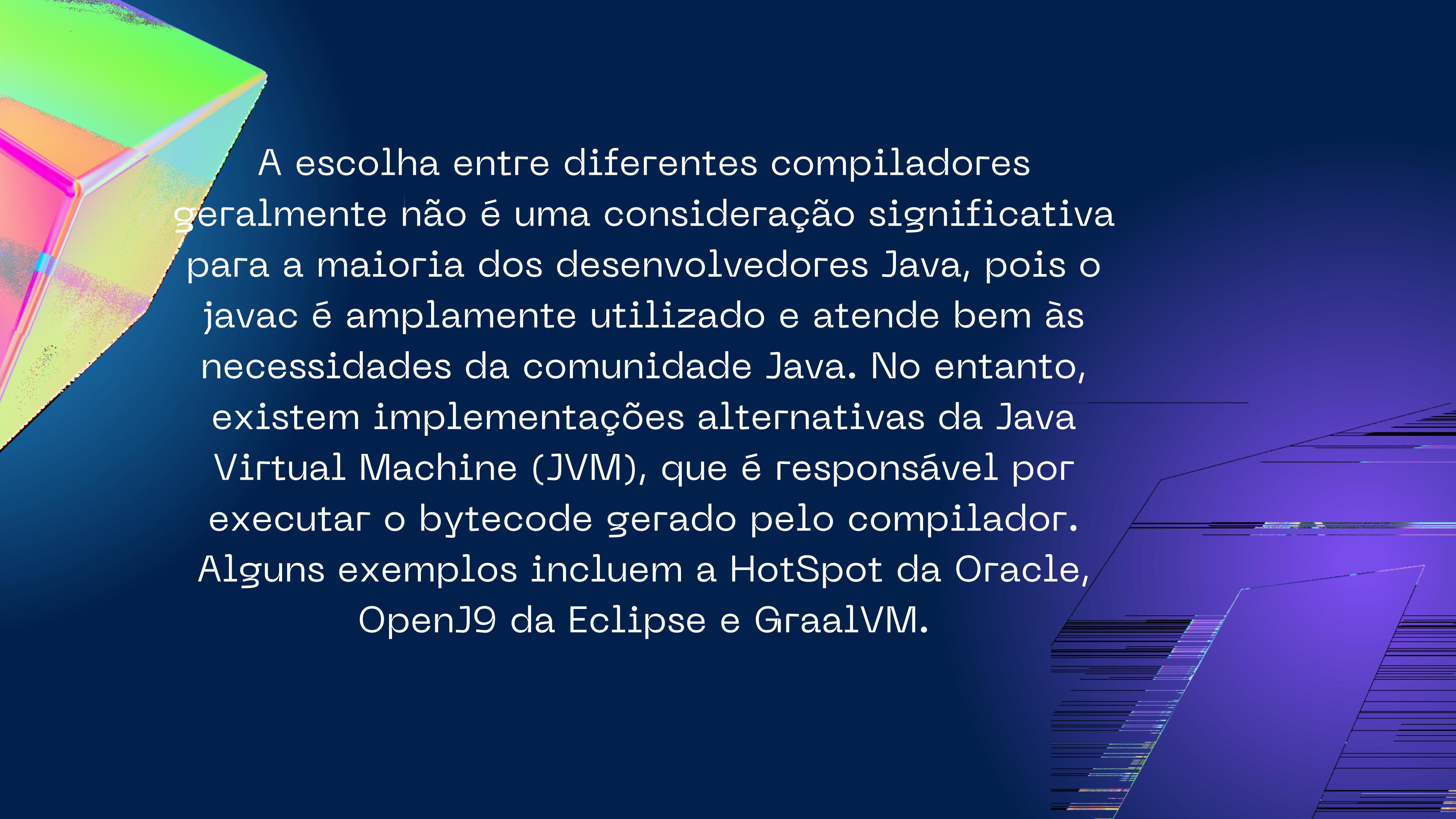


A linguagem de programação Java foi criada por James Gosling, Mike Sheridan e Patrick Naughton na Sun Microsystems. A equipe começou a trabalhar no projeto no início da década de 1990, com o objetivo de desenvolver uma linguagem de programação que pudesse ser usada em dispositivos eletrônicos de consumo. A primeira versão oficial do Java foi lançada pela Sun Microsystems em 23 de maio de 1995.



O compilador principal utilizado na linguagem Java é o javac, que faz parte do Kit de Desenvolvimento Java (JDK - Java Development Kit). O JDK é distribuído pela Oracle. O Java foi lançado oficialmente em 1995, e o javac tem sido uma parte essencial do ambiente de desenvolvimento Java desde então.

É importante observar que, além do compilador, o Java também depende fortemente da Java Virtual Machine (JVM) para a execução dos programas. A JVM é responsável por interpretar ou compilar just-in-time (JIT) o bytecode gerado pelo compilador Java. Diferentes implementações da JVM podem ser usadas, como a HotSpot da Oracle, OpenJ9 da Eclipse, entre outras. Estas implementações contribuem para a execução eficiente dos programas Java em diferentes plataformas.



A escolha entre diferentes compiladores geralmente não é uma consideração significativa para a maioria dos desenvolvedores Java, pois o javac é amplamente utilizado e atende bem às necessidades da comunidade Java. No entanto, existem implementações alternativas da Java Virtual Machine (JVM), que é responsável por executar o bytecode gerado pelo compilador. Alguns exemplos incluem a HotSpot da Oracle, OpenJ9 da Eclipse e GraalVM.

Existem várias IDEs populares para o desenvolvimento em Java, cada uma com suas vantagens e desvantagens. Abaixo estão algumas das IDEs mais conhecidas para Java:

Eclipse:

- **Vantagens:**
 - **Ampla Comunidade:** O Eclipse tem uma comunidade grande e ativa, oferecendo suporte e uma ampla gama de plugins.
 - **Versatilidade:** Pode ser estendido para suportar várias linguagens e tecnologias.
- **Desvantagens:**
 - **Curva de Aprendizado:** Alguns usuários podem achar a curva de aprendizado inicial íngreme.

IntelliJ IDEA:

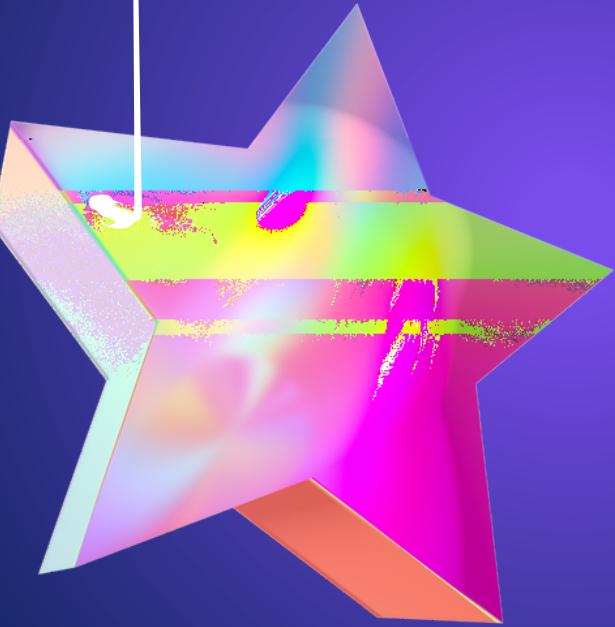
- **Vantagens:**
 - **Excelente Suporte Java:** Reconhecida por seu excelente suporte Java e ferramentas avançadas.
 - **Experiência do Usuário:** Interface de usuário intuitiva e fácil de usar.
- **Desvantagens:**
 - **Licença Paga:** Embora exista uma versão gratuita, algumas funcionalidades avançadas requerem uma licença paga.

Ne Bea

◦ Va a e
• I e açã c Se d e O e ege e açã ác c e d e
Ja a EE e G a F Ve c ee b d a a de e
• S e a Ja aFX Ja aFX a ad a Ec e de e e e e e
◦ De a a e
• Me E e e C a ad a Ec e de e e e e e e
c

JDe e e O ac e

◦ Va a e
• I e açã c Fe a e a O ac e O e ece e açã e ec ea
a a e a e a e ec a a e a O ac e P de e e ade ad a a
• S e a e ec a ad a a S ac O ac e a O ac e
◦ De a a e ec a ad a a S ac O ac e P de e e ade ad a a

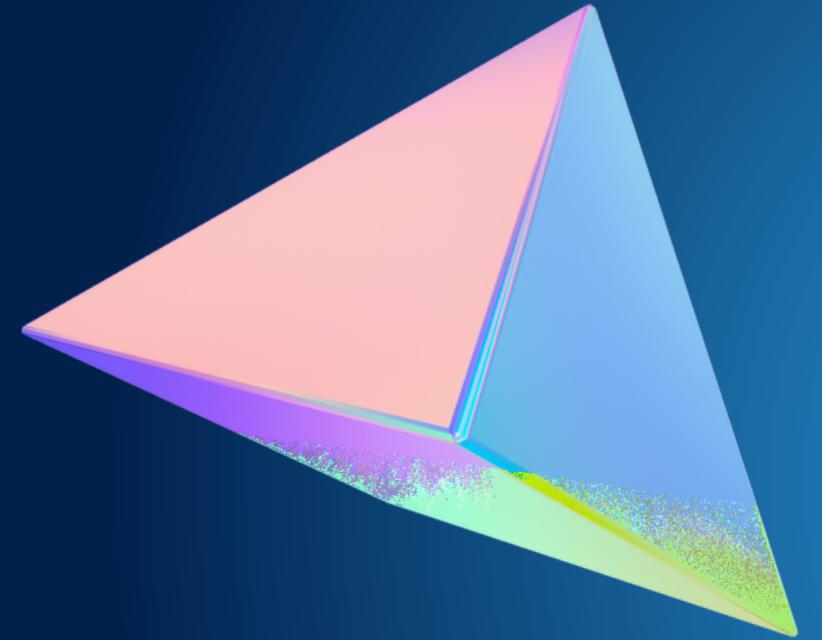


Tipo de dados

- Tipos Primitivos:
- Tipos de Referência
- Além desses tipos básicos, é possível criar tipos de dados personalizados através de classes e interfaces.

Tipos Primitivos:

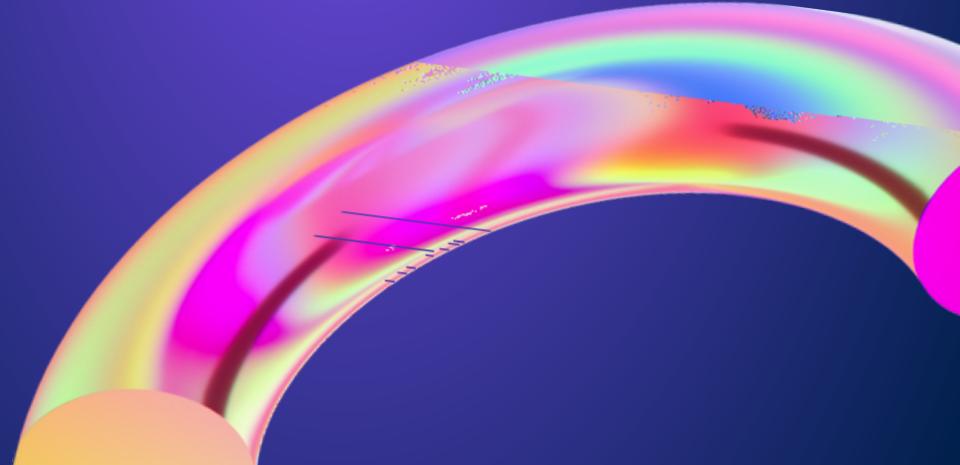
- Numéricos Inteiros:
 - byte: 8 bits, com sinal.
 - short: 16 bits, com sinal.
 - int: 32 bits, com sinal (o mais comum para inteiros).
 - long: 64 bits, com sinal.
- Numéricos de Ponto Flutuante:
 - float: 32 bits, ponto flutuante de precisão simples.
 - double: 64 bits, ponto flutuante de precisão dupla (o mais comum para números decimais).
- Caracteres:
 - char: 16 bits, representando caracteres Unicode.
- Booleano:
 - boolean: Representa valores verdadeiro ou falso.





Tipos de Referência

- Objetos:
 - Todas as classes que você cria ou usa em Java são tipos de referência.
- Arrays:
 - Um tipo especial de objeto que contém uma coleção de elementos do mesmo tipo.



Palavras Reservadas

- abstract; assert; boolean; break ;bytecase; catch; char; class; const; continue; default; do; double; else; enum; extends; final; finally; float; for; goto; if; implements ; import; instanceof; int; interface; long; native; new; package; private; protected; public; return; short; static; strictfp; superswitch; synchronized; this; throw ;throwstransient; try; void; volatile; while.

Estruturas de controlo

As estruturas de controle em Java são mecanismos que permitem alterar o fluxo de execução do programa.

Estruturas Condicionais (if else; switch case)

Estruturas de Repetição (for; while; do-while)

Estruturas de Controle de Transferência

Variáveis compostas homogenias e heterogenias

- **Variáveis Compostas Homogêneas:**

- **Arrys (arrays):** Um array é uma estrutura de gênero que armazena elementos do mesmo tipo, com uma única classe e estrutura.

- **Listas Homogêneas:** Classes que implementam a interface `List` por exemplo `ArrayList` e `LinkedList` em que podem armazenar elementos de um só tipo.

- **Variáveis Compostas Heterogêneas:**

- **Objetos (Classes):** Um classe é uma estrutura que cria objetos de gênero e armazena e modifica estes.

- **Mapeamento:** Estrutura de dados que armazena aplicações entre tipos diferentes.

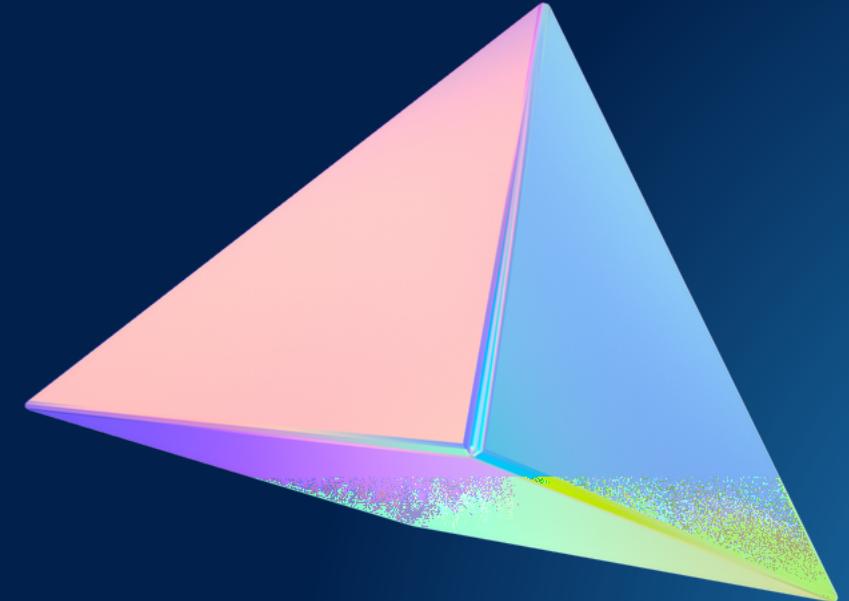
Função e Procedimento

- Em Java, o termo "procedimento" não é usado da mesma forma que em algumas outras linguagens de programação, como Pascal. Em Java, a ênfase recai mais sobre "métodos" e "funções". Ambos os termos referem-se a blocos de código que realizam uma tarefa específica, mas há uma diferença sutil entre eles.
- Métodos:
- Em Java, um método é uma função que está associada a uma classe ou objeto.

Classe em Java

Em Java, uma classe é uma estrutura fundamental que é usada para organizar e modelar objetos. Uma classe define o comportamento e as propriedades de objetos específicos que são criados a partir dela.

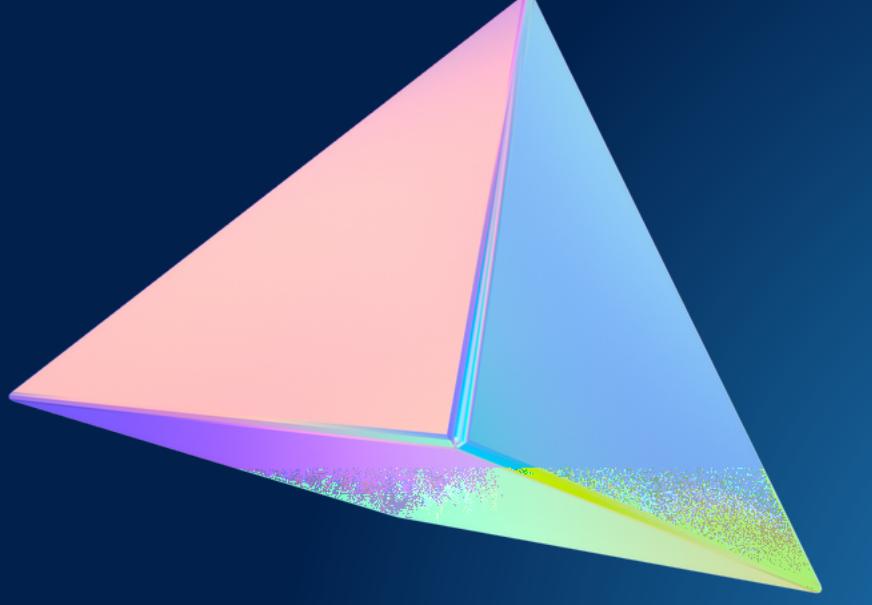
A nível académico



A linguagem Java é amplamente utilizada no meio acadêmico e em várias instituições de ensino.

1. **Orientação a Objetos:** Java é uma linguagem orientada a objetos, e muitos cursos acadêmicos ensinam conceitos fundamentais de programação orientada a objetos (OOP) usando Java como exemplo.
2. **Ampla Comunidade e Recursos de Aprendizado:** Java possui uma grande comunidade de desenvolvedores e uma abundância de recursos de aprendizado, incluindo tutoriais, documentação oficial e exemplos de código. Isso facilita para os estudantes encontrar suporte e recursos educacionais.
3. **Uso Empresarial:** Java é amplamente utilizado na indústria, especialmente em desenvolvimento empresarial. Os estudantes muitas vezes veem a aprendizagem de Java como uma preparação para carreiras no setor de tecnologia.

A nível empresarial



Java é uma escolha popular para o desenvolvimento empresarial devido à sua portabilidade, orientação a objetos, ecossistema extenso, desempenho sólido e suporte a tecnologias modernas.



Grandes ganhos da Lingagem

Java é uma linguagem de programação amplamente utilizada em uma variedade de domínios, e vários softwares notáveis foram desenvolvidos usando Java.

Alguns dos maiores e mais conhecidos softwares feitos com Java incluem

1. Sistemas de Gerenciamento de Banco de Dados:

- Oracle Database: O banco de dados Oracle é um dos mais usados no mundo empresarial e muitas de suas ferramentas são desenvolvidas em Java.

2. Aplicações Empresariais e Portais:

- SAP NetWeaver: Uma plataforma de integração e aplicação empresarial da SAP, muitas de suas partes são implementadas em Java.
- Liferay: Um sistema de gerenciamento de portais empresariais construído em Java.

3. Servidores de Aplicativos:

- Apache Tomcat: Um servidor web e de servlets open-source implementado em Java.
- JBoss/Wildfly: Um servidor de aplicativos Java EE de código aberto.

4. Ferramentas de Desenvolvimento Integrado (IDE):

- Eclipse: Uma popular IDE de código aberto para desenvolvimento em Java e outras linguagens.

5. Aplicações Web e Frameworks:

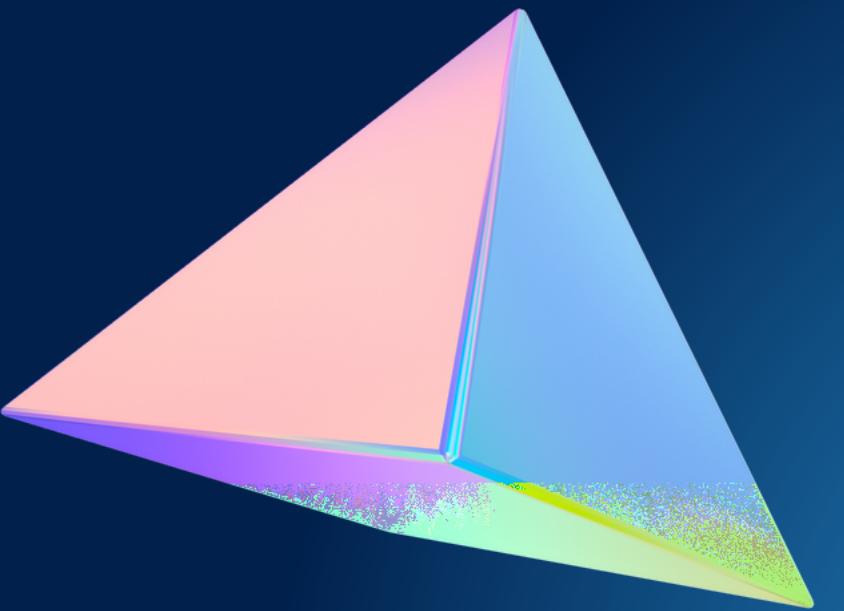
- Spring Framework: Um framework de desenvolvimento de aplicações Java que é amplamente utilizado na indústria.
- Apache Struts: Um framework MVC para construção de aplicações web.

Java em Video Jogos

Java é amplamente utilizado por empresas independentes de desenvolvimento de jogos e para a criação de jogos móveis, dada a versatilidade da linguagem e sua rica coleção de material de código aberto. Muitos dos melhores jogos para celular do mundo foram desenvolvidos em Java. Minecraft, Mission Impossible III e Asphalt 6 são apenas alguns nomes populares com os quais você provavelmente está familiarizado.

Por que recomendar?

- Há vários motivos pelos quais alguém pode recomendar o uso do Java em determinados contextos. Aqui estão alguns dos motivos mais comuns:
- **Plataforma Independente:** O Java é conhecido por ser uma linguagem de programação "write once, run anywhere" (escreva uma vez, execute em qualquer lugar). Isso significa que o código Java pode ser executado em diversas plataformas sem a necessidade de recompilação, o que facilita a portabilidade.
- **Comunidade Ativa e Ecossistema Robusto:** Java possui uma comunidade de desenvolvedores muito ativa, com uma vasta quantidade de bibliotecas, frameworks e ferramentas disponíveis. Isso facilita o desenvolvimento e a manutenção de projetos, pois muitas soluções já foram implementadas e testadas pela comunidade.
- **Segurança:** A Máquina Virtual Java (JVM) inclui recursos robustos de segurança, o que é crucial em muitos ambientes de desenvolvimento. A arquitetura da JVM ajuda a proteger contra várias ameaças, como ataques de buffer overflow.
- **Desenvolvimento Web e Empresarial:** Java é amplamente utilizado no desenvolvimento web e empresarial, especialmente em grandes sistemas corporativos. Estruturas como Spring tornam o desenvolvimento mais eficiente e escalável.
-

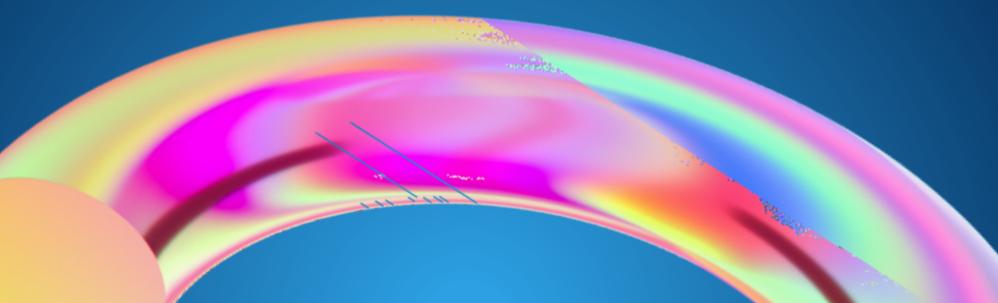


• D J S A A
• H L D J A I J
• R O O J
• D E J I A C
J M



Impacto do java na informatica

o Java teve um impacto profundo na informática, influenciando práticas de desenvolvimento, arquiteturas de software e facilitando a criação de aplicações escaláveis e portáteis em uma variedade de setores.





Спасибо!

compiladores

prof. Andre' Filemon

EINF6_M2

Eliane Watele