

# Linguagem GO



# Origem

- Robert Griesemer, Rob Pike e Ken Thompson
- Foco em programação concorrente

# Características

- Tipagem estática
- Compilada
- Orientação a objetos ?

# Comparativo

- Escrita e legibilidade semelhantes à Python
- Inferência de tipo semelhante à PHP
- Menos expressiva que linguagens como C#

```
package main

import "fmt"

func valores() (int, int) {
    return 3, 7
}

func main() {
    a, b := valores()
    fmt.Println(a)
    //Retorna 3
    fmt.Println(b)
    //Retorna 7
}
```

Código em C#:

```
using System;

namespace ConsoleApplication1 {
    public class Program {
        static void Main(string[] args) {

            Tuple<string, int> dados = Tuple.Create("Olá Mundo", 50);
            Console.WriteLine(dados);
            //retorna ("Olá mundo", 50)
            Console.ReadKey();

        }
    }
}
```

# Goroutines

```
package main
import "fmt"
func f(from string) {
    for i := 0; i < 3; i++ {
        fmt.Println(from, ":", i)
    }
}
func main() {

    f("direto")

    //Ao invocarmos uma goroutine a função não irá esperar a execução
    //da função anterior acabar.
    go f("goroutine")

    go func(msg string) {
        fmt.Println(msg)
    }("indo")

    var input string
    fmt.Scanln(&input)
    fmt.Println("pronto")
}
```

# Conclusão

- Indicado para sistemas concorrentes
- Alternativa como ferramenta com bom desempenho