

TI-2600 Laboratório de Desenvolvimento Multiplataforma

Antonio Carvalho - Treinamentos

Introdução ao Desenvolvimento Multiplataforma

Tipos de arquiteturas de desenvolvimento

- ▶ Nativo
- ▶ Web Progressivo
- ▶ Cross plataforma
- ▶ Multi plataforma

Tipos de arquiteturas de desenvolvimento

Nativo

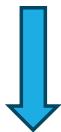
- ▶ **Definição:** Criação de aplicativos específicos para cada plataforma, utilizando as ferramentas e linguagens de programação nativas de cada uma.
- ▶ **Exemplo:** Desenvolvimento de aplicativos Android usando Kotlin ou Java.
- ▶ **Vantagens:**
 - ▶ Desempenho otimizado.
 - ▶ Acesso total a todas as APIs e funcionalidades da plataforma.
- ▶ **Desvantagens:**
 - ▶ Maior custo e tempo de desenvolvimento.
 - ▶ Necessidade de manter bases de código separadas para cada plataforma.

Nativo - Programação em linguagem clássica

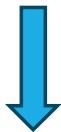
#include <stdio.h>

```
int main() {
    printf("Hello World");
    return 0;
}
```

Código
Fonte



Compilação



Código de
Máquina

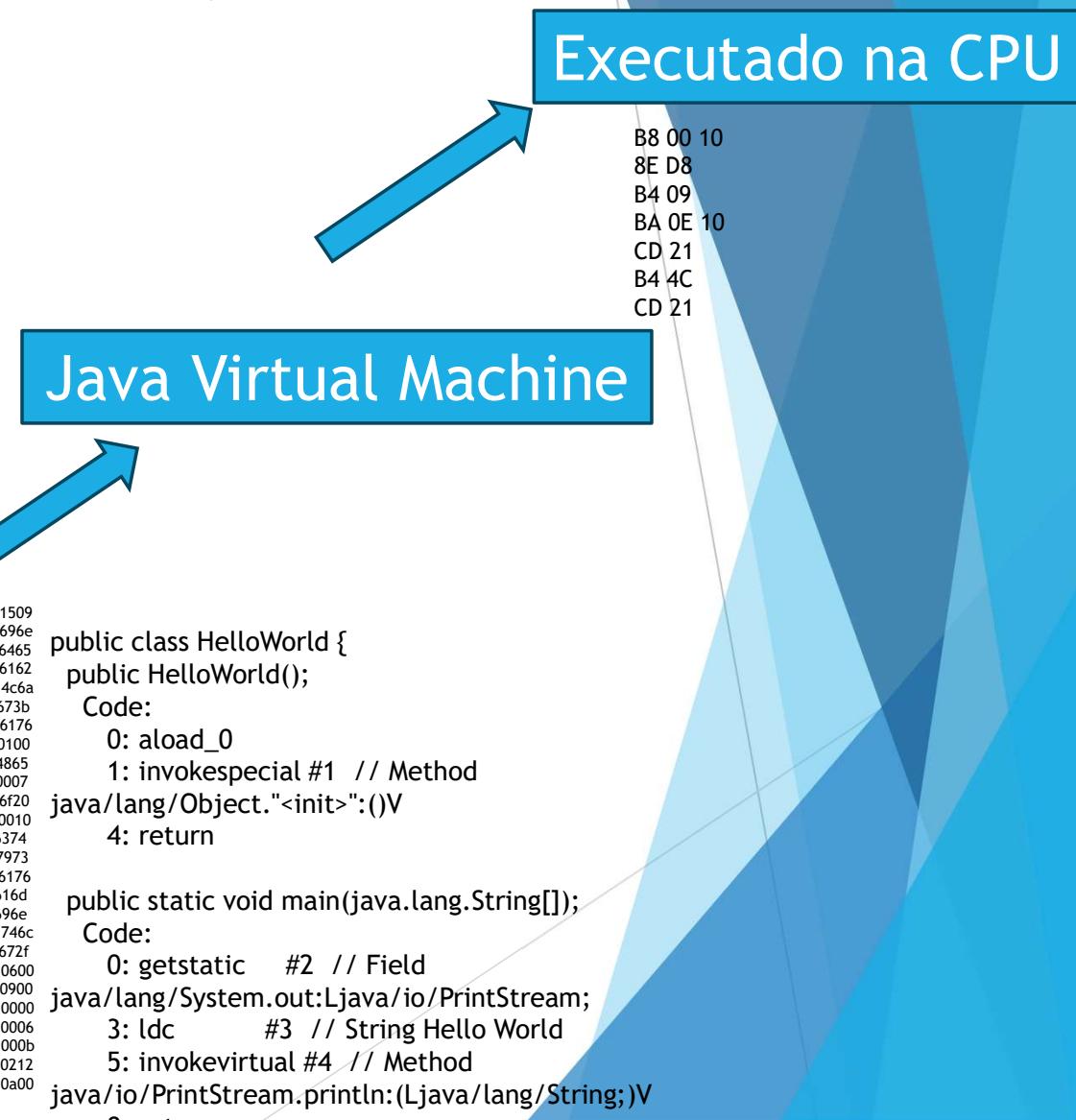
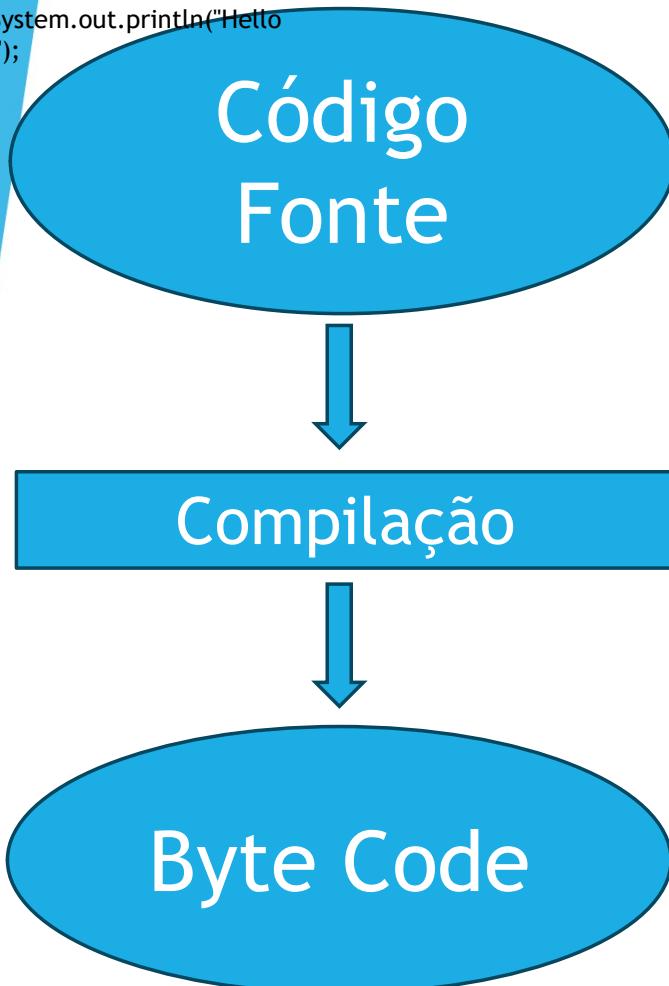
B8 00 10
8E D8
B4 09
BA 0E 10
CD 21
B4 4C
CD 21

Executado na CPU

```
.model small
.stack 100h
.data
msg db "Hello World$"
.code
main:
    mov ax, @data
    mov ds, ax
    mov ah, 9
    mov dx, offset msg
    int 21h
    mov ah, 4Ch
    int 21h
end main
```

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello World");  
    }  
}
```

Nativo - Programação em linguagem Moderna



Tipos de arquiteturas de desenvolvimento

Web Progressivo

- ▶ **Definição:** Criação de aplicativos web que podem ser instalados no dispositivo do usuário como se fossem aplicativos nativos.
- ▶ **Exemplo:** Aplicativos construídos com frameworks como Angular, React ou Vue.js.
- ▶ **Vantagens:**
 - ▶ Compatibilidade com qualquer dispositivo com navegador web.
 - ▶ Atualizações automáticas e sem necessidade de aprovação em lojas de aplicativos.
- ▶ **Desvantagens:**
 - ▶ Limitação no acesso a funcionalidades específicas do dispositivo.
 - ▶ Desempenho inferior comparado aos aplicativos nativos.

Web Progressivo

```
<html>  
  <h1>Hello  
  World</h1>  
</html>
```

Código Fonte em
HTML, CSS, JS

Compactado

Bundle

Executado no Navegador

```
<html> <h1>Hello World</h1></html>
```

Tipos de arquiteturas de desenvolvimento

Cross Plataforma

- ▶ **Definição:** Utilização de ferramentas e frameworks que permitem criar aplicativos que funcionam em várias plataformas, mas que não necessariamente compartilham uma base de código única.
- ▶ **Exemplo:** React Native
- ▶ **Vantagens:**
 - ▶ Redução de tempo de desenvolvimento.
 - ▶ Acesso a uma ampla variedade de bibliotecas e plugins.
- ▶ **Desvantagens:**
 - ▶ Complexidade na integração com componentes nativos.
 - ▶ Possíveis limitações em termos de desempenho e experiência do usuário.

Cross Plataforma

```
<View>  
  <Text> Ola Mundo </Text>  
</View>
```

Código Fonte em
Conjunto de Tags
+ JS + CSS

Compactado

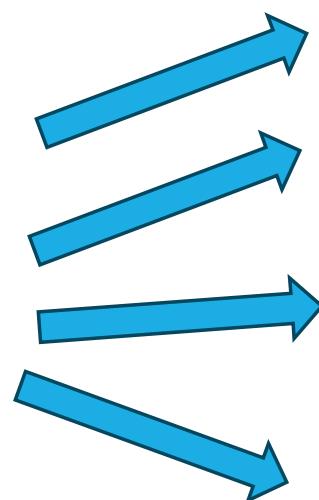
Bundle

App para a plataforma Web

App para a plataforma IoS

App para a plataforma Windows
Desktop

App para a plataforma Android



Tipos de arquiteturas de desenvolvimento

Multi Plataforma

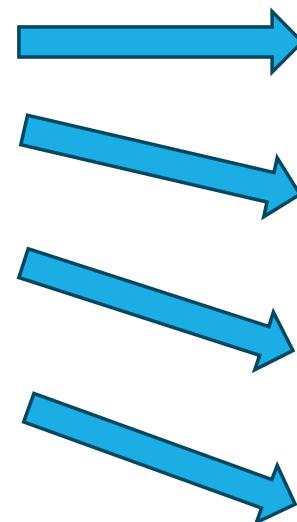
- ▶ **Definição:** Criação de aplicativos que podem ser executados em diferentes plataformas a partir de uma única base de código.
- ▶ **Exemplo:** Kotlin Multiplataforma
- ▶ **Vantagens:**
 - ▶ Redução de custos de desenvolvimento.
 - ▶ Reutilização de código.
 - ▶ Manutenção simplificada.
- ▶ **Desvantagens:**
 - ▶ Desempenho ligeiramente inferior em comparação ao desenvolvimento nativo.
 - ▶ Potencial limitação de acesso a APIs específicas de cada plataforma.

Multi Plataforma

```
Column() {  
    Text("Hello World")  
}
```

Código Fonte em
Linguagem
usando um
Framework

Compilado



Plataforma Web

Plataforma IoS

Plataforma Desktop

Plataforma Android