

- 1) Verifique nas alternativas abaixo quais compilam ou não. Explique o motivo:
 - a) `let a = 10;`
`a = "2";`
 - b) `let b: any = 10;`
`b = 2;`
 - c) `let c: number = 10;`
`c = 2;`
- 2) Dada a função soma abaixo, tente executar os scripts das alternativas e exiba os eventuais resultados:

```
function soma(x: number, y?: any): number {  
    return x + y  
}
```

 - a) `console.log(soma(1, 2));`
 - b) `console.log(soma(1, "2"));`
 - c) `console.log(soma(1));`
- 3) Crie uma enum com as siglas dos estados "PI", "CE", "MA" e implemente as duas alternativas abaixo:
 - a) Crie um laço usando for para imprimir esses valores;
 - b) Crie um laço que imprima os índices dessa enum e diga o que aconteceu.
- 4) Sobre enums, implemente o seguinte:
 - a) Crie uma enum chamada DiasSemana com os valores representando os dias da semana segunda a domingo;
 - b) Crie um namespace com mesmo nome e dentro dele crie uma função chamada isDiaUtil recebe um parâmetro do tipo DiasSema e retorna false se for um sábado ou domingo e retorna true caso contrário;
 - c) Escreva também um script que declara uma variável do tipo da enum e que testa a função DiasSemana.isDiaUtil().
- 5) Crie uma função chamada exhibir receba como parâmetro um "rest parameter" representando strings. A função deve exhibir no log cada um dos elementos do "rest parameter". Chame a função usando diferentes quantidade de parâmetros conforme abaixo:

```
exibir("a", "b");  
exibir("a", "b", "c");  
exibir("a", "b", "c", "d");
```
- 6) Converta em arrow function a seguinte função:

```
function ola() {  
    console.log("Olá");  
}
```
- 7) Dado método filter dos arrays, crie uma implementação usando arrow function que filtre todos os elementos pares do array abaixo:

```
const array = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15];
```

8) Crie uma classe chamada MeuNumero tenha:

- a) Um atributo chamado numero que seja somente leitura e inicializado no construtor;
- b) Um método chamado getInteiro que retorna a parte inteira do atributo numero;
- c) Um método chamado getDecimal que retorne a parte decimal do atributo numero.

Dica: utilize a função Math.floor(n)

- d) Instancie uma classe MeuNumero e teste os métodos da classe.

9) Crie uma classe chamada Transacao que tenha:

- a) Um atributo chamado valor e um outro chamado desconto, ambos somente leitura;
- b) Um método que calcule e retorne o valor do desconto aplicado ao valor original: $\text{valor} * (1 - \text{desconto}/100)$.
- c) Crie métodos de acesso get para ambos os atributos.
- d) Instancie uma classe Transacao e teste seus métodos