



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

Código
Fonte

```
#include <stdio.h>
```

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

Compilação

Código de
Máquina

Executado na CPU

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

```
.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
```

Nativo - Programação em linguagem Moderna

Executado na CPU

Java Virtual Machine

Código Fonte

Compilação

Byte Code

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

cafe babe 0000 0034 0021 0a00 0600 1509
0016 0017 0700 1807 0019 0100 063c 696e
6974 3e01 0003 2829 5601 0004 436f 6465
0100 0f4c 696e 654e 756d 6265 7254 6162
6c65 0100 046d 6169 6e01 0016 285b 4c6a
6176 612f 6c61 6e67 2f53 7472 696e 673b
2956 0100 0461 7267 7301 0014 4c6a 6176
612f 6c61 6e67 2f53 7472 696e 673b 0100
0a53 6f75 7263 6546 696c 6501 000f 4865
6c6c 6f57 6f72 6c64 2e6a 6176 610c 0007
0008 0c00 0900 0a01 0012 4865 6c6c 6f20
576f 726c 6407 001b 0c00 1c00 1d01 0010
6a61 7661 2f6c 616e 672f 4f62 6a65 6374
0100 106a 6176 612f 6c61 6e67 2f53 7973
7465 6d01 0003 6f75 7401 0015 4c6a 6176
612f 696f 2f50 7269 6e74 5374 7265 616d
3b01 0015 6a61 7661 2f69 6f2f 5072 696e
7453 7472 6561 6d01 0007 7072 696e 746c
6e01 0015 284c 6a61 7661 2f6c 616e 672f
5374 7269 6e67 3b29 5600 2100 0500 0600
0000 0200 0100 0700 0800 0000 0200 0900
0a00 0100 0b00 0000 1d00 0100 0100 0000
052a b700 01b1 0000 0001 000c 0000 0006
0001 0000 0001 0009 000d 000e 0001 000b
0000 0025 0002 0001 0000 0009 b200 0212
03b6 0004 b100 0000 0100 0c00 0000 0a00
0200 0000 0300 0700 0400 0800 01

```
public class HelloWorld {  
    public HelloWorld();  
    Code:  
        0: aload_0  
        1: invokespecial #1 // Method  
            java/lang/Object."<init>":()V  
        4: return  
  
    public static void main(java.lang.String[]);  
    Code:  
        0: getstatic   #2 // Field  
            java/lang/System.out:Ljava/io/PrintStream;  
        3: ldc         #3 // String Hello World  
        5: invokevirtual #4 // Method  
            java/io/PrintStream.println:(Ljava/lang/String;)V  
        8: return  
}
```

Tipos de arquiteturas de desenvolvimento

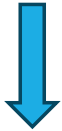
Web Progressivo

- ▶ **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.

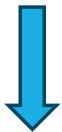
Web Progressivo

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

Código Fonte em
HTML, CSS, JS

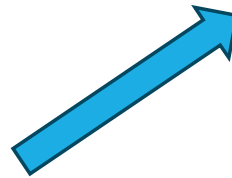


Compactado



Bundle

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



Executado no Navegador

Tipos de arquiteturas de desenvolvimento

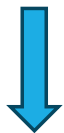
Cross Plataforma

- ▶ **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.

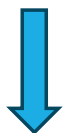
```
<View>  
<Text> Ola Mundo </Text>  
</View>
```

Cross Plataforma

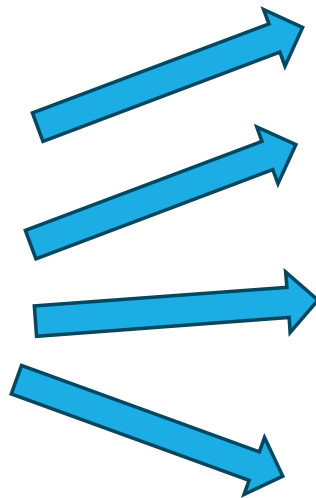
Código Fonte em
Conjunto de Tags
+ JS + CSS



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.

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

Multi Plataforma

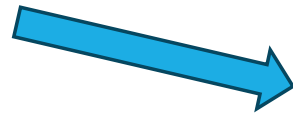
Código Fonte em
Linguagem
usando um
Framework



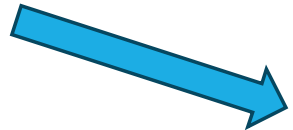
Compilado



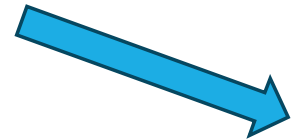
Plataforma Web



Plataforma iOS



Plataforma Desktop



Plataforma Android