



**INSTITUTO  
FEDERAL**

Sudeste de  
Minas Gerais

Campus  
Manhuaçu

## Desenvolvimento Web III

### JSON

Prof. Leonardo C. R. Soares - [leonardo.soares@ifsudestemg.edu.br](mailto:leonardo.soares@ifsudestemg.edu.br)

Instituto Federal do Sudeste de Minas Gerais

5 de dezembro de 2024





# Introdução

## Introdução

JSON (JavaScript Object Notation) é um formato leve para intercâmbio de dados. De acordo com a documentação oficial ([json.org](https://json.org)) é um formato de fácil compreensão humana e que facilita o processamento da informação por máquinas.





# Introdução

## Introdução

JSON (JavaScript Object Notation) é um formato leve para intercâmbio de dados. De acordo com a documentação oficial ([json.org](http://json.org)) é um formato de fácil compreensão humana e que facilita o processamento da informação por máquinas.

## Estruturas básicas

Um arquivo JSON é formado por duas estruturas básicas:

- ▶ Uma coleção de pares nome/valor (como um objeto, struct, ou vetor associativo);
- ▶ Uma lista ordenada de valores (como um vetor ou lista).





# JSON Sintaxe

A sintaxe do JSON é derivada da notação de objetos do JavaScript:

- ▶ Dados são pares de nome/valor separados por dois pontos;
- ▶ Dados são separados por vírgula;
- ▶ Chaves definem objetos;
- ▶ Colchetes definem vetores;





# JSON Sintaxe

A sintaxe do JSON é derivada da notação de objetos do JavaScript:

- ▶ Dados são pares de nome/valor separados por dois pontos;
- ▶ Dados são separados por vírgula;
- ▶ Chaves definem objetos;
- ▶ Colchetes definem vetores;

## Exemplo

Em JSON, diferente de JavaScript, a identificação do campo (nome) deve ser uma *string* escrita entre aspas duplas.

```
{"name" : "John", "age" : 30, "car" : null}
```





# JSON Sintaxe

Em JSON, os valores permitidos devem pertencer aos tipos:

- ▶ string
- ▶ number
- ▶ object
- ▶ array
- ▶ boolean
- ▶ null





# JSON Sintaxe

Em JSON, os valores permitidos devem pertencer aos tipos:

- ▶ string
- ▶ number
- ▶ object
- ▶ array
- ▶ boolean
- ▶ null

**Arquivos JSON** devem possuir a extensão *.json* e o tipo MIME *"application/json"*





# JSON x XML

JSON e XML podem ser utilizados para o intercâmbio de dados em servidores web. O exemplo a seguir, retirado de [w3schools.com](https://www.w3schools.com/), apresenta os dois formatos para um mesmo conjunto de dados.







# JSON x XML

## JSON Example

```
{ "employees": [
  { "firstName": "John", "lastName": "Doe" },
  { "firstName": "Anna", "lastName": "Smith" },
  { "firstName": "Peter", "lastName": "Jones" }
]}
```

## XML Example

```
<employees>
  <employee>
    <firstName>John</firstName> <lastName>Doe</lastName>
  </employee>
  <employee>
    <firstName>Anna</firstName> <lastName>Smith</lastName>
  </employee>
  <employee>
    <firstName>Peter</firstName> <lastName>Jones</lastName>
  </employee>
</employees>
```





# PHP

PHP possui funções nativas para a manipulação de JSON.  
Objetos em PHP podem ser convertidos em JSON usando-se a função `json_encode()`





# PHP

PHP possui funções nativas para a manipulação de JSON.  
Objetos em PHP podem ser convertidos em JSON usando-se a função *json\_encode()*

```
<?php
$myObj = new stdClass();
$myObj->name = "John";
$myObj->age = 30;
$myObj->city = "New York";

$myJSON = json_encode($myObj);

echo $myJSON;
?>
```

```
{"name":"John","age":30,"city":"New York"}
```





# PHP

De forma análoga, vetores em PHP podem ser convertidos em JSON usando-se a função `json_encode()`





# PHP

De forma análoga, vetores em PHP podem ser convertidos em JSON usando-se a função *json\_encode()*

```
<?php
$myArr = array("John", "Mary", "Peter",
               "Sally");

$myJSON = json_encode($myArr);

echo $myJSON;
?>
```

```
["John","Mary","Peter","Sally"]
```





# PHP

Para fazermos o caminho inverso, ou seja, manipular dados em JSON no PHP, utilizamos a função `json_decode(string $json, bool $associative = null)`. Quando o parâmetro *associative* é `true`, objetos JSON serão retornados como vetores associativos do PHP.





# PHP

```
<?php
$json = '{"a":1,"b":2,"c":3,"d":4,"e":5}';

var_dump(json_decode($json));
var_dump(json_decode($json, true));

?>

object(stdClass)#1 (5) {
    ["a"] => int(1)
    ["b"] => int(2)
    ["c"] => int(3)
    ["d"] => int(4)
    ["e"] => int(5)
}

array(5) {
    ["a"] => int(1)
    ["b"] => int(2)
    ["c"] => int(3)
    ["d"] => int(4)
    ["e"] => int(5)
}
```





# Exemplo

```
1  <!DOCTYPE html><html lang="pt-br"><head>
2      <meta charset="UTF-8">
3      <meta name="viewport" content="width=device-width, initial-scale=1.0">
4      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.
      css" rel="stylesheet" integrity="sha384-4bw+/aepP/YC94hEpVNVgiZdgIC5+VKNBQNGCHeKRQN
      +PtmoHDEXuppvnDJzQIu9" crossorigin="anonymous">
5      <title>Lista de Filmes</title>
6  </head>
7  <body>
8  <?php
9      $json = file_get_contents("http://www.larback.com.br/Film.JSON");
10     $data = json_decode($json);
11     foreach($data as $filme){
12         echo '<div class="alert alert-primary" role="alert">'. $filme->Title . "(" .
            $filme->Year . ')</div>';
13         foreach ($filme->Images as $imagem)
14             echo "<img src='$imagem' width=10% >";
15         echo "<hr><br>";
16     }
17     ?>
18 </body>
19 </html>
```







## Exercícios

- ▶ O site <https://api.tut tiempo.net/pt/json.html> fornece um arquivo JSON contendo informações sobre o clima. Para acessar uma determinada cidade, basta informar sua latitude e longitude. A url para acessar os dados de Manhuaçu é <https://api.tut tiempo.net/json/?lan=pt&apid=4CYqz4aa4aa8ooA&ll=-20.267,-42.0366>. Analise o arquivo JSON retornado e escreva um *script* em PHP que forneça a previsão do tempo para Manhuaçu.





# Exercícios

- ▶ O site <https://newsapi.org/s/brazil-news-api> fornece uma API para publicação das últimas notícias. O retorno da API é um arquivo no formato JSON. Registre-se para a utilização da API (grátis para desenvolvedores com um limite de 100 acessos por dia) e desenvolva uma pequena aplicação em PHP que apresente as últimas notícias.



