

# GOLANG

# Introdução

## PARTICIPANTES

.Cauã  
.Diego  
.Israel  
.Guilherme  
.Luiz Gustavo





# Características e Paradigmas de programação



## LINGUAGEM VERSÁTIL

Predominantemente imperativa, mas incorpora elementos de programação concorrente e funcional.

## SIMPLICIDADE DE SINTAXE

Sintaxe concisa, mecanismos mais básicos.

## EFICIÊNCIA DE DESEMPENHO

## SUORTE À CONCORRÊNCIA NATIVA

Bibliotecas.



# Sintaxe e Estrutura de Controles Básicas



TIPAGEM ESTÁTICA E INFERÊNCIA DE TIPOS

TIPAGEM FORTE

TIPOS DE DADOS VERSÁTEIS

VARIÁVEIS E CONSTANTES

CONVENÇÕES DE NOMECLATURA

ESTRUTURA DE CONTROLES BÁSICAS



# Recursos avançados e bibliotecas



CONCORRÊNCIA EM GO E GOROUTINES

GERENCIAMENTO DE MEMÓRIA EFICIENTE

TRATAMENTO DE ERROS E EXCEÇÕES

BIBLIOTECAS PADRÃO

MÓDULOS E PACOTES EXTERNOS

# Exemplos Práticos e Casos de Uso Relevantes

## DEMONSTRAÇÃO DE PROGRAMAS SIMPLES EM GO:

```
package main

import "fmt"

func main() {
    fmt.Println("Hello World!")
}
```

## EXEMPLO DE USO DE GOROUTINES PARA CONCORRÊNCIA:

```
package main

import (
    "fmt"
    "sync"
)

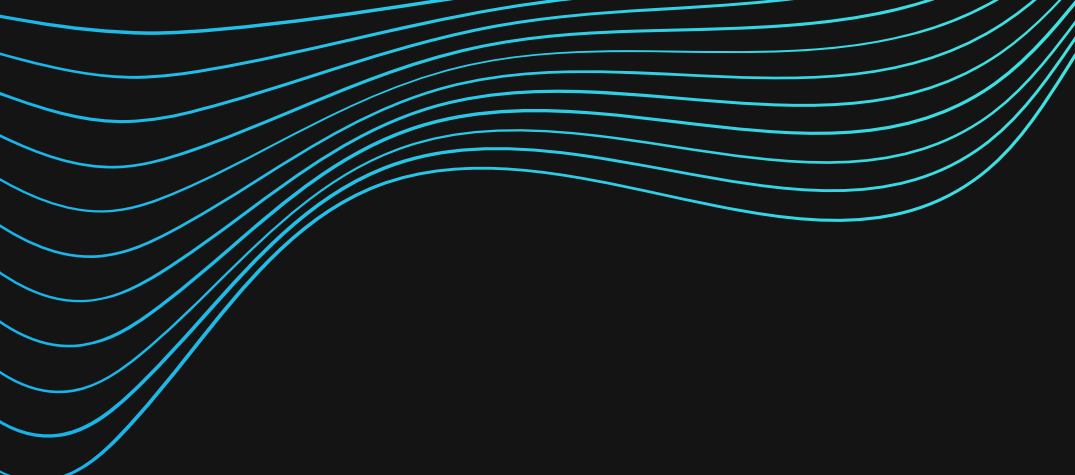
func printNumbers(wg *sync.WaitGroup) {
    defer wg.Done()
    for i := 0; i < 10; i++ {
        fmt.Printf("%d ", i)
    }
}

func printLetters(wg *sync.WaitGroup) {
    defer wg.Done()
    for i := 'A'; i < 'A'+10; i++ {
        fmt.Printf("%c ", i)
    }
}

func main() {
    var wg sync.WaitGroup
    wg.Add(2)

    go printNumbers(&wg)
    go printLetters(&wg)

    wg.Wait()
}
```



# Comparação com outras linguagens populares



## LINGUAGENS POPULARES

Python, Java, C++.

## VANTAGENS E DESVANTAGENS EM DIFERENTES CENÁRIOS

## EMPRESAS QUE UTILIZAM O GOLANG

Google, Uber, Dropbox.

# Conclusão

PERGUNTAS



# Referências