**Exercício: MD01\_CP03\_AgenteSecreto**

Você é um agente secreto altamente treinado da organização MI7, e recebeu uma missão crítica: proteger a base secreta contra invasões inimigas. o local contém informações vitais para a segurança mundial e só pode ser acessado por um único sistema de segurança confirável. a sua tarefa é implementar um sistema de segurança para a base secreta usando o padrão singleton em TypeScript com NodeJS. a ideia é que apenas uma instância do sistema de segurança possa ser criada, garantindo que o local esteja bem protegido. siga as seguintes instruções:

1) crie uma classe chamada SistemaSeguranca. ela deve implementar o padrão singleton. certifique-se de que apenas uma instância da classe possa ser criada;

2) a classe SistemaSeguranca deve ter um método chamado acessarBaseSecreta. ele verifica se a senha inserida pelo agente coincide com a da Base Secreta. se estiver correta, o acesso é concedido. caso contrário, é negado;

3) crie um programa principal que demonstre o uso do singleton SistemaSeguranca. no programa um agente secreto tentará acessar a Base Secreta inserindo uma senha. se a senha estiver correta, o acesso será concedido. caso contrário será negado.

**Model/ SistemaSegurança.ts**

import { EventEmitter } from 'events';

export class SistemaSeguranca extends EventEmitter {

  private static instancia: SistemaSeguranca;

  private senhaBaseSecreta: *string* = 'senhaUltraSecreta123';

  private constructor() {

*super*();

  }

  public static obterInstancia(): SistemaSeguranca {

    if (!SistemaSeguranca.instancia) {

      SistemaSeguranca.instancia = **new** SistemaSeguranca();

    }

    return SistemaSeguranca.instancia;

  }

  public acessarBaseSecreta(*senhaInserida*: *string*): *void* {

    if (*senhaInserida* === *this*.senhaBaseSecreta) {

      console.log('Acesso concedido à Base Secreta!');

*this*.emit('acessoConcedido');

    } else {

      console.log('Acesso negado à Base Secreta. Senha incorreta.');

*this*.emit('acessoNegado');

    }

  }

}

**View/ ProgramaPrincipal.ts**

import { AgenteSecreto } from '../Controller/AgenteSecreto';

export const programaPrincipal = () => {

  const agenteController = **new** AgenteSecreto();

  // Agente tenta acessar a Base Secreta com senha correta

  agenteController.tentarAcessarBaseSecreta('senhaUltraSecreta123');

  // Agente tenta acessar a Base Secreta com senha incorreta

  agenteController.tentarAcessarBaseSecreta('senhaIncorreta456');

};

**Controller/ AgenteSecreto.ts**

import { SistemaSeguranca } from '../Model/SistemaSeguranca';

export class AgenteSecreto {

  public tentarAcessarBaseSecreta(*senha*: *string*): *void* {

    const sistemaSeguranca = SistemaSeguranca.obterInstancia();

    sistemaSeguranca.on('acessoConcedido', () => {

      console.log('O Agente Secreto está dentro da Base Secreta!');

    });

    sistemaSeguranca.on('acessoNegado', () => {

      console.log('O Agente Secreto foi detectado! Acesso negado!');

    });

    sistemaSeguranca.acessarBaseSecreta(*senha*);

  }

}

**App/ AgenteSecreto.ts**

import { programaPrincipal } from './View/ProgramaPrincipal';

// Executando o programa principal

programaPrincipal();