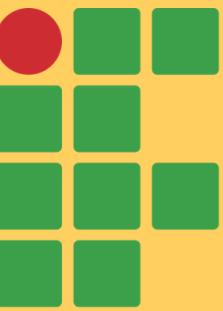


Linguagem de programação Java

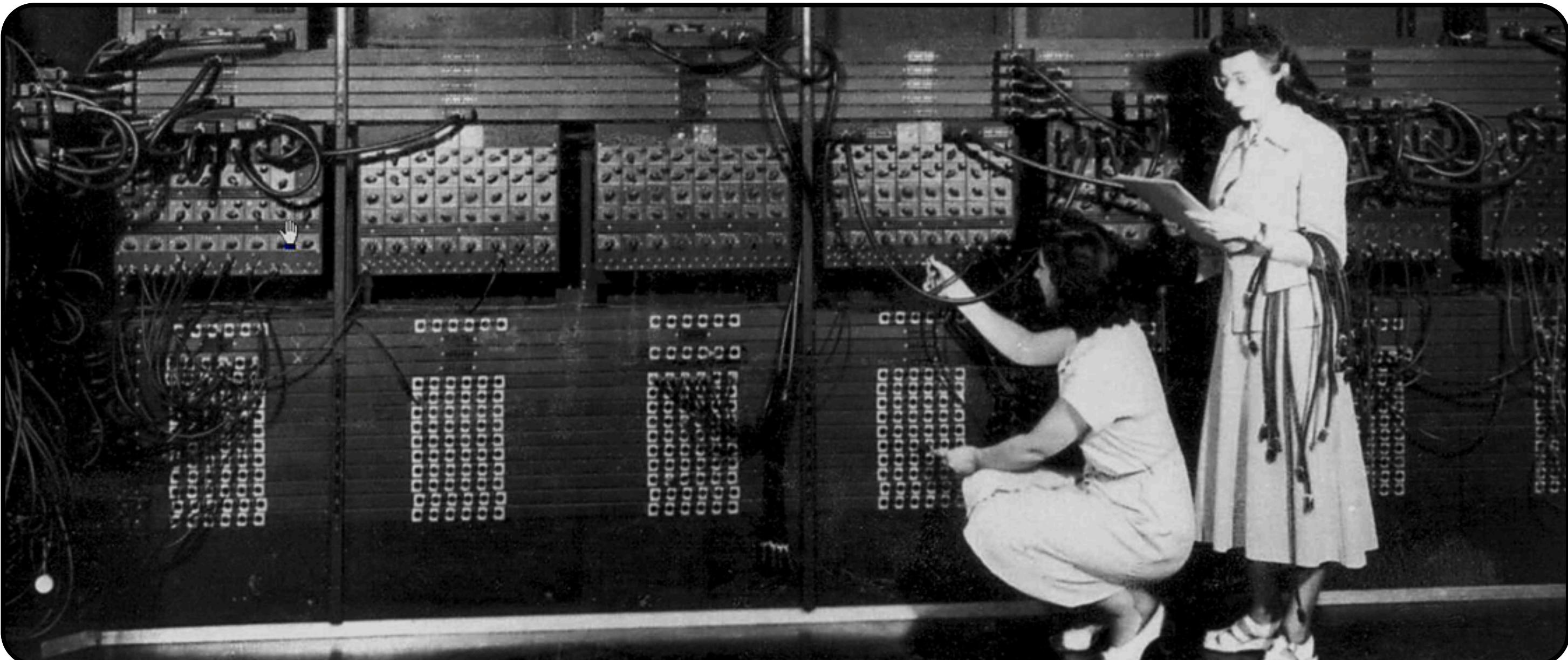
Orientação
a Objetos

Professor
Me. Jonas Pontes

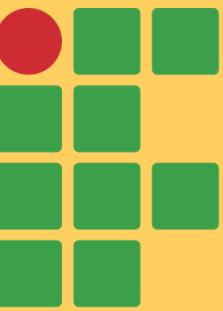




História dos computadores



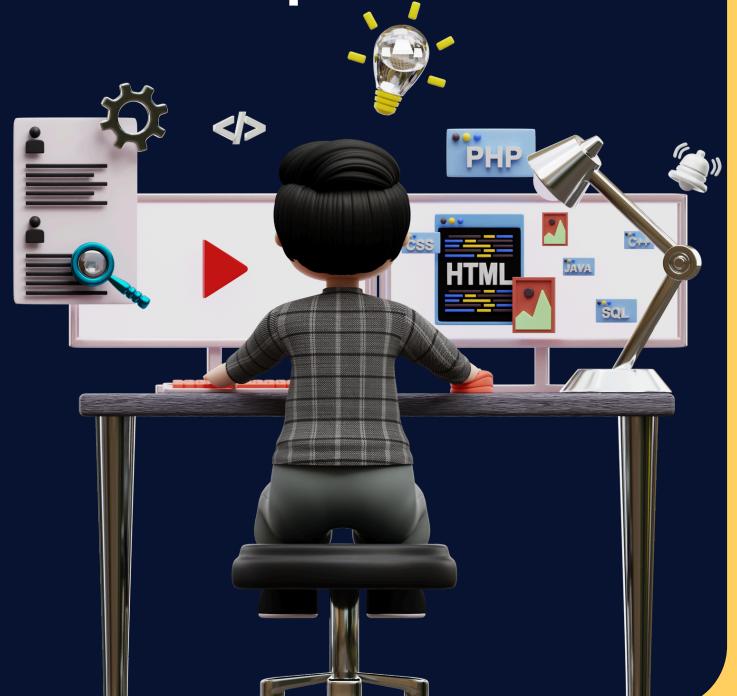
Como programar?

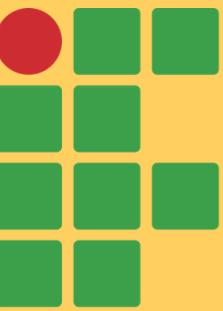


Tipos de linguagens

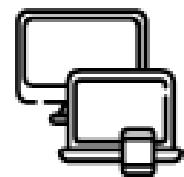


Programadores escrevem instruções em várias linguagens de programação, algumas diretamente compreensíveis por computadores e outros requerem passos intermediários de tradução.





Tipos de linguagens



Tipos de linguagens

Máquina

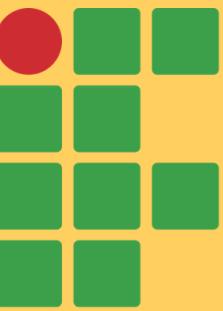


Assembly



Alto nível





Tipos de linguagens

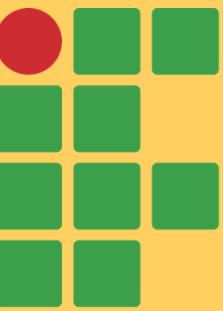


Linguagens de máquina: strings de números que instruem os computadores a realizar suas operações mais elementares;

Linguagem assembly: abreviações para representar operações elementares são a base de assembly;

Linguagens de alto nível: instruções únicas podem ser escritas para realizar tarefas substanciais.



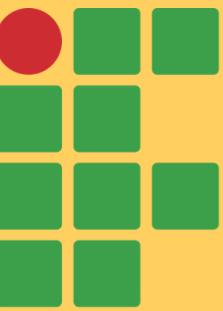


Tipos de linguagens



```
program Exemplo;
var
  numero1, numero2, resultado: integer;
begin
  write('Digite o primeiro número: ');
  readln(numero1);
  write('Digite o segundo número: ');
  readln(numero2);
  resultado := numero1 + numero2;
  writeln('Resultado ', resultado);
  readln;
end.
```





Tradutores

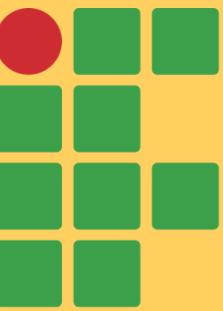


Assemblers: tradutores com o propósito de converter programas de linguagem assembly em linguagem de máquina de forma muito rápida;

Compiladores: tradutores que convertem programas escritos em alto nível em linguagem de máquina;

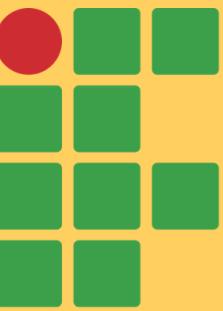
Interpretadores: executam código-fonte sem a necessidade de compilação prévia, convertendo as instruções para LM em tempo de execução.



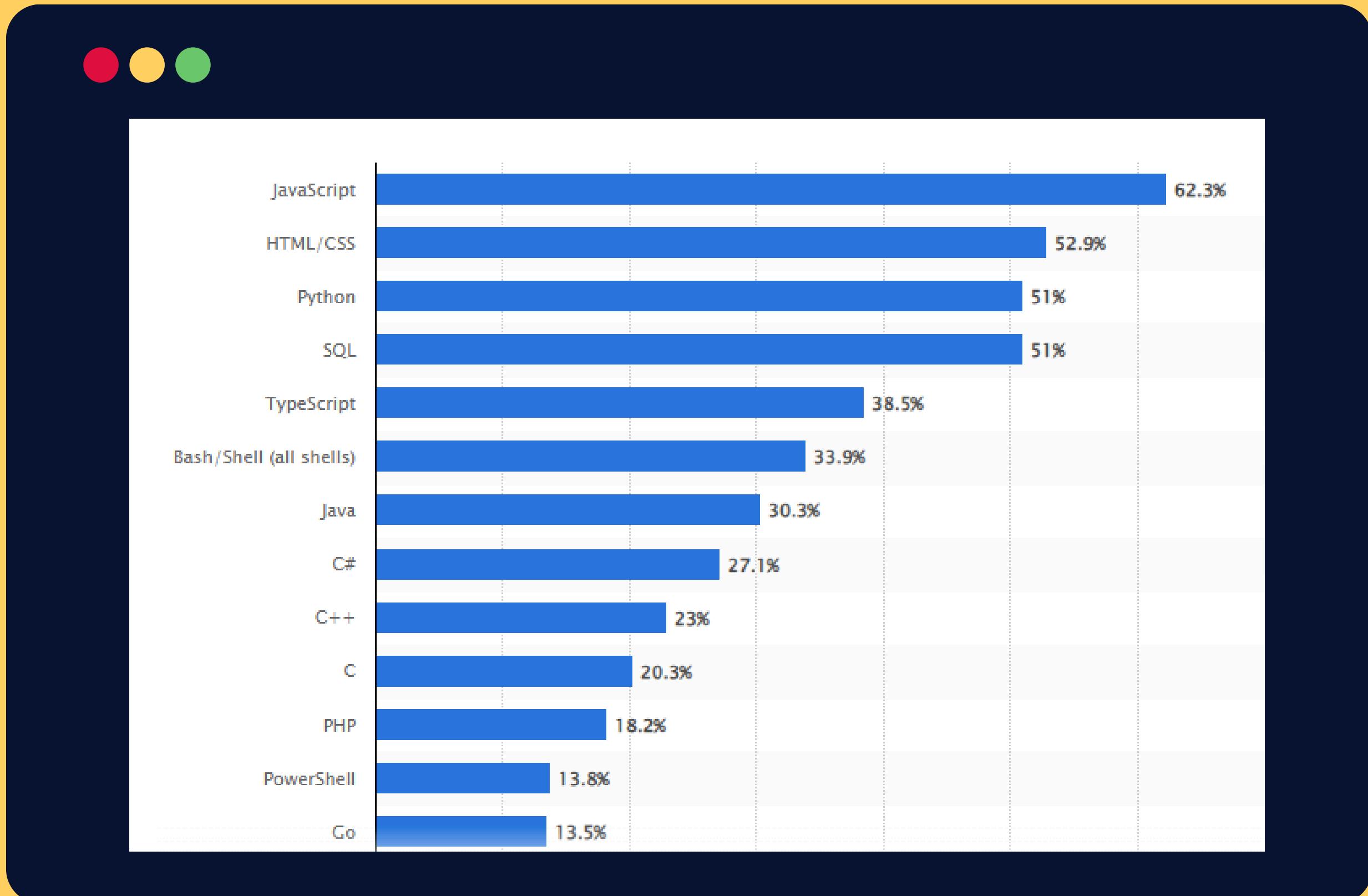


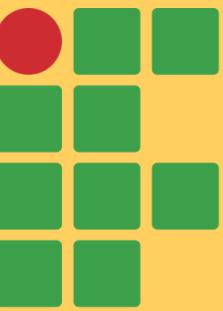
A linguagem de programação Java



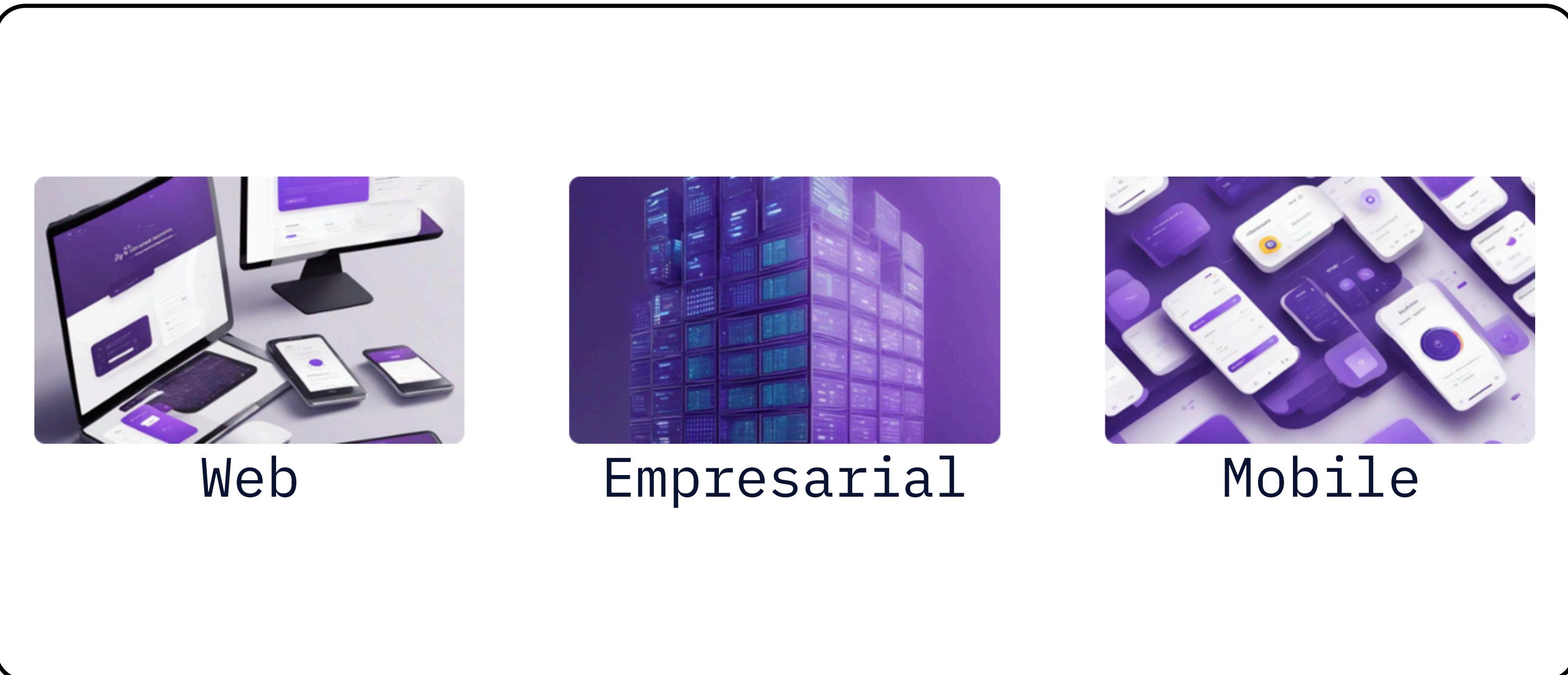


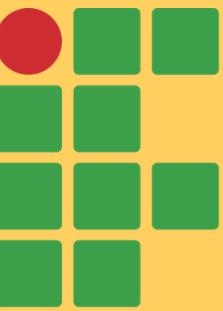
LP mais usadas em 2024





Exemplos de Aplicação do Java





Por que estudar Java?

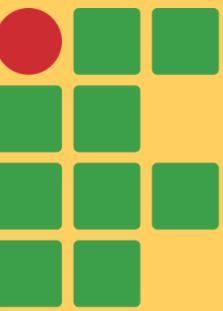


Relevância contínua: mantém-se amplamente adotada na indústria de software e infraestrutura de tecnologia;

Versatilidade e robustez: capaz de lidar com variedades de cenários de desenvolvimento e demandas;

Retorno: aprendizado e domínio dessa tecnologia é um investimento valioso para profissionais de TI.





Java



1

Criação por Sun Microsystems

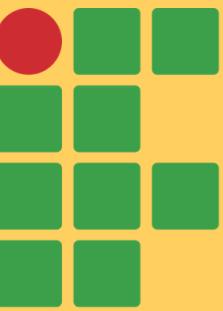
2

Lançamento oficial em 1995

O impacto e a recepção da linguagem após seu lançamento

3

Aquisição pela Oracle



Java

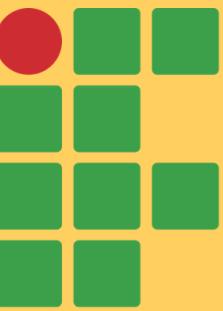


Sum Microsystems: desejava ingressar no mercado de dispositivos inteligentes;

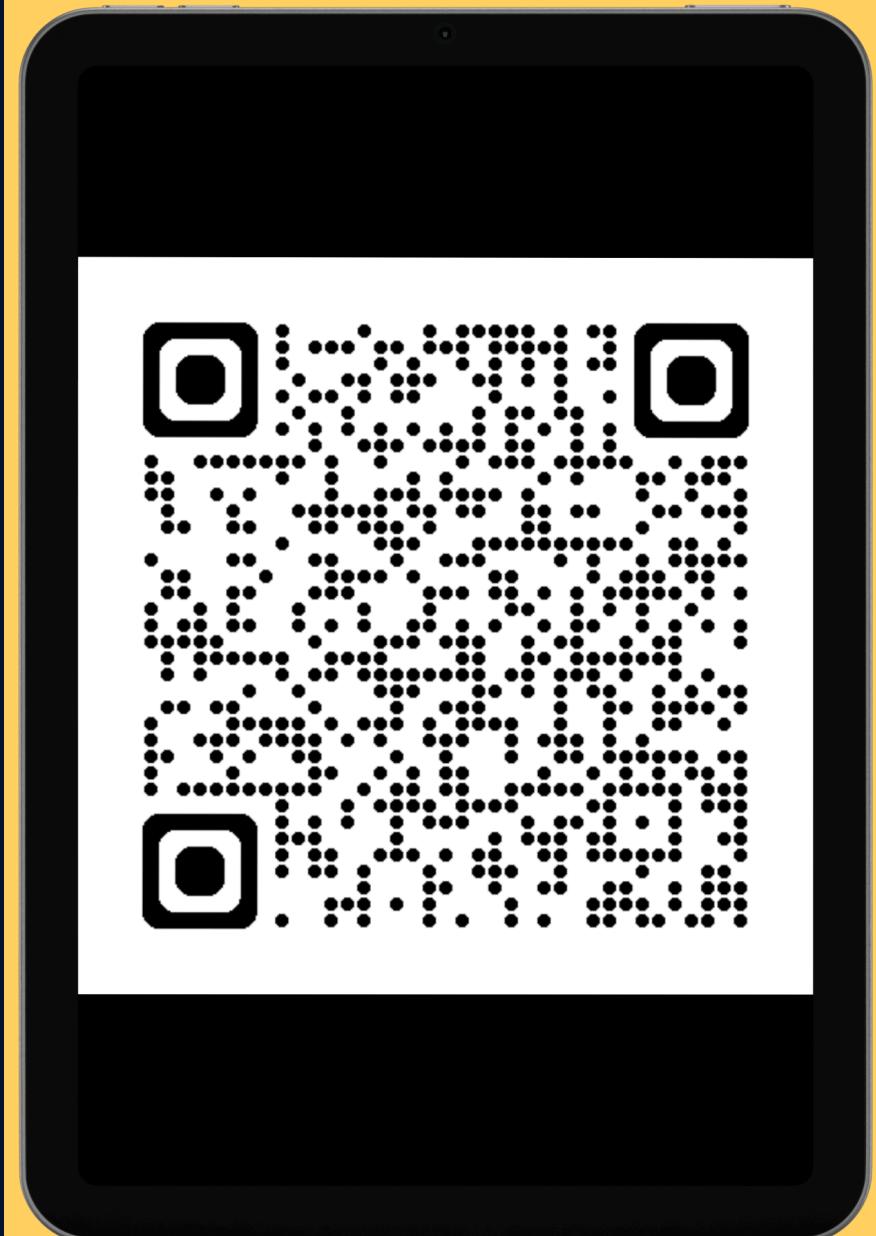
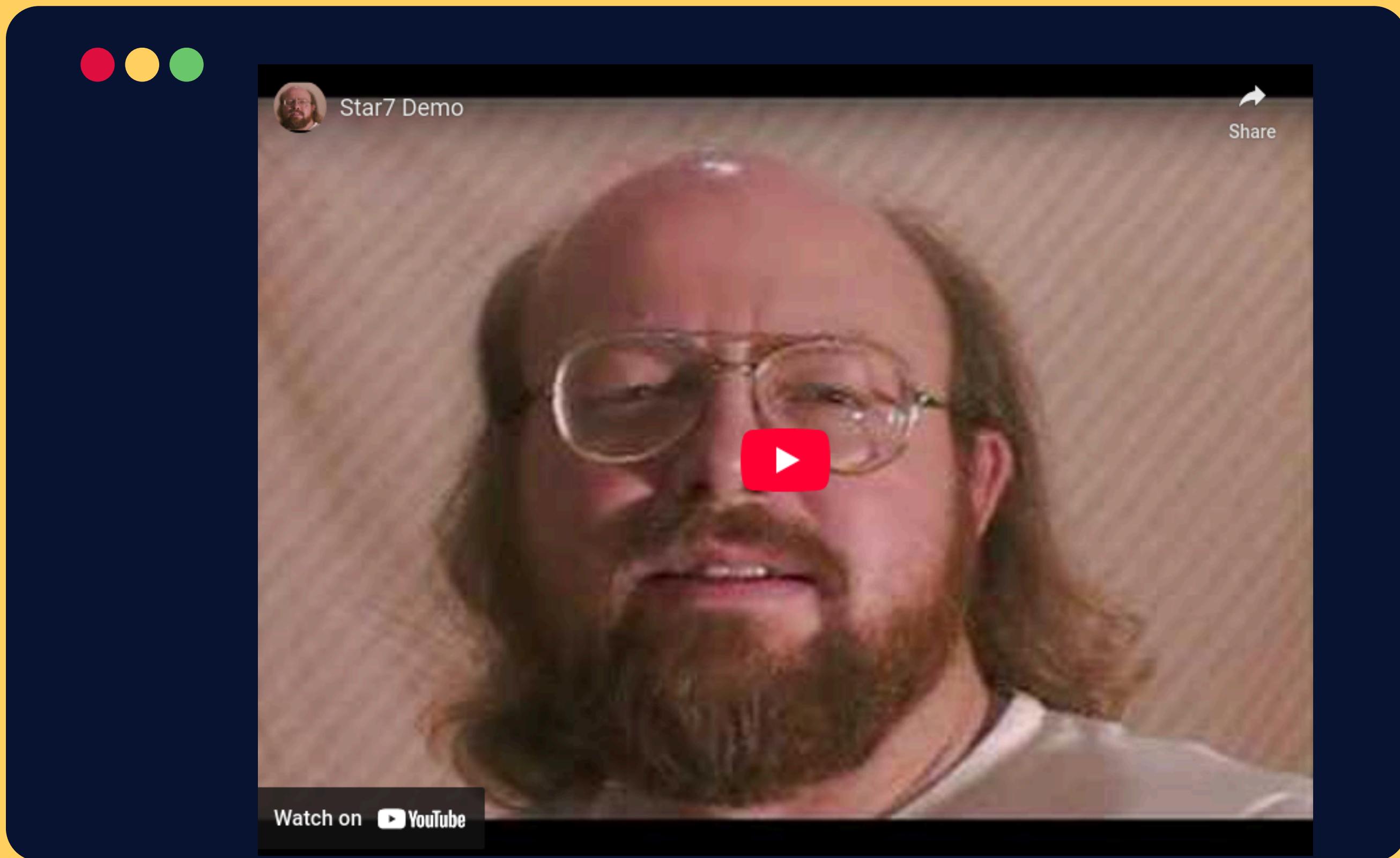
Green Project: lançou o Star7, controle remoto com touchscreen, capaz de movimentar o Duke em equipamentos como televisores;

James Gosling: líder de desenvolvimento da Oak, a linguagem para o *7



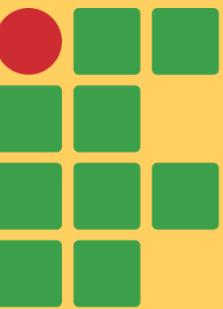


Java



www.youtube.com/watch?v=1CsTH9S79qI

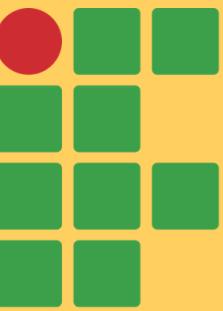
Desenvolvimento do Java



Estouro da internet: estabelecimento da Web, uma rede interativa;

Visão da Sun: utilizar o Java para adicionar conteúdo dinâmico a páginas da Web;

Impacto: a linguagem chamou a atenção da comunidade de negócios por causa do enorme interesse pela Web



Aquisição pela Oracle



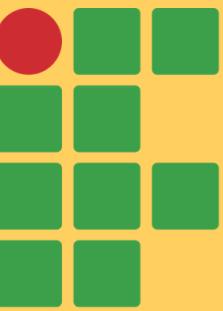
2009: Oracle comprou a Sun, dona da marca;

Oracle: a empresa sempre investiu e fez negócios através do uso da plataforma Java;

2011: lançamento do Java 7, com algumas pequenas mudanças na linguagem;

Versão atual: Java 24 (23 LTS);

Licença: gratuito inclusive para fins comerciais, desde que seja para computação de uso geral.



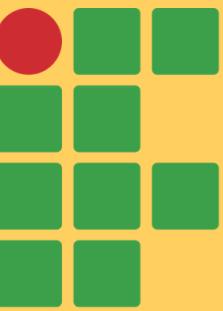
Edições do Java



Java ME: Java Micro Edition – dispositivos embarcados e móveis (IoT);

Java SE: Java Standard Edition – código portátil para desktops e servidores;

Java EE: Java Enterprise Edition – aplicações corporativas.



Vantagens do Java



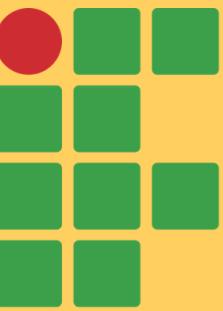
Portabilidade: capacidade de ser executada em diferentes plataformas;

OOP: abordagem modular, flexível e reutilizável para concepção e manutenção de software;

Comunidade: comunidade de desenvolvedores grande e ativa (muitos materiais online e gratuitos);

Segurança: mecanismos integrados para proteger os dados e sistemas.





Desvantagens do Java

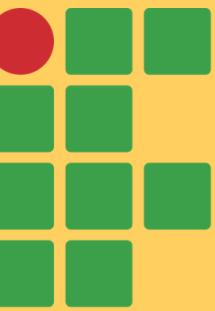


Complexidade: na aprendizagem e faz uso excessivo de bibliotecas;

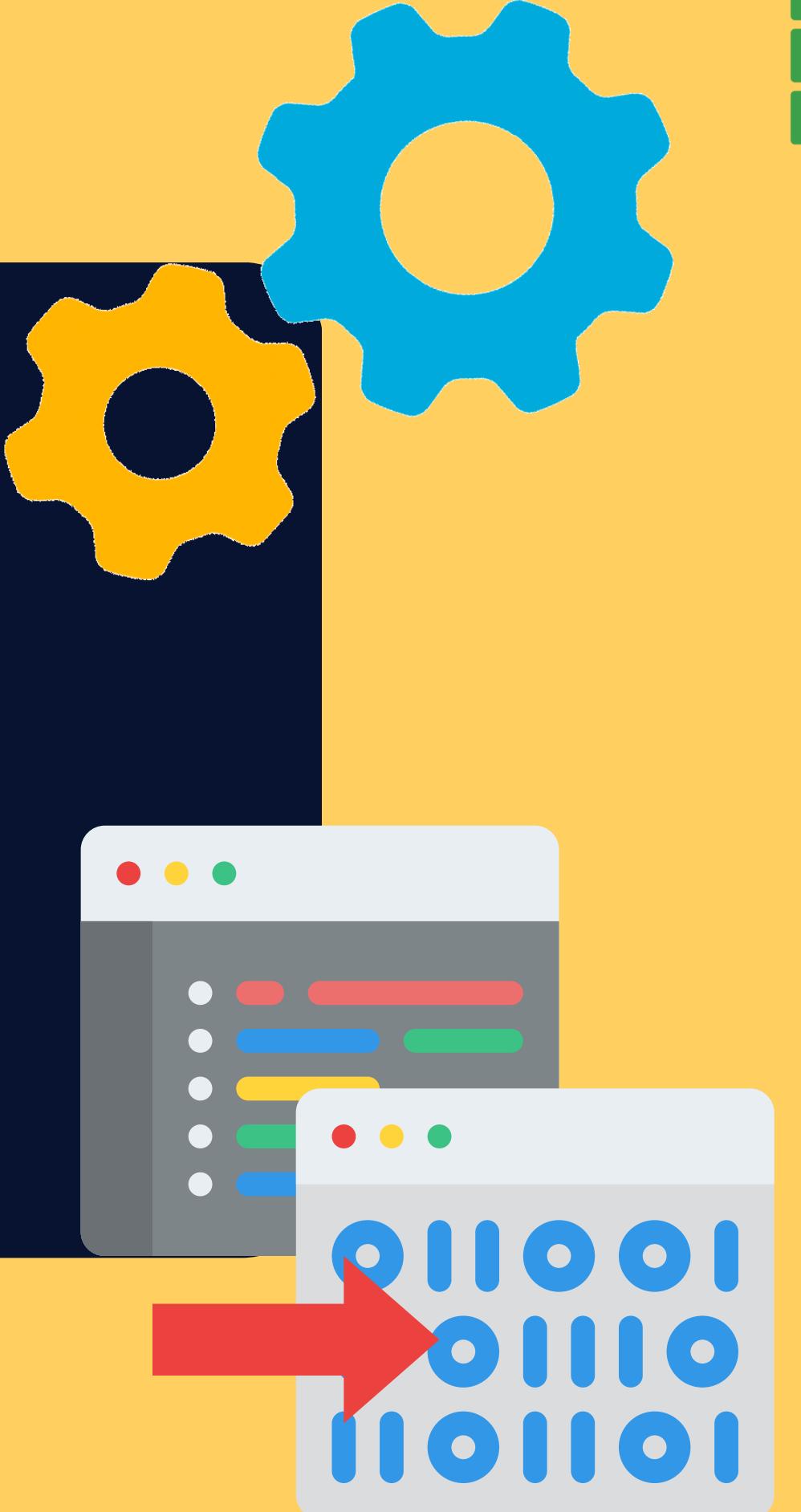
Memória: JVM consume muita memória;

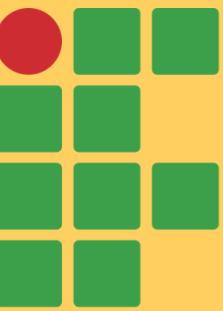
Lentidão na inicialização: em razão da camada a mais de software em relação a outras linguagens;

Sobrecarga: de tempo de execução em sistemas com recursos limitados.

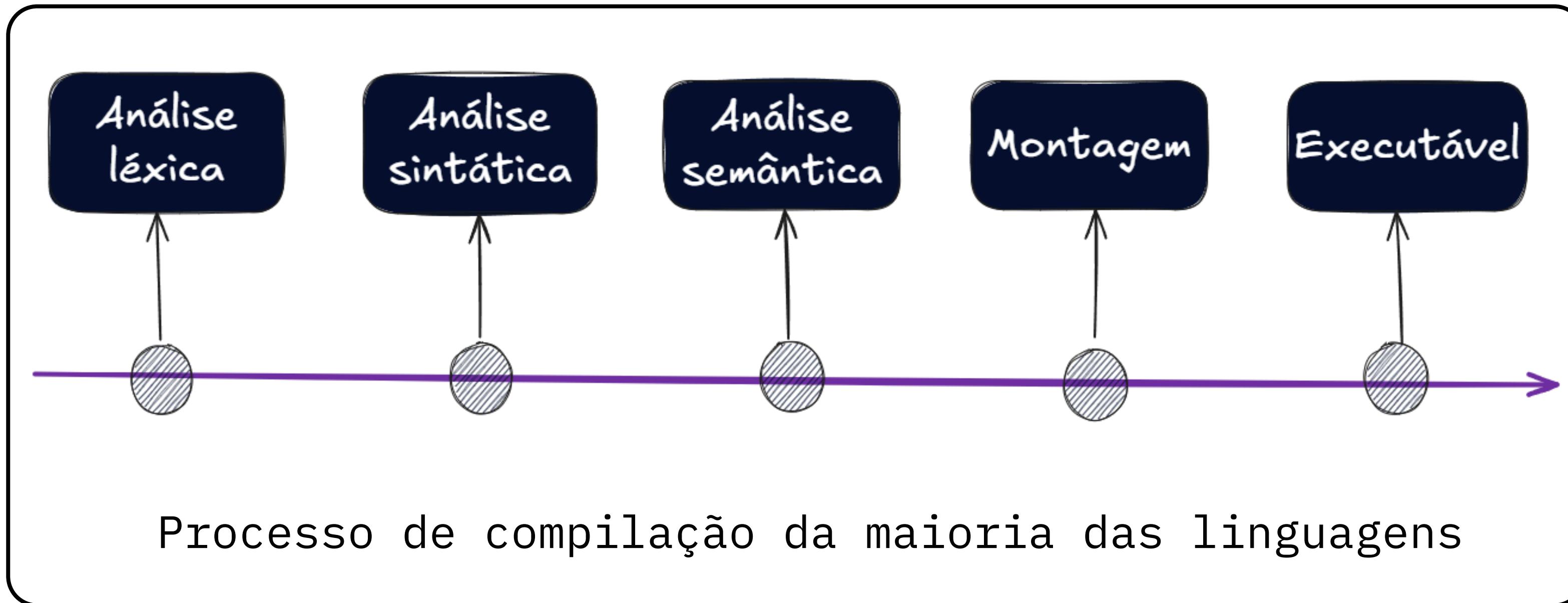


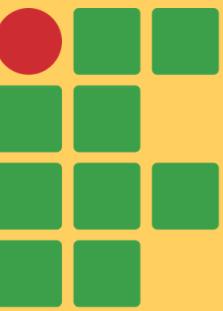
Aspectos técnicos





Compilação e execução





Compilação e execução



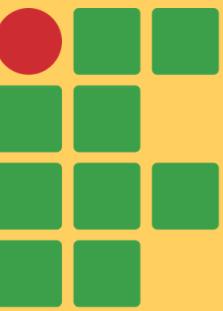
Executável transmite as instruções ao SO



SO aloca e gerencia os recursos
necessários para a execução do programa



Computador executa as instruções
passadas pelo SO



Compilação e execução



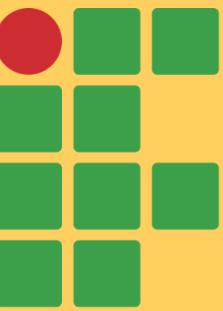
Executável transmite as instruções ao SO



SO aloca e gerencia os recursos
necessários para a execução do programa



Computador executa as instruções
passadas pelo SO



Compilação e execução



Executável transmite as instruções ao SO

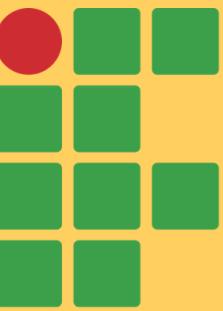


SO aloca e gerencia os recursos
necessários para a execução do programa

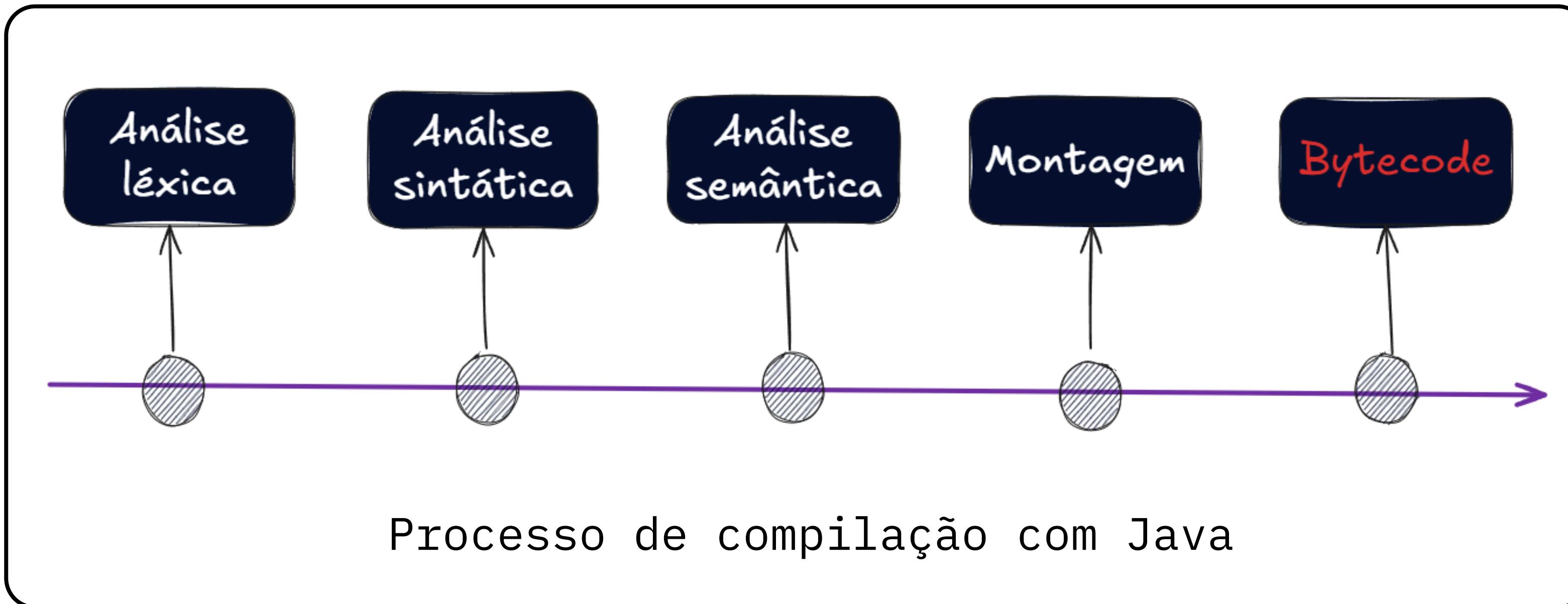


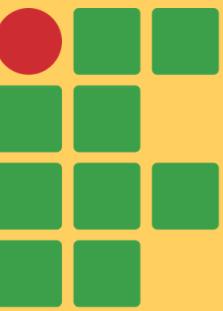
Computador executa as instruções
passadas pelo SO



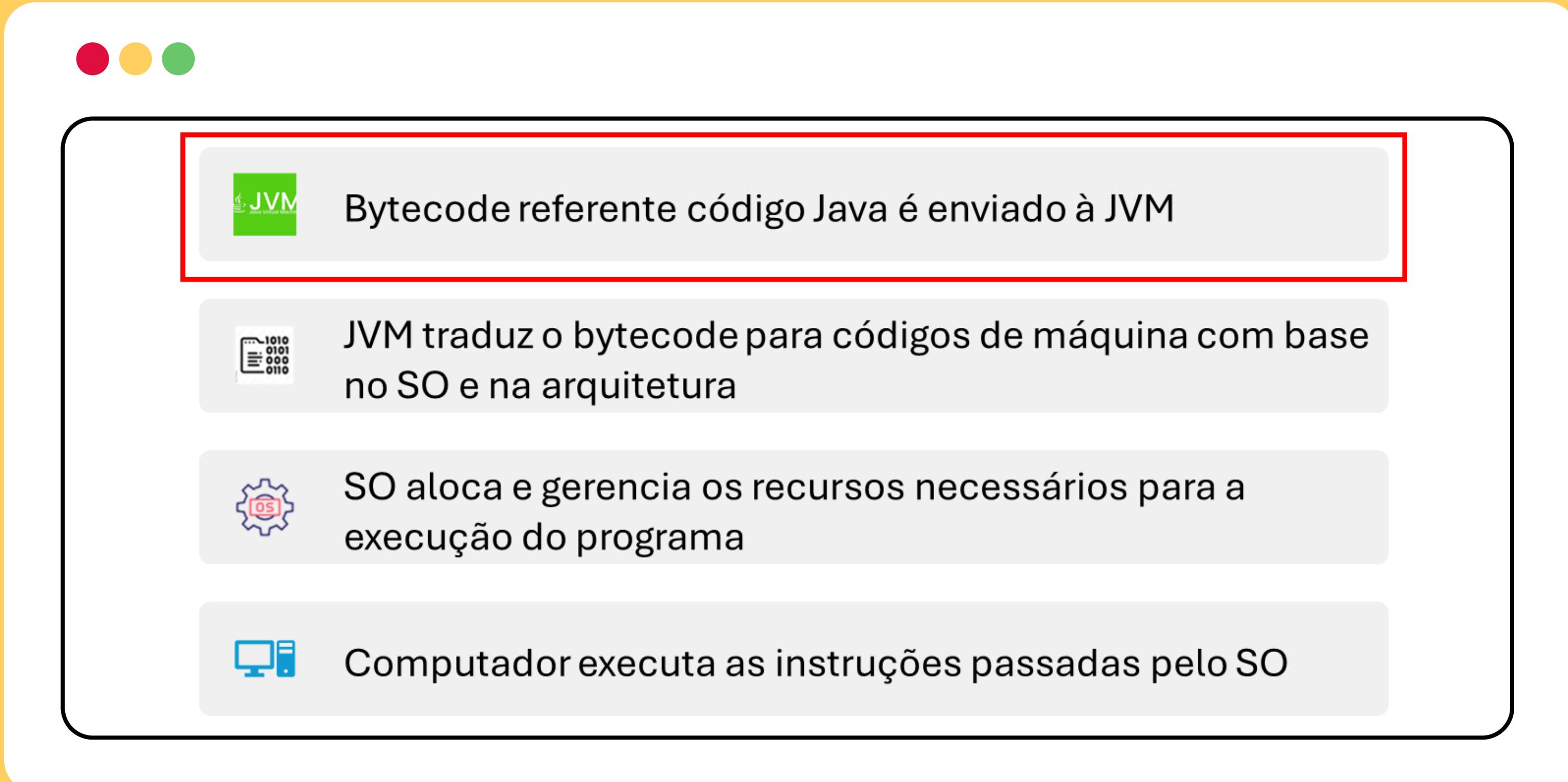


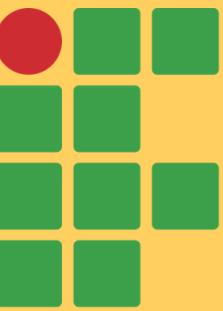
Compilação e execução



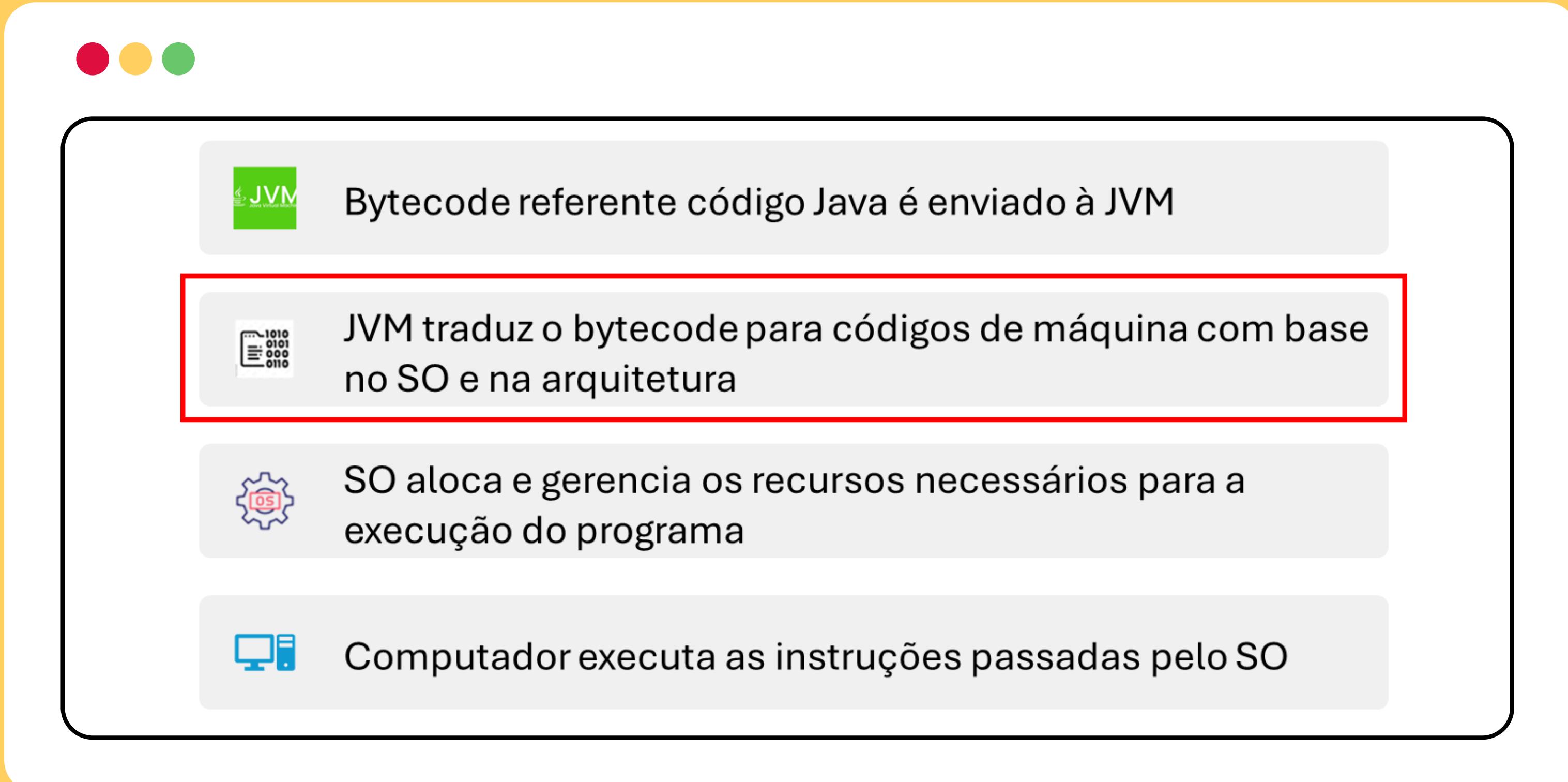


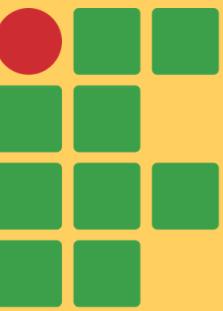
Compilação e execução



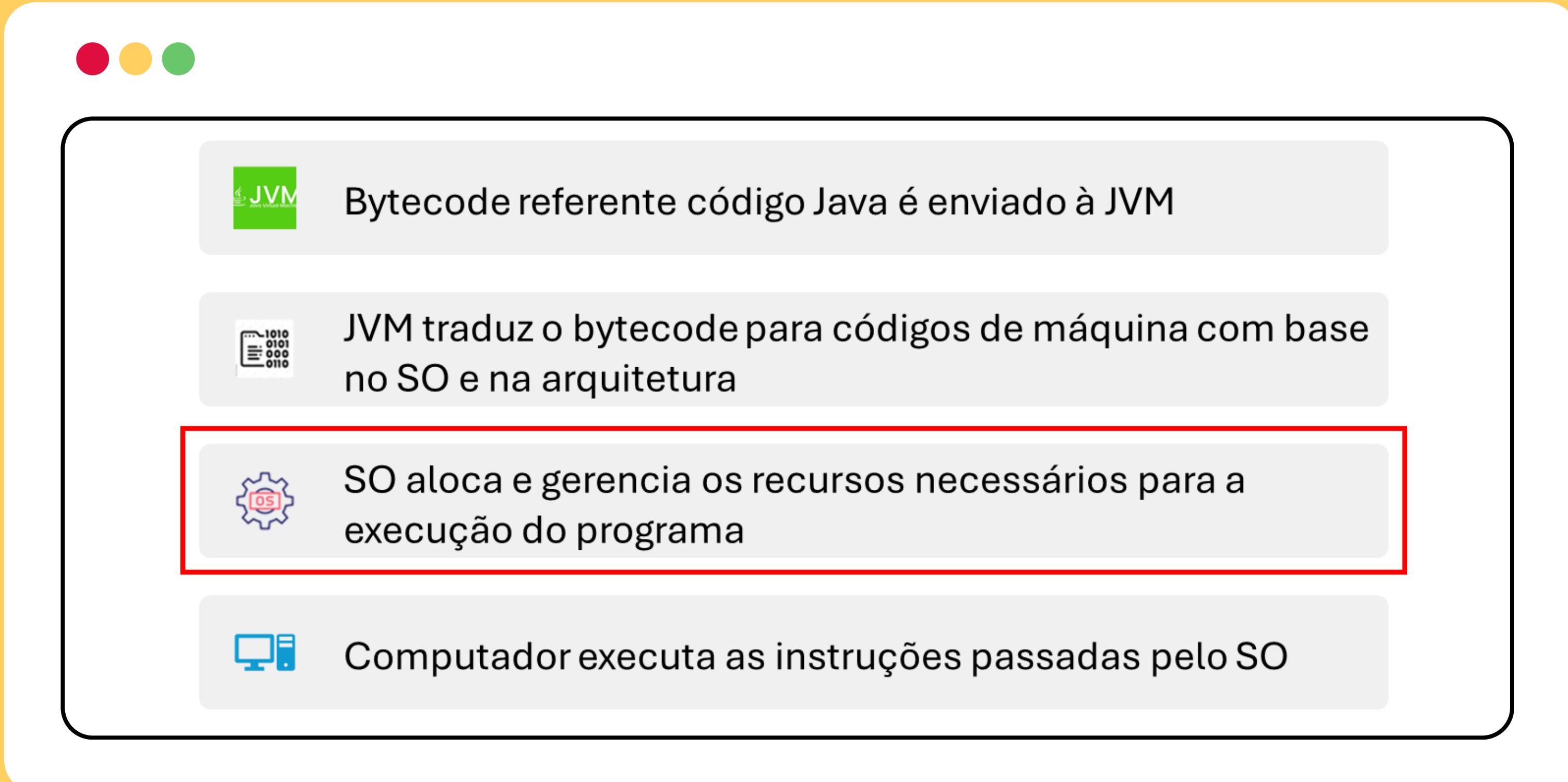


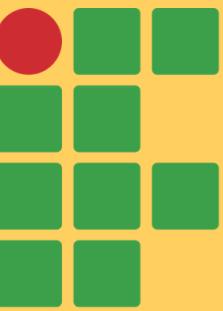
Compilação e execução



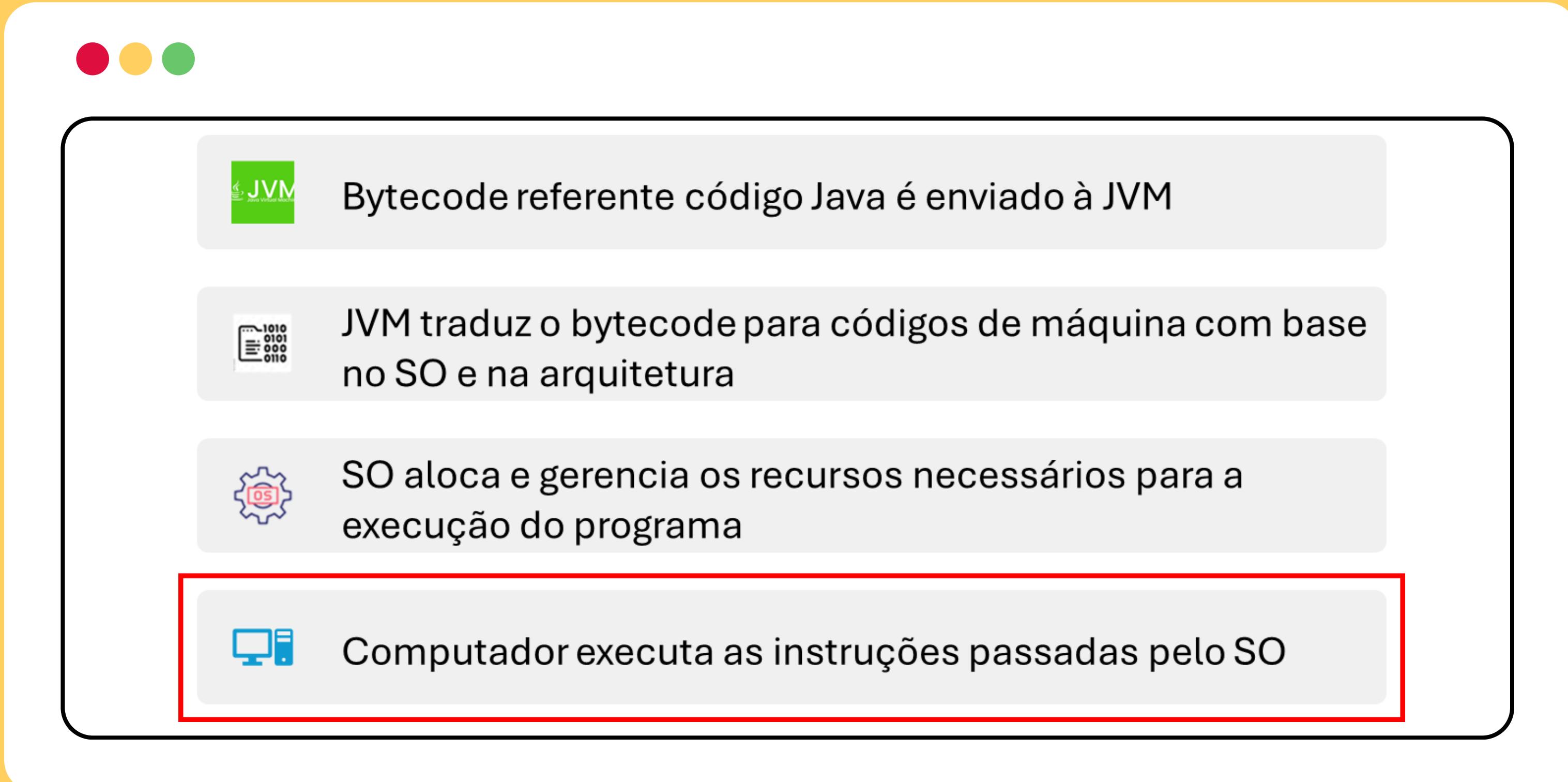


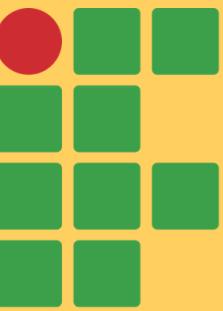
Compilação e execução





Compilação e execução

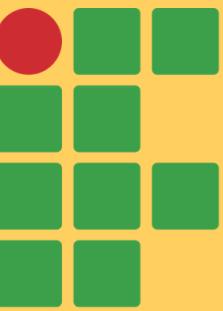




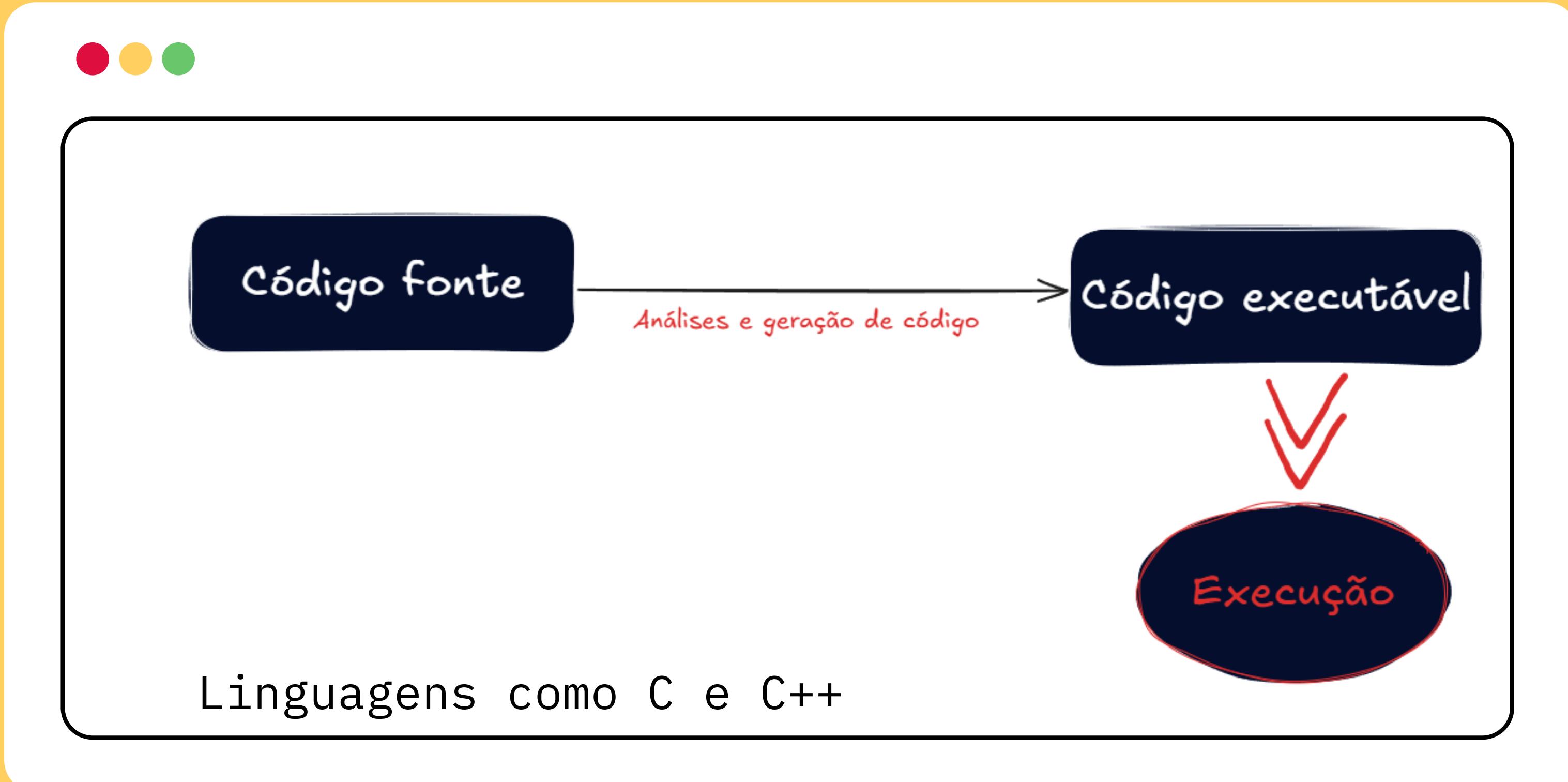
Em resumo

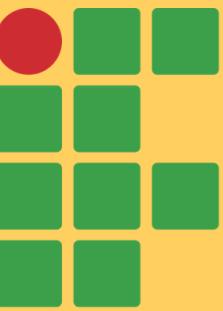


- Compilação;
- Interpretação;
- O processo híbrido do Java

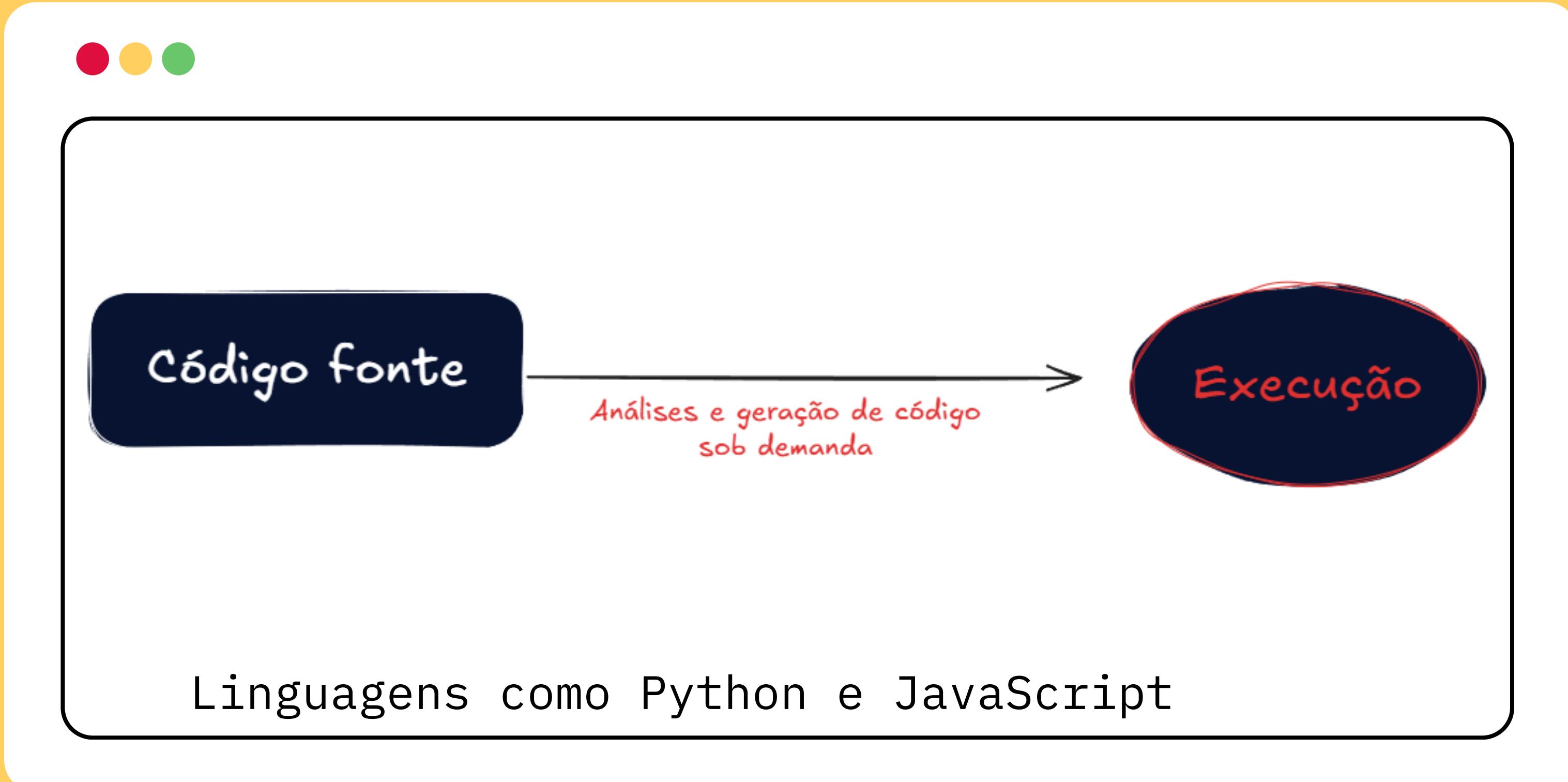


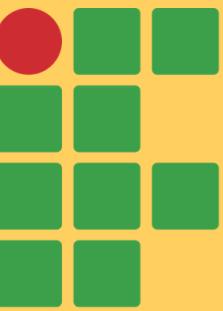
Compilação e execução



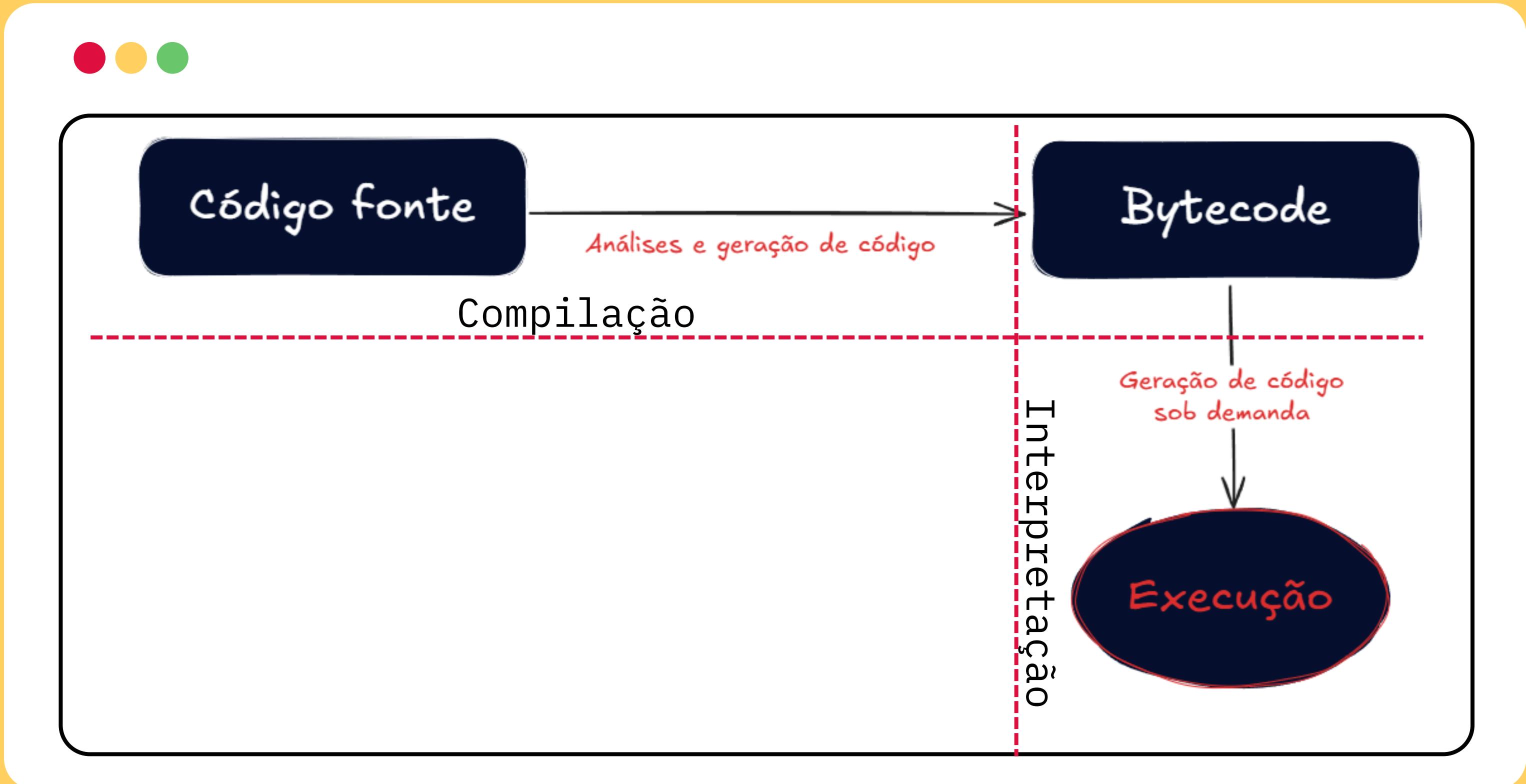


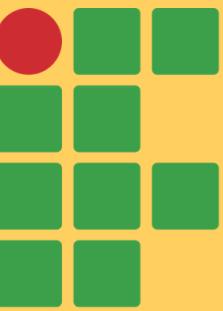
Interpretação





Abordagem híbrida do Java





Ganho de portabilidade



Código binário
para Windows



Windows

Código binário
para Linux



Linux

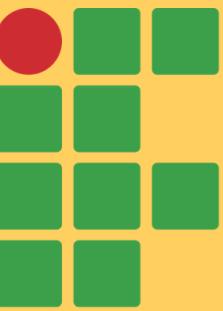
Código binário
para macOS



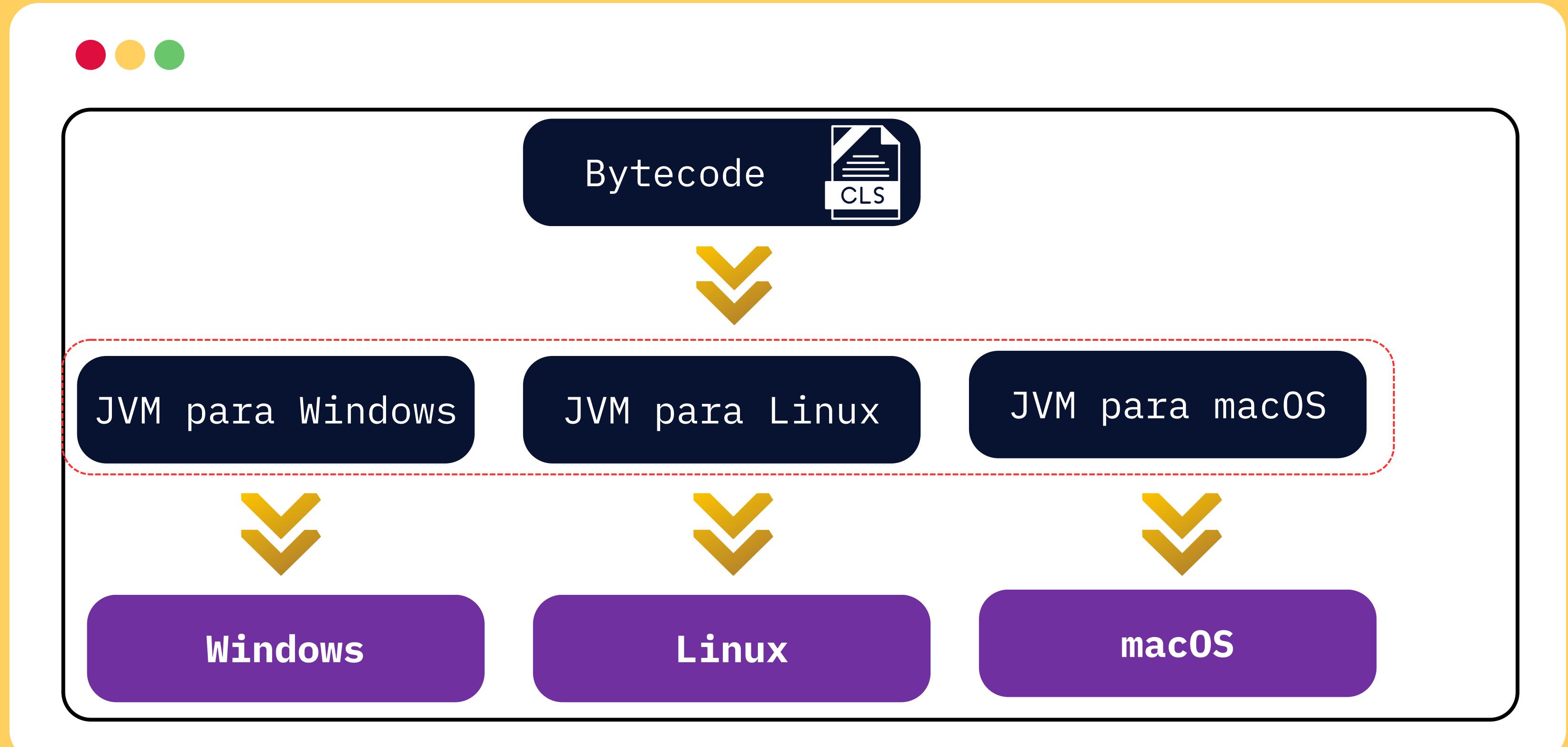
macOS

Linguagens tipicamente compiladas

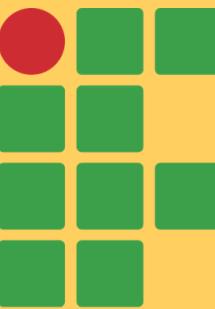




Ganho de portabilidade



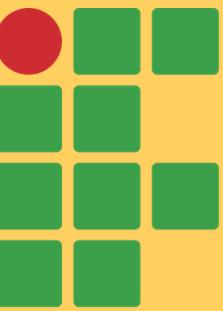
JVM: Java Virtual Machine



JVM: a JVM não entende código Java, mas um código intermediário específico – o bytecode;

Bytecode: gerado pelo compilador Java, o Javac;

Interpretação: o bytecode é interpretado pela JVM.



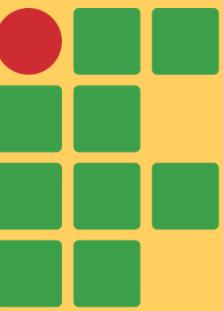
Downloads e instalações



JVM: apenas a máquina virtual, esse download não existe, ela sempre vem acompanhada;

JRE: Java Runtime Environment, ambiente de execução Java, formado pela JVM e bibliotecas, tudo que é necessário para executar uma aplicação Java;

JDK: Java Development Kit, é formado pela JRE somada a ferramentas, como o compilador.

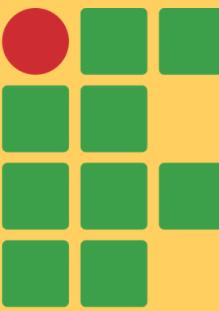


Vamos para a prática?



1. Baixe uma versão do SDK, como a disponível em <https://www.oracle.com/java/technologies/downloads/#jdk23-windows>;
2. Instale ou descompacte o arquivo em C:\Program Files\Java, a depender da escolha de download;
3. Crie a variável de ambiente JAVA_HOME com esse endereço acima;
4. Adicione à variável Path a referência %JAVA_HOME%\bin

Vamos para a prática?



THANK YOU!