

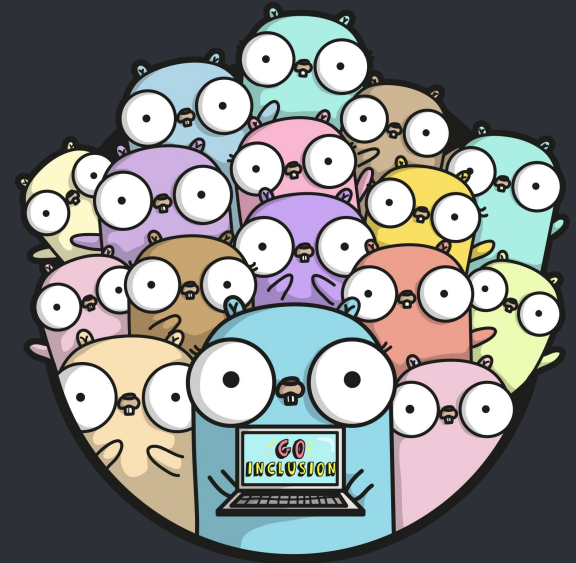
- Go Lang

BRUNO FREIRE
VINÍCIUS TEIXEIRA



1

ORIGENS



“

Com problemas para desenvolver sistemas para trabalhar com sua infraestrutura, o Google criou em 2007 a linguagem Go!





PROBLEMAS

- Tecnologia avançada porém complexa
- Programas com milhares de linhas
- Milhares de engenheiros trabalhando simultaneamente
- Processos de Merge e Compilação tornavam-se muito demorados

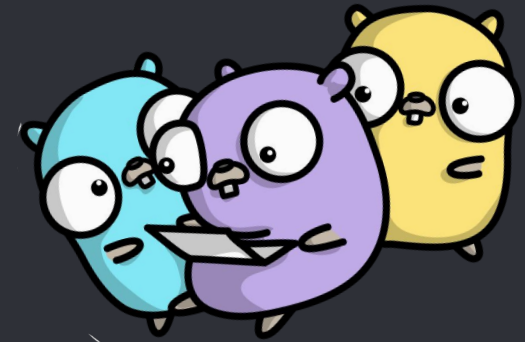
Precisavam de uma linguagem que atendesse todas as necessidades



CRIADORES

Assim nasceu a Go! em 2007 pelos engenheiros:

- Rob Pike
- Ken Thompson
- Robert Griessemer



Em 2009, Go se tornou Open Source e em 2010, começou a ser adotada por desenvolvedores de fora do Google.

INFLUÊNCIAS E CARACTERÍSTICAS

- Pascal
- C
- Oberon
- Newsqueak
- Limbo



- Imperativa
- Orientada a Objetos (Sem herança)
- Compilada (Goroutines)
- Estaticamente tipada
- Fortemente tipada
- Garbage-Collected
- Sintaxe simples
- Excelente pacote padrão
- Multiplataforma
- Concorrente
- Suporte a Closures
- Dependências explícitas
- Funções com múltiplos retornos
- Ponteiros

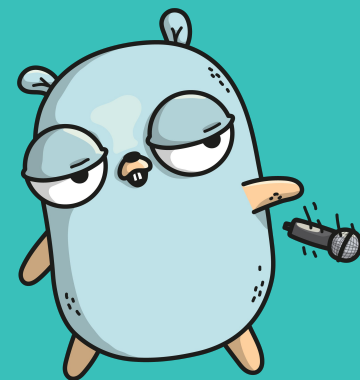
C#

```
1  using System;
2  using System.Linq;
3  using System.Threading;
4
5  namespace ConsoleApplication1
6  {
7      class Program
8      {
9          static void Main(string[] args)
10         {
11             Thread t1 = new Thread(contThread1);
12             Thread t2 = new Thread(contThread2);
13
14             t1.Start();
15             t2.Start();
16
17         }
18     }
```

Go!

```
1  package main
2
3  import "fmt"
4
5  func main() {
6
7      go f()
8      go g()
9
10 }
```

- Maior simplicidade na implementação
- Economia de memória com o uso de Goroutines
- Segurança ao compartilhar dados, com segurança, entre Goroutines durante a execução



Garbage-Collected

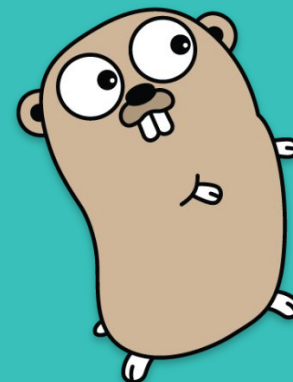
```
#include <stdio.h>
#include <stdlib.h>

void aloca()
{

    int *ptr;
    ptr = (int *) malloc(100);

    free(ptr);
}
```

Exemplo de alocação de memória em C



- Diferente de C, Go não necessita que o programador libere a memória
- Go tem um eficiente e performático garbage collector. Não se preocupe com gerenciamento de memória!



● ALGUMAS EMPRESAS QUE UTILIZAM GO!



- Amazon
- Google
- Globo.com
- Mozilla Foundation
- YouTube
- Vimeo
- Soundcloud
- Docker
- Nokia
- Heroku
- Bitly

