



3. Tipos de Dados e Variáveis

Neste capítulo você aprenderá a:

- Identificar os tipos de dados básicos utilizados na programação.
- Utilizar o Replit para praticar a programação em *javascript*.
- Criar variáveis e constantes em *javascript*.

3.1 Tipos de Dados

Parabéns por chegar até aqui, a partir de agora todos os conceitos da Parte I serão estruturados para uma linguagem de programação. Como linguagem optei por usar o *javascript*, por ser mais acessível visto que todo navegador de internet o possui nativamente. Nossa jornada a partir de agora será conhecer os tipos de dados e de variáveis. Vimos na Seção 2.2 que as variáveis são essenciais para o funcionamento de todos os algoritmos.

Podemos destacar dois tipos de informações que toda linguagem de programação possui: instruções (comandos) e dados.

- Instruções: determinam como os dados serão tratados, cada computador pode processar de uma forma diferente. Exemplos: comandos de entrada, comandos de saída, comandos condicionais (*se-senão*), entre outros.
- Dados: representam uma porção de informações que serão processadas, manipuladas, pelo computador. Exemplo: os valores de *a*, *b* e *x* do Exemplo 2.9.

Você deve ter percebido que todas as variáveis tratadas nos pseudocódigos (Seção 2.2.3) estavam em um estilo de texto diferente. Essas variáveis são representadas por um tipo de dado. Faço uma restrição neste material introdutório a três dados básicos: numérico, literal e lógico (Figura 3.1). Porém existem outros tipos de dados que são específicos de cada linguagem de programação; no caso do *javascript* existem os tipos: **function** (função), **object** (objeto), **undefined** (indefinido) e **null** (nulo).

Na Figura 3.1, temos três tipos de dados básicos:

1. Literal (**string**): representam uma sequência de caracteres que podem ser formados por letras, dígitos e símbolos especiais, delimitados por aspas duplas, aspas simples

ou crase (no *javascript*). Exemplos: "A","Hoje irei programar", 'Como está?', ` Bem!;)` . Observe que o texto inicia por um desses símbolos e finaliza com o mesmo que iniciou. Não se pode misturar a representação. Caso opte por usar aspas duplas inicie e finalize o literal com aspas duplas.

❗ A representação mais utilizada, na maioria das linguagens de programação, é a de aspas duplas.

2. Numérico (**number**): correspondem aos números, são divididos em reais e inteiros. Os números reais são números que possuem uma parte fracionada (indicada por uma vírgula), por exemplo, 3,54. Inteiro é um número que não possui vírgula, por exemplo, 10.

❗ A vírgula nos números reais serão substituídos em nossos programas por ponto. Logo, se você quiser utilizar o valor 3,54 então no seu programa você deverá escrever 3.54, ok!?

3. Lógico (**boolean**): São usados para representar dois únicos valores possíveis, verdadeiro (**true**) ou falso (**false**).

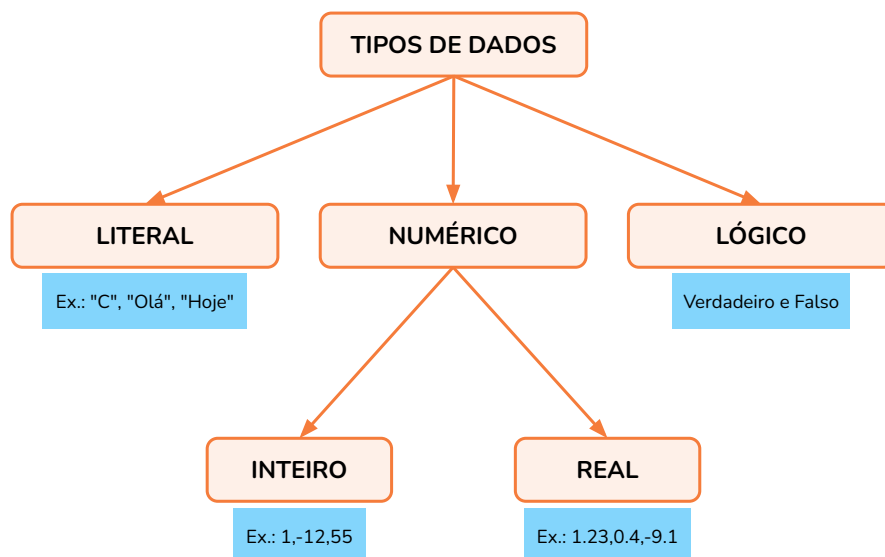


Figura 3.1: Tipos de dados básicos com exemplos.

3.2 Tipos de Dados no Javascript

Agora vamos ver esses três tipos de dados básicos no *javascript* e como identificá-los. Inicialmente irei mostrar como você pode reproduzir todos os exemplos no seu computador. Por praticidade utilizei o software Replit¹ (site) para hospedar todos os códigos. Criei também uma lista com tutoriais² em forma de vídeo para que consiga executar os primeiros passos neste software. A linguagem que deverá ser selecionada no Replit para realização das implementações será o **nodejs**, uma linguagem baseada em *javascript* (TILKOV; VINOSKI, 2010).

¹<<https://replit.com/>>

²<<https://bit.ly/tutoriais-replit>>

Na Figura 3.2, destaco as principais áreas que iremos trabalhar no Replit. Seguem algumas informações adicionais relacionadas a cada indicador da figura:

- A) É o nome do projeto que você define ao criar um replit. Pode ser alterado sempre que quiser.
- B) É o botão utilizado para executar o programa. Caso deseje que o programa em execução pare basta clicar nele novamente.
- C) Nesta área ficarão listados os arquivos do seu projeto. Em nossos exemplos e práticas usaremos somente um arquivo por projeto.
- D) O nome do arquivo principal do nosso projeto é o `index.js` (por padrão). O `.js` no final do arquivo determina o tipo de arquivo: *javascript*. Você não precisará alterar o nome desse arquivo em nossos exemplos e práticas.
- E) Nesta área colocaremos o código-fonte do programa que será executado ao apertar no botão verde (indicador B).
- F) Nesta área, denominada **console**, você verá a saída de dados do seu programa. E também é a área de entrada de dados, onde você irá inserir os dados de entrada que seu programa irá processar.

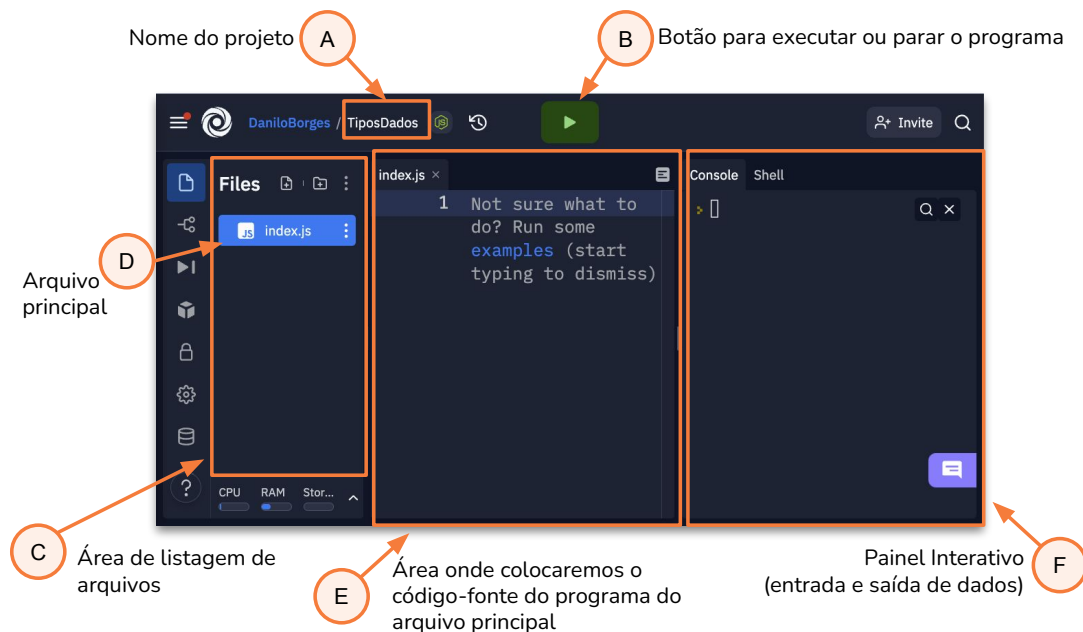


Figura 3.2: Tela de um projeto de programação no Replit. Em destaque as principais áreas do projeto utilizadas nos exercícios e práticas.

Vamos focar na área do painel interativo (Figura 3.2F), o **console**, para observar os tipos de dados. Cada linha será um comando. O comando finaliza com um *enter* do teclado. Vamos ver um exemplo!? Trabalharemos com o comando `typeof()`. Este comando mostra o tipo de dado considerado no *javascript*. Para utilizá-lo colocaremos entre parênteses um dado. Como exemplo usaremos os seguintes comandos:

```
1 typeof (34);  
2 typeof (37.5);  
3 typeof ('A');  
4 typeof (`Hoje`);  
5 typeof ("Tudo bem?");  
6 typeof (5>10);
```

❗ Para melhor organizarmos nosso código, encerraremos este comando `typeof()` com um ponto e vírgula (;).

Verifique se conseguiu reproduzir os mesmos resultados no Replit, como mostra a Figura 3.3. Esses serão os tipos de dados que iremos utilizar em nossas variáveis.

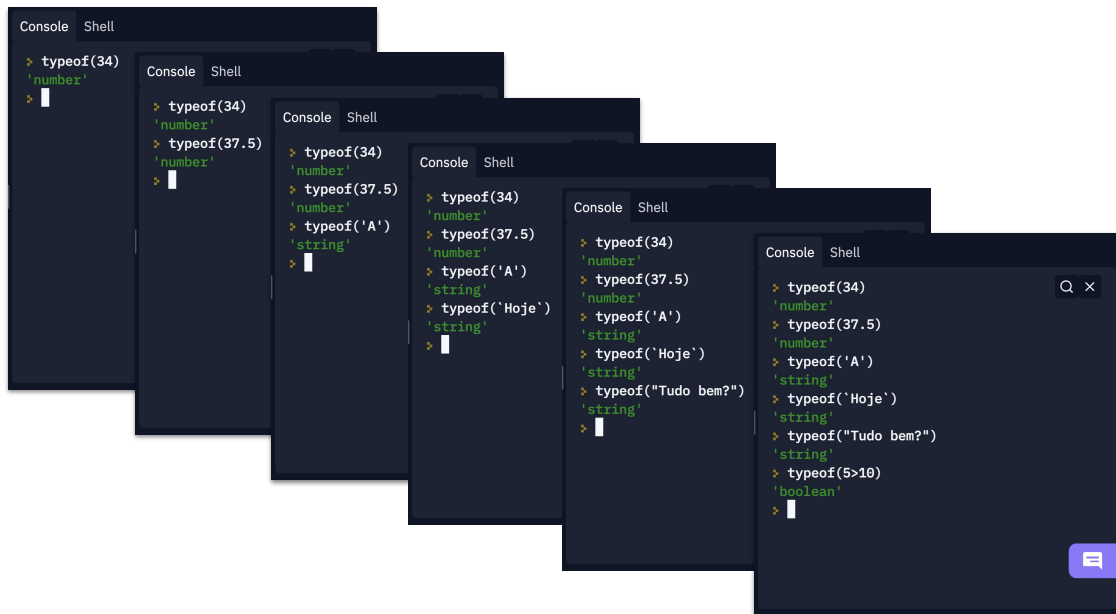


Figura 3.3: Execução de código de verificação de tipo passo a passo no Replit.

3.3 Variáveis

Como o próprio nome diz, variável é algo que pode variar, ou seja, pode ter seu valor alterado. No contexto da programação, as variáveis são identificadores que irão armazenar um tipo de dado recebido em uma atribuição. Como no Exemplo 2.1 onde `valor` armazena o preço de um produto. Pensando neste mesmo exemplo, qual tipo de dado você acha que seria armazenado na variável `valor` no *javascript*? Você está certo se tiver respondido `number`.

3.4 Variáveis no Javascript

Para definir uma variável você precisará utilizar uma palavra-chave: `var` seguindo de um identificador. Utilizando o `valor` do Exemplo 2.1, nosso identificador seria o mesmo nome, então ficaria assim:

```
1 var valor;
```

❗ Para melhor organizarmos nosso código, encerraremos este comando com um ponto e vírgula (;).

Neste exemplo a variável `valor` foi criada, mas não possui nenhum valor atribuído. Para atribuir um valor para variável podemos fazer do seguinte modo:

```
1 var valor = 150;
```

Neste caso o `valor` é criado e inicializado com 150.

Outro modo de fazer que produz o mesmo efeito é este:

```
1 var valor;  
2 valor = 150;
```

Nesse modo a variável é criada (linha 1) e é posteriormente atribuído um valor a ela (linha 2).

O tipo de dado armazenado na variável depende do dado. No caso da variável `valor` seria um `number`, porém, se fosse realizado o seguinte comando:

```
1 var valor = "Hoje";
```

`valor` seria uma `string`. Entendeu!? A última atribuição no `valor` é que definirá seu tipo de dado. Para ficar mais claro, vejamos este outro exemplo:

```
1 var valor;  
2 valor = "Hoje";  
3 valor = 150;  
4 valor = 5>2;
```

Qual seria o tipo de dado de `valor` e qual seria seu valor? Resposta correta: tipo de dado `boolean` e valor `true`. Ou seja, `valor` sempre irá receber o valor *setado* (atribuído) em última instância.

Esta característica da linguagem *javascript* de atribuir a uma mesma variável tipos de dados diferentes lhe configura como uma linguagem fracamente tipada. Ou seja, após declarada a variável pode ser sobrecarregada (atualizada) com novo tipo de dado.

Agora que entendemos o que é um identificador vamos às regras... É como criar uma senha, onde existem algumas regras como: colocar pelo menos uma letra maiúscula, ter pelo menos um número, no mínimo oito caracteres, etc; caso não siga essas regras a senha não é criada. Quando falamos das nossas variáveis se não obedecermos às regras um erro aparecerá e nosso programa irá parar a execução no local do erro. Temos as seguintes regras para os identificadores de variáveis:

1. Devem ser iniciadas sempre por uma letra ou *underline*(`_`);
2. Não devem conter caracteres especiais (`$,-,#,!,?,...`) exceto o *underline*;
3. Não devem conter espaços em branco;

■ **Exemplo 3.1** Identificadores válidos no *javascript*:

```
1 var valor;  
2 var _endereço0;  
3 var data_atual;  
4 var Casa;  
5 var _0_ok;
```

■

■ **Exemplo 3.2** Identificadores inválidos no *javascript*:

```
1 var valorR#;  
2 var _endereço0;  
3 var dia de hoje;  
4 var 01_nome;
```

Os identificadores da linha 1 e da linha 2, ferem a Regra 2. O da linha 3, fere a Regra 3. E o da linha 4 fere a Regra 1.

■

3.5 Variável Constante no Javascript

Existe um tipo de variável que tem uma característica especial, a imutabilidade – qualidade, estado ou condição de imutável. Essas variáveis são declaradas no *javascript* usando a palavra-chave **const** seguido de um identificador. Além das regras de identificadores que a variável deve seguir temos algumas regras adicionais para o **const**:

1. É obrigatória a declaração de uma constante com atribuição de um valor.
2. Não será possível alterar o valor após a criação da variável.

Ou seja, a variável constante garante que seu valor não será alterado de forma alguma, como a própria definição matemática recomenda.

■ **Exemplo 3.3** Variáveis do tipo constante são encontrados no dia a dia como: nome da capital do Piauí, número do π , número de euler e , o seu nome, nome da mãe, seu CPF. ■

Em várias situações é conveniente o uso das constantes. Vamos aprender a criá-las:

```
1 const CAPITAL_PI = "Teresina";  
2 const PI = 3.1416;  
3 const EULER = 2.7182;
```

No código acima, note que utilizei somente letras maiúsculas para definir cada variável constante. Isso não é uma regra, porém, para melhor legibilidade do código é um bom padrão de desenvolvimento quando você tiver uma variável constante colocá-la em caixa alta, assim, você saberá que não pode alterar o valor da variável.

3.6 Resumo

Vimos neste capítulo três tipos de dado que nossas variáveis podem ter no *javascript*: **string**, **number**, e **boolean**; pudemos verificar o tipo de dado utilizando a função **typeof**(). Também vimos como podem ser criadas variáveis e constantes no *javascript* e como são as regras de criação de identificadores.

Para praticar execute as seguintes linhas no **console**:

```
1 var valor = "Teresina";  
2 typeof(valor)  
3 var valor = 3.1416;  
4 typeof(valor)  
5 valor = true;  
6 typeof(valor)
```

Você verá que os resultados de cada **typeof** deve ser, em ordem, **string**, **number** e **boolean**.

3.7 Exercícios

Exercício 3.1 Entre as opções abaixo quais estão relacionadas aos tipos de dados básicos:

- a) Inteiro
- b) Real
- c) Data
- d) Literal
- e) Imagem
- f) Lógico

Exercício 3.2 A afirmação “O tipo de dado Lógico pode assumir somente os valores: verdadeiro ou falso.” é:

- a) Verdadeira.
- b) Falsa.



Exercício 3.3 Qual item define as informações presentes no tipo de dado Literal:

- a) Somente letras.
- b) Somente números.
- c) Somente letras e caracteres especiais.
- d) Letras, números e caracteres especiais



Exercício 3.4 Sobre variáveis no *javascript* marque TODAS as afirmações verdadeiras:

- a) O *javascript* não é uma linguagem fortemente tipada, ou seja, uma variável (**var**) pode mudar seu tipo de dado ao longo do programa.
- b) Uma variável constante (**const**) pode ter seu valor alterado após sua criação.
- c) Pode ser criada uma variável constante (**const**) sem atribuir nenhum valor inicial.
- d) Uma variável (**var**) pode ter seu valor alterado ao longo do programa.



Exercício 3.5 Verifique e indique quais dos nomes identificadores abaixo são válidos, nos inválidos diga o porquê:

- a) voo
- b) 3xct
- c) __praia
- d) __apartamento@
- e) _0_product



Exercício 3.6 Execute o seguinte código em *javascript* e responda cada item.

```
1 var apartamento;  
2 apartamento = "Clara"  
3 apartamento = "350"  
4 apartamento = 350
```

- a) Qual o tipo de dado da variável **apartamento**, na linha 2.
- b) Qual o tipo de dado da variável **apartamento**, na linha 3.
- c) Qual o tipo de dado da variável **apartamento**, na linha 4.

