

Aula 05

*Banco do Brasil (Escriturário - Agente de
Tecnologia) Desenvolvimento de
Software - 2023 (Pós-Edital)*

Autor:
**Raphael Henrique Lacerda, Paolla
Ramos e Silva**

12 de Janeiro de 2023

Índice

1) TypeScript - Teoria	3
2) TypeScript - Questões Comentadas	6
3) TypeScript - Lista de Questões	10



TYPESCRIPT

Conceitos Básicos

INCIDÊNCIA EM PROVA: BAIXA

TYPESCRIPT é uma evolução do Javascript proposta pela Microsoft, que tem como principal característica a tipagem forte, sendo compilado para javascript, open-source e utilizado em vários frameworks web.

Diante das limitações que o Javascript possui, Anders Hejlsberg, que também participou da criação do C#, do Delphi, do Turbo Pascale da plataforma .NET, desenvolveu o Typescript. Que até então não suportava a programação OO. Importante destacar que no momento da compilação o Typescript se torna Javascript, a razão disso é que o Typescript foi construído em cima do Javascript. Isso facilita a vida dos navegadores que não precisam entender outra linguagem, somente o javascript. *Professor, e quais são as diferenças do JS para o TS?*



Anote aí! Ouviu TypeScript, pensou: **Estático, Tipagem forte, Orientado a objetos, Genérico, Namespaces e Decorators.** A característica mais marcante do TypeScript é sem dúvida é sua tipagem forte, motivo pelo qual leva no seu nome: type (tipagem). Temos o aspecto da Orientação a Objetos que é muito importante do ponto de vista estrutural da linguagem, sendo um paradigma que facilita a adoção da mesma.

O JavaScript, na maior parte do seu projeto de linguagem, não é tipado e a inferência de tipo só vai até certo ponto, o que abre caminho para utilização de TypeScript, a fim de suprir essa deficiência.

Vejamos um exemplo de código escrito em TypeScript e depois em Javascript:



CONDICÃO DE APROVAÇÃO EM TYPESCRIPT

```
type Result = "aprovado" | "reprovado"

function verify(result: Resultado) {
  if (result === "aprovado") {
    console.log("Aprovado")
  } else {
    console.log("Reprovado")
  }
}
```

CONDICÃO DE APROVAÇÃO EM JAVASCRIPT

```
function verify(result) {
  if (result === "aprovado") {
    console.log("Aprovado")
  } else {
    console.log("Reprovado")
  }
}
```

Pessoal, a diferença nesse exemplo é bem simples, com TypeScript temos adição de sintaxe natural para fornecimento de types (tipos). No typescript, uma string é uma sequencia de caracteres, considerado objeto. Um objeto em typescript é declarado com {}, enquanto que um array é []

TIPOS MAIS EMPREGADOS NO TYPESCRIPT

NUMBER: É para todo e qualquer tipo de número, seja ele ponto flutuante ou inteiro.

STRING: Representa uma string costumeiramente conhecida em outras linguagens de programação.

BOOLEAN: Representa um valor booleano: true ou false.

ANY: A tradução de Any é qualquer e, como sua tradução sugere, é um tipo que pode ser modificado para qualquer outro tipo presente na linguagem, seja string, number, boolean ou qualquer outra coisa.

ARRAY: Representa o tipo Array dentro da linguagem. É válido informar que podemos criar arrays de duas formas dentro do TypeScript.

Importante: Para indicar a raíz de arquivos Typescript basta encontrarmos arquivos do tipo tsconfig.json e a linguagem Typescript não aceita heranças múltiplas.

(CCV – 2019) Sobre TypeScript, assinale a alternativa correta.

- a) Toda função deve possuir um nome.
- b) TypeScript é um subconjunto de JavaScript.
- c) Em um enum não é possível misturar membros de tipos baseados em string e number.



- d) Em uma classe TypeScript é possível utilizar herança múltipla por meio da palavra chave `extends`.
- e) A presença de um arquivo `tsconfig.json` em um diretório indica que o diretório é a raiz de um projeto TypeScript.

Comentários: (a) Errado. Função não precisa necessariamente ter nome, é possível invocar uma função anônimas; (b) Errado. Como vimos, Typescript é uma linguagem e não um subconjunto de JS; (c) Errado. É possível sim misturar strings, number e boolean, que são tipos em typescript em uma enum; (d) Errado. Não se utiliza herança múltipla em Typescript pela adoção ao paradigma orientado a objeto, logo não cabe herança múltipla conforme vimos; (e) Correto. A presença de um arquivo `tsconfig.json` em um diretório indica que o diretório é a raiz de um projeto TypeScript (Letra E).



QUESTÕES COMENTADAS – TYPESCRIPT

1. (CESGRANRIO / BB - 2021) Considere o fragmento de código TypeScript a seguir.

```
const a=<T extends {b:string}> (obj:T)=>{<códigoremovido>};
```

Com relação ao código apresentado acima, a(o)

- a) função a() retorna um objeto do tipo string.
- b) variável a é uma lista de objetos do tipo string.
- c) variável a é um dicionário cujas chaves são objetos do tipo string.
- d) objeto que for passado para a função a() deve ter um campo b do tipo string.
- e) valor retornado pela função a() é um objeto que estende um objeto do tipo string.

Comentários:

No typescript, string é uma sequência de caracteres, considerado objeto. um objeto em typescript é declarado com {}, enquanto que um array é [].

Gabarito: Letra D

2. (CESGRANRIO / BB - 2021) Considere o código HTML a seguir.

```
<!doctype html>
<html lang="pt-br">
<head>
<script src="script.js"></script>
</head>
<body>
<form>
Texto: <input type='text' name='texto' id='idTexto' class='classe-input' value='Texto inicial'><br>
<input type='submit' value='Envia'>
</form>
</body>
</html>
```

Considere, também, o arquivo TypeScript script.ts, listado a seguir, que irá gerar o arquivo script.js no mesmo diretório do arquivo HTML, apresentado acima.

```
onload = (event) => {
  const texto = document.querySelector('??') as HTMLInputElement;
  console.log('Texto inicial: ', texto.value);
};
```



Que texto o programador deverá utilizar no lugar de ???, no código do arquivo TypeScript script.ts, para exibir o valor do campo HTML input na console?

- a) #classe-input
- b) #idTexto
- c) #texto
- d) .idTexto
- e) .texto

Comentários:

No lugar de ??? basta utilizar o #idTexto, a fim de buscar o atributo id no código do arquivo TypeScript script.ts, a partir do campo HTML input.

Gabarito: Letra B

3. (FCC / MPE-PB - 2018) Considere o fragmento de código TypeScript abaixo.

```
interface CriaArrayString {  
  [indice: number]: string;  
}  
var nomes: CriaArrayString;  
nomes = ["Ana", "Pedro", "Mariana"];  
document.body.innerHTML = nomes[1];
```

Ao executar esse código:

- a) ocorrerá um erro na linha que contém o comando [indice: number]: string;
- b) será exibido na tela o nome Pedro.
- c) ocorrerá um erro na linha que contém o comando var nomes: CriaArrayString;
- d) será exibido na tela o nome Ana.
- e) ocorrerá um erro na linha que contém o comando document.body.innerHTML = nomes[1];

Comentários:

Quando for executado o nome Pedro será exibido na tela, porque ao definir o array com as strings nome = ["Ana", "Pedro", "Mariana"] e invocar o nome [1], devemos lembrar que um array começa em 0, logo:

Nome [0] = Ana
Nome [1] = Pedro
Nome [2] = Mariana

Gabarito: Letra B



4. (UFC / CCV - 2019) Para o desenvolvimento de aplicações Web, qual item abaixo contém apenas frameworks/bibliotecas/plataformas que foram desenvolvidas ou que dependem de JavaScript ou TypeScript:

- a) Node.js, CSS, Java.
- b) React, Node.js, Scala.
- c) Angular, React, Vue.js.
- d) Angular, Node.js, Java.
- e) Java AWT, Angular, Scala.

Comentários:

Os frameworks que utilizam tanto JS como TS são Angular, React e Vue, respectivamente todos podemos inclusive ser escritos como AngularJS, ReactJS e VueJS.

Gabarito: Letra C

5. (CVV / UFC - 2019) Sobre TypeScript, assinale a alternativa correta.

- a) Toda função deve possuir um nome.
- b) TypeScript é um subconjunto de JavaScript.
- c) Em um enum não é possível misturar membros de tipos baseados em string e number.
- d) Em uma classe TypeScript é possível utilizar herança múltipla por meio da palavra chave extends.
- e) A presença de um arquivo tsconfig.json em um diretório indica que o diretório é a raiz de um projeto TypeScript.

Comentários:

(a) Errado. Não necessariamente toda função deve possuir nome, é possível aplicar funções anônimas por exemplo, com arrow functions, por exemplo; (b) Errado. Typescript não é subconjunto do Javascript; (c) Errado. É possível misturar tanto strings, numbers ou boolean, que são os únicos tipos em typescript, em uma "enum"; (d) Errado. Typescript não utiliza herança múltipla conforme estudamos; (e) Correto. A presença de um arquivo tsconfig.json em um diretório indica que o diretório é a raiz de um projeto TypeScript.

Gabarito: Letra E

6. (FCC / MPE-PE - 2018) Considere o fragmento de código TypeScript abaixo.



```
interface CriaArrayString {  
  [indice: number]: string;  
}  
var nomes: CriaArrayString;  
nomes = ["Ana", "Pedro", "Mariana"];  
document.body.innerHTML = nomes[1];
```

Ao executar esse código:

- a) ocorrerá um erro na linha que contém o comando [indice: number]: string;
- b) será exibido na tela o nome Pedro.
- c) Ocorrerá um erro na linha que contém o comando var nomes: CriaArrayString;
- d) Será exibido na tela o nome Ana.
- e) Ocorrerá um erro na linha que contém o comando document.body.innerHTML = nomes[1];

Comentários:

(a) Errado. Não existe motivo para ocorrer um erro no comando: [indice: number]: string; (b) Correto. Será exibido Pedro; (c) Errado. Não existe motivo para ocorrer um erro na linha var nomes: CriaArrayString; (d) Errado. Será exibido Pedro e não Ana; (e) Errado. Também não existe erro no comando document.body.innerHTML = nomes[1];

Gabarito: Letra B



LISTA DE QUESTÕES – TYPESCRIPT

1. (CESGRANRIO / BB - 2021) Considere o fragmento de código TypeScript a seguir.

```
const a=<T extends {b:string}> (obj:T)=>{<códigoremovido>};
```

Com relação ao código apresentado acima, a(o)

- a) função a() retorna um objeto do tipo string.
- b) variável a é uma lista de objetos do tipo string.
- c) variável a é um dicionário cujas chaves são objetos do tipo string.
- d) objeto que for passado para a função a() deve ter um campo b do tipo string.
- e) valor retornado pela função a() é um objeto que estende um objeto do tipo string.

2. (CESGRANRIO / BB - 2021) Considere o código HTML a seguir.

```
<!doctype html>
<html lang="pt-br">
<head>
<script src="script.js"></script>
</head>
<body>
<form>
Texto: <input type='text' name='texto' id='idTexto' class='classe-input' value='Texto inicial'><br>
<input type='submit' value='Envia'>
</form>
</body>
</html>
```

Considere, também, o arquivo TypeScript script.ts, listado a seguir, que irá gerar o arquivo script.js no mesmo diretório do arquivo HTML, apresentado acima.

```
onload = (event) => {
    const texto = document.querySelector('??') as HTMLInputElement;
    console.log('Texto inicial: ', texto.value);
};
```

Que texto o programador deverá utilizar no lugar de ???, no código do arquivo TypeScript script.ts, para exibir o valor do campo HTML input na console?

- a) #classe-input
- b) #idTexto
- c) #texto
- d) .idTexto



e) .texto

3. (FCC / MPE-PB - 2018) Considere o fragmento de código TypeScript abaixo.

```
interface CriaArrayString {  
  [indice: number]: string;  
}  
var nomes: CriaArrayString;  
nomes = ["Ana", "Pedro", "Mariana"];  
document.body.innerHTML = nomes[1];
```

Ao executar esse código:

- a) ocorrerá um erro na linha que contém o comando [indice: number]: string;
- b) será exibido na tela o nome Pedro.
- c) ocorrerá um erro na linha que contém o comando var nomes: CriaArrayString;
- d) será exibido na tela o nome Ana.
- e) ocorrerá um erro na linha que contém o comando document.body.innerHTML = nomes[1];

4. (UFC / CCV - 2019) Para o desenvolvimento de aplicações Web, qual item abaixo contém apenas frameworks/bibliotecas/plataformas que foram desenvolvidas ou que dependem de JavaScript ou TypeScript:

- a) Node.js, CSS, Java.
- b) React, Node.js, Scala.
- c) Angular, React, Vue.js.
- d) Angular, Node.js, Java.
- e) Java AWT, Angular, Scala.

5. (CVV / UFC - 2019) Sobre TypeScript, assinale a alternativa correta.

- a) Toda função deve possuir um nome.
- b) TypeScript é um subconjunto de JavaScript.
- c) Em um enum não é possível misturar membros de tipos baseados em string e number.
- d) Em uma classe TypeScript é possível utilizar herança múltipla por meio da palavra chave extends.
- e) A presença de um arquivo tsconfig.json em um diretório indica que o diretório é a raiz de um projeto TypeScript.

6. (FCC / MPE-PE - 2018) Considere o fragmento de código TypeScript abaixo.



```
interface CriaArrayString {
  [indice: number]: string;
}
var nomes: CriaArrayString;
nomes = ["Ana", "Pedro", "Mariana"];
document.body.innerHTML = nomes[1];
```

Ao executar esse código:

- a) ocorrerá um erro na linha que contém o comando [indice: number]: string;
- b) será exibido na tela o nome Pedro.
- c) Ocorrerá um erro na linha que contém o comando var nomes: CriaArrayString;
- d) Será exibido na tela o nome Ana.
- e) Ocorrerá um erro na linha que contém o comando document.body.innerHTML = nomes[1];



GABARITO – TYPESCRIPT

- | | |
|------------|------------|
| 1. LETRA D | 4. LETRA C |
| 2. LETRA B | 5. LETRA E |
| 3. LETRA B | 6. LETRA B |



ESSA LEI TODO MUNDO CONHECE: PIRATARIA É CRIME.

Mas é sempre bom revisar o porquê e como você pode ser prejudicado com essa prática.



1

Professor investe seu tempo para elaborar os cursos e o site os coloca à venda.



2

Pirata divulga ilicitamente (grupos de rateio), utilizando-se do anonimato, nomes falsos ou laranjas (geralmente o pirata se anuncia como formador de "grupos solidários" de rateio que não visam lucro).



3

Pirata cria alunos fake praticando falsidade ideológica, comprando cursos do site em nome de pessoas aleatórias (usando nome, CPF, endereço e telefone de terceiros sem autorização).



4

Pirata compra, muitas vezes, clonando cartões de crédito (por vezes o sistema anti-fraude não consegue identificar o golpe a tempo).



5

Pirata fere os Termos de Uso, adultera as aulas e retira a identificação dos arquivos PDF (justamente porque a atividade é ilegal e ele não quer que seus fakes sejam identificados).



6

Pirata revende as aulas protegidas por direitos autorais, praticando concorrência desleal e em flagrante desrespeito à Lei de Direitos Autorais (Lei 9.610/98).



7

Concursado(a) desinformado participa de rateio, achando que nada disso está acontecendo e esperando se tornar servidor público para exigir o cumprimento das leis.



8

O professor que elaborou o curso não ganha nada, o site não recebe nada, e a pessoa que praticou todos os ilícitos anteriores (pirata) fica com o lucro.



Deixando de lado esse mar de sujeira, aproveitamos para agradecer a todos que adquirem os cursos honestamente e permitem que o site continue existindo.