

Instituto Federal de Educação, Ciência e Tecnologia do RN – Campus Parnamirim
Programação Orientado a Objetos

Exercício Typescript – variáveis básicas, arrays e funções

Prof. Bruno E. G. Gomes
27/08/2018

1. Dada a lista de nomes abaixo use as funções de *array* para realizar as operações pedidas.

Referência dos métodos de *array*: https://www.w3schools.com/jsref/jsref_obj_array.asp

```
let nomes: string[] = ['Bruno', 'Elaine', 'José', 'Matheus', 'Maria',  
'Alana', 'Erika', 'Alison', 'Lucas', 'Rickson', 'Alberto', 'Beatriz',  
'Renata', 'Sayonara'];
```

- a) Adicione mais dois nomes na lista;
- b) Obtenha a quantidade de nomes adicionados;
- c) Remova o terceiro nome da lista;
- d) Realize uma busca por um nome qualquer e diga qual a posição dele na lista;
- e) Conte quantos números começam com a letra 'A'.

Nas questões 2 e 3 a seguir é preciso usar entrada de dados. Nesse caso, vamos usar uma biblioteca externa, chamada *readline-sync*. Para inserí-la, siga os passos a seguir:

1. No diretório do seu projeto *Typescript* digite o comando **npm init**. Responda às questões de configuração que se seguem e aperte ENTER após cada resposta. Caso queira, pode dar ENTER em todas as perguntas aceitando a configuração padrão. Esse comando irá gerar um arquivo chamado *package.json* que permite configurar algumas informações do seu projeto, como nome, versão e as bibliotecas (pacotes) das quais ele depende.
2. Instale a biblioteca *readline-sync* com o comando **npm i readline-sync**. Essa biblioteca permite a leitura de dados no modo console do *Typescript*
3. Após isso, no arquivo em que você for usar a biblioteca faça a sua importação inserindo, no início do arquivo, a linha de código:

```
import * as readline from 'readline-sync';
```

4. Para realizar uma leitura do teclado de um valor texto (*string*) use o método *question* que recebe o texto que irá aparecer para o usuário e devolve o valor digitado pelo usuário no teclado. Para ler inteiros, use *questionInt* e reais, use *questionFloat*. Exemplos:

```
let nome: string = readline.question('Qual o seu nome? ');
```

```
let palpite: number = readline.questionInt('Digite um número (de 0 a 30): ');
```

2. Faça um programa que peça ao usuário que ele adivinhe um número (gerado aleatoriamente) no intervalo de 0 (zero) a 30 (trinta), incluindo 0 e 30. Oriente o usuário em cada rodada, dizendo se o seu palpite é maior ou menor que o número gerado. O usuário tem até 5 (cinco) chances de acertar o número. A seguir segue um esqueleto do programa que você pode usar, incluindo a função para gerar o número aleatório.

```
import * as readline from 'readline-sync';

/* Faça o restante do seu programa a partir daqui */

while (/*teste de parada*/) {
    let palpite = readline.questionInt('Digite um número (0 a 30): ');
}

/* obtém um inteiro no intervalo (min, max) incluindo min e max */
function obterInt(min: number, max: number) : number {
    min = Math.ceil(min); //arredonda pra cima
    max = Math.floor(max); //arredonda para baixo
    return Math.floor(Math.random() * (max - min + 1)) + min;
}
```

3. Um problema recorrente na sociedade atual é o número de felicitações de parabéns que temos que dar em programas de mensagens. Para ajudar a melhorar essa realidade, faça um programa que receba como entrada o nome de uma pessoa e gere uma frase de felicitação de parabéns de forma automática. Para dar mais variedade, faça uma função que contenha no mínimo 5 (cinco) frases em uma lista (*array*). A função deve receber o nome da pessoa que foi digitado como entrada, selecionar uma frase e retornar a frase completa com o nome da pessoa inserido.

Dicas: Use a função de gerar um número aleatório do exercício anterior para obter uma frase de forma aleatória. No caso, o número gerado corresponde à posição da frase na lista. Use a string de template (com ` `) para permitir a inserção do nome nas frases de forma mais fácil.