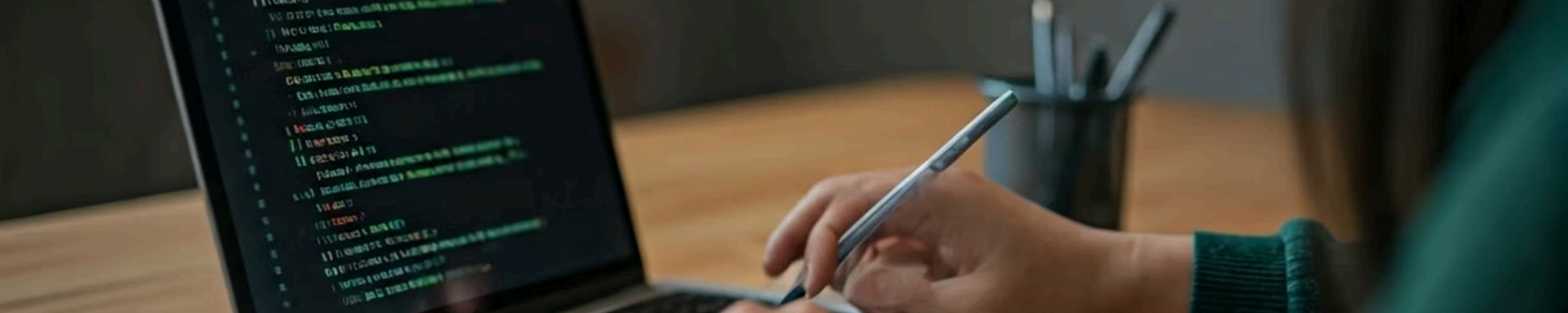


# Entendendo Estruturas de Controle em Swift

Condicionais e Laços de Repetição na prática

I por Isaac Mesquita



# Introdução



## Condicionais (if, else, else if)

Conceitos fundamentais de estruturas de controle de fluxo na programação Swift



## Laços de repetição (for, while, repeat while)

Esses conceitos são essenciais para criar programas que tomam decisões e executam tarefas repetidas.

# Condicionais: if / else

## Conceito

Condicionais permitem que um programa tome decisões com base em uma condição.

## Exemplo

```
let idade = 17

if idade >= 18 {
  print("Pode dirigir.")
} else {
  print("Não pode dirigir.")
}
```

# Condicional com else if

1

## Conceito

Permite testar várias condições diferentes.

2

## Exemplo

Verificação de notas em diferentes faixas

3

## Código

Estrutura com if, else if e else

```
let nota = 75

if nota >= 90 {
  print("Aprovado com mérito")
} else if nota >= 60 {
  print("Aprovado")
} else {
  print("Reprovado")
}
```

# Operadores em Condicionais

## Operadores Relacionais

- == (igual)
- != (diferente)
- <, >, <=, >=

## Operadores Lógicos

- && (e)
- || (ou)
- ! (negação)

## Exemplo

```
let idade = 20
```

```
let temCarteira = true
```

```
if idade >= 18 && temCarteira {
```

```
  print("Pode dirigir")
```

```
}
```

# Repetição com for



## Inicialização

for é usado quando sabemos quantas vezes queremos repetir.



## Exemplo

```
for i in 1...5 {
```



## Execução

```
print("Rodada (i)")
```



## Finalização

```
}
```

# for-in com arrays

## Definição do Array

```
let frutas = ["maçã", "banana", "laranja"]
```

## Finalização

```
}
```



## Início do Loop

```
for fruta in frutas {
```

## Processamento

```
print("Fruta: \ (fruta)")
```



# Repetição com while

1

## Conceito

while executa o bloco enquanto a condição for verdadeira.

2

## Exemplo

```
var contador = 3
```

```
while contador > 0 {
```

```
  print("Contando: \({contador}")
```

```
  contador -= 1
```

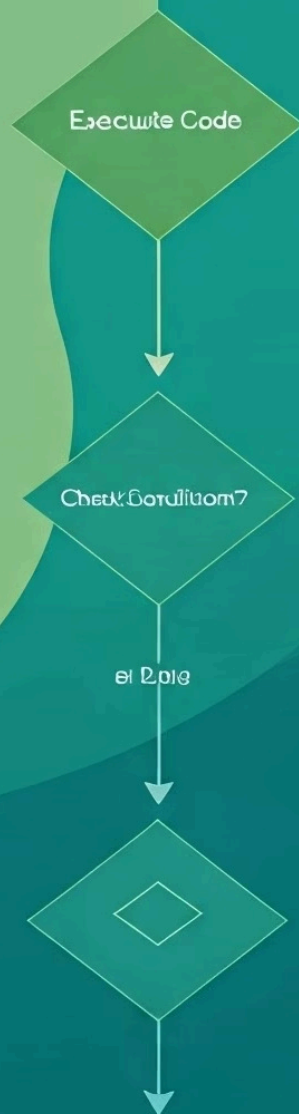
```
}
```

3

## Resultado

Contagem regressiva de 3 até 1





Udacity

# Repetição com repeat while

## Conceito

Semelhante ao while, mas garante ao menos uma execução.

## Exemplo

```
var tentativas = 0  
  
repeat {  
  print("Tentativa (tentativas)")  
  tentativas += 1  
} while tentativas < 3
```

## Diferença do while

A condição é verificada após a execução do bloco, garantindo pelo menos uma execução.

# Conclusão



Aprender a usar essas estruturas é o primeiro passo para desenvolver programas interativos e funcionais.