

# **LARAVEL**

**AULA 09**

# APIs JSON

The background of the slide features abstract, overlapping geometric shapes in various colors including light blue, green, cyan, magenta, orange, and red. These shapes are arranged in a way that creates a sense of depth and movement, with some shapes appearing to be layered on top of others. The overall aesthetic is modern and clean.

***Uma API pode ser considerada como o conjunto de funções e rotinas criadas para serem utilizadas por outro software.***

*Isso oferece uma grande vantagem na hora de desenvolver aplicativos multiplataforma, pois desta forma é possível concentrar toda a lógica do negócio em um só lugar, acessível por muitos outros aplicativos ou programas.*



## **API** - *Application Programming Interface*

*(Em português, significa Interface de Programação de Aplicações)*

Trata-se de um conjunto de rotinas e padrões estabelecidos e documentados por uma aplicação A, para que outras aplicações consigam utilizar as funcionalidades desta aplicação A, sem precisar conhecer detalhes da implementação do software.

A aplicação B pode, por exemplo, utilizar qualquer outra linguagem de programação para utilizar os recursos oferecidos pela API.

## ***JSON***

É basicamente um formato leve de troca de informações/dados entre sistemas.

```
{  
  "employees": [{  
    "firstName": "John",  
    "lastName": "Doe"  
  }]  
}
```

XML é um outro formato bastante utilizado.

```
<employees>  
  <employee>  
    <firstName>John</firstName>  
    <lastName>Doe</lastName>  
  </employee>  
</employees>
```

# Rotas da API

www.meusite.com.br/[api/filmes](#)

*routes/api.php*

```
// Rotas da API de Filmes
Route::get('filmes', 'FilmesController@index');
Route::get('filmes/{id}', 'FilmesController@show');
Route::post('filmes', 'FilmesController@store');
Route::put('filmes/{id}', 'FilmesController@update');
Route::delete('filmes/{id}', 'FilmesController@delete');

// Forma simplificada
Route::apiResource('filmes', 'FilmesController');
```

**GET** - Consulta alguma informação (ou um conjunto)

**POST** - Cria alguma informação

**PUT** - Atualiza alguma informação

**DELETE** - Apaga alguma informação

# Métodos do Controller da API

*php artisan make:controller FilmesController --api*

```
class FilmesController extends Controller
{
    public function index()
    {
        // Para consultar todos os filmes
    }

    public function store(Request $request)
    {
        // Para armazenar um novo filme
    }

    public function show($id)
    {
        // Para exibir um filme específico através do ID
    }

    public function update(Request $request, $id)
    {
        // Para atualizar as informações de um filme através do ID
    }

    public function destroy($id)
    {
        // Para excluir um filme através do ID
    }
}
```

## Respostas (RESPONSE)

```
return response()->json($filme, CODIGO_HTTP);
```

200: OK. The standard success code and default option.

201: Object created. Useful for the store actions.

204: No content. When an action was executed successfully, but there is no content to return.

206: Partial content. Useful when you have to return a paginated list of resources.

400: Bad request. The standard option for requests that fail to pass validation.

401: Unauthorized. The user needs to be authenticated.

403: Forbidden. The user is authenticated, but does not have the permissions to perform an action.

404: Not found. This will be returned automatically by Laravel when the resource is not found.

500: Internal server error. Ideally you're not going to be explicitly returning this, but if something unexpected breaks, this is what your user is going to receive.

503: Service unavailable. Pretty self explanatory, but also another code that is not going to be returned explicitly by the application.



## ***Algumas APIs que outros já criaram =>***

Site com uma biblioteca de APIs encontradas por aí

**<https://any-api.com/>**

Viacep: Para pesquisar um endereço a partir de um CEP

**<https://viacep.com.br/ws/01001000/json/>**

NY Times: Para obter as últimas notícias publicadas

**<https://api.nytimes.com/svc/topstories/v2/home.json?api-key=9502f2de0b2d4369bb6ae9873ed79ecd>**

Para localizar informações de um CNPJ

**<https://www.receitaws.com.br/v1/cnpj/27865757000102>**

The background features abstract, overlapping geometric shapes in various colors including light blue, green, cyan, magenta, orange, and red. These shapes are arranged in a way that creates a sense of depth and movement, framing the central text.

**ATÉ A  
PRÓXIMA!**