

Introdução ao Go

Aula 18: Exercício - Construindo um Calculador de Lucros

Neste exercício, você criará uma ferramenta de **Calculadora de Lucros** em linha de comando utilizando Go. Essa aplicação pedirá ao usuário algumas informações e fará cálculos baseados nelas.

Objetivo

Desenvolver um programa que:

1. Solicite os seguintes valores ao usuário:
 - **Receita (Revenue);**
 - **Despesas (Expenses);**
 - **Taxa de Imposto (Tax Rate)** (em percentual).
2. Calcule e exiba:
 - **Lucro antes dos impostos** (Earnings Before Tax - EBT);
 - **Lucro após os impostos** (Earnings After Tax - EAT);
 - **A razão entre EBT e EAT.**

Lógica do Programa

1. **Entradas do Usuário:**
 - Obter os valores de receita, despesas e taxa de imposto.
2. **Cálculos:**
 - **Lucro antes dos impostos (EBT)** = Receita - Despesas;
 - **Lucro após os impostos (EAT)** = $EBT - (EBT * Taxa\ de\ Imposto / 100)$;

- Razão entre EBT e EAT = EBT / EAT .

3. Saída:

- Mostrar os valores calculados para EBT, EAT, e a razão.

Solução Base

A seguir está um exemplo de como implementar o programa em Go:

```
go
```

```
package main

import (
    "fmt"
)

func main() {
    var revenue, expenses, taxRate float64

    // Solicitar entrada do usuário
    fmt.Print("Enter the revenue: ")
    fmt.Scan(&revenue)

    fmt.Print("Enter the expenses: ")
    fmt.Scan(&expenses)

    fmt.Print("Enter the tax rate (in percentage): ")
    fmt.Scan(&taxRate)

    // Cálculos
    ebt := revenue - expenses
    eat := ebt - (ebt * taxRate / 100)
    ratio := ebt / eat

    // Exibir os resultados
    fmt.Printf("\nEarnings Before Tax (EBT): %.2f\n", ebt)
    fmt.Printf("Earnings After Tax (EAT): %.2f\n", eat)
```

```
fmt.Printf("Ratio (EBT / EAT): %.2f\n", ratio)
}
```

Exemplo de Uso

Entrada do Usuário:

mathematica

```
Enter the revenue: 10000
Enter the expenses: 7000
Enter the tax rate (in percentage): 10
```

Saída no Terminal:

java

```
Earnings Before Tax (EBT): 3000.00
Earnings After Tax (EAT): 2700.00
Ratio (EBT / EAT): 1.11
```

Tópicos Reforçados

1. Uso de variáveis para armazenar valores;
2. Entrada do usuário com `fmt.Scan()`;
3. Operações matemáticas básicas em Go;
4. Exibição formatada com `fmt.Printf()`.

Próximos Passos

Tente implementar o exercício por conta própria! Se tiver dúvidas ou quiser aprimorar o código com validações ou melhorias, posso ajudar!