

Response Entity e Status de Resposta



DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

<LAB365>

AGENDA

- Response Entity
- Resposta padrão
- Response Entity Exemplos
- Status de Resposta
- Status de Sucesso
- Status de Informação e Direcionamento
- Status de Erro do Client-Side
- Status de Erro no Server-Side

Response Entity

Já vimos alguns dos status que uma API pode retornar ao ser chamada, no Spring Boot utilizamos o `ResponseEntity`.

Essa classe nos permite criar respostas customizadas para cada endpoint que temos, assim podemos gerar uma resposta de forma simples.

Para que uma API funcione de forma correta precisamos ter os erros previstos e testados para que os erros que ocorrem não impactem o Cliente da API.

Resposta padrão

Para criarmos um body devemos ter principalmente o status e a classe que define a resposta. Vamos criar uma classe para compor todas as respostas que podemos ter.

```
public class DefaultResponse<T> {  
    private String status;  
    private LocalDateTime timeStamp;  
    private T data;  
}
```

Essa resposta tem que ser retornada no body de resposta.

Response Entity Exemplos

```
@GetMapping("/status")
ResponseEntity<String> ok() {
    return ResponseEntity.status(HttpStatus.OK).body("OK");
}
```

```
@GetMapping("/notfound")
ResponseEntity<String> notfound() {
    return ResponseEntity.notFound().build();
}
```

```
@GetMapping("/badrequest")
ResponseEntity<String> badRequest() {
    return ResponseEntity.badRequest().body("Bad request");
}
```

INTERVALO DE AULA

DEV!

Finalizamos o nosso primeiro período de hoje. Que tal descansar um pouco?!

Nos vemos em 20 minutos.

Início: 20:20

Retorno: 20:40



Status de Resposta

O Response Entity é onde colocamos o status da mensagem e é onde colocamos a classe de resposta.

Vamos olhar um pouco mais sobre os status que devemos adicionar ao nosso programa

Status de Sucesso

Status de Sucesso na requisição:

200 OK — Código mais utilizado, e que indica que a requisição foi processada com sucesso.

201 Created — Indica que a requisição foi bem sucedida e que um novo registro foi criado como resultado. Resposta utilizada em requisições do método POST.

202 Accepted — Indica que a requisição foi bem sucedida porém não há uma resposta do servidor para essa requisição

204 No Content — Indica que a requisição foi bem sucedida porém não há dados para serem retornados. Indica que não há nada de errado com a requisição, no caso de um CRUD pode mostrar que o sistema está com o banco vazio.

Status de Informação e Direcionamento

São utilizadas com a finalidade de informar o cliente sobre mudanças na requisição e o redirecionamento para uma nova URL

301 Moved Permanently — Informa que o recurso A agora é o recurso B, de forma que quando o cliente solicitar o recurso A ele será automaticamente encaminhado ao recurso B.

304 Not Modified — Resposta utilizada em cenários de cache. Informa ao cliente que a resposta não foi modificada, e nesse sentido, o cliente pode usar a mesma versão em cache da resposta.

307 Temporary redirect — Se trata de um redirecionamento de uma página para outro endereço, porém que é com caráter temporário, e não permanente.

Status de Erro do Client-Side

São status retornados caso a requisição recebida pelo servidor tenha sido montada de forma incorreta:

400 Bad Request — Essa resposta significa que o servidor não consegue entender sua requisição, pois existe uma sintaxe ou estrutura inválida.

401 Unauthorized — Erro que informa que existe uma camada de segurança no recurso solicitado ao servidor, e que você não está utilizando as credenciais corretas nessa requisição. Lembrando que as credenciais podem ser algo como usuário e senha, token, etc. E tudo vai depender da API que você estará consumindo.

403 Forbidden — Nesse caso as credenciais do cliente são reconhecidas, porém o servidor informa que ele não tem os direitos de acesso necessários para aquele recurso.

Status de Erro do Client-Side

404 Not Found — Código que informa que o servidor não encontra o recurso solicitado pelo cliente.

429 Too Many Requests — Não é tão comum, mas pode ser utilizado para informar ao cliente que ele excedeu o limite permitido de requisições.

Status de Erro no Server-Side

A requisição foi feita corretamente pelo lado do cliente, porém houve um erro de processamento no servidor.

500 Internal Server Error — Erro mais genérico desse grupo é muito utilizado. Ele informa que o servidor encontrou um cenário inesperado de erro que não soube tratar, e por isso não conseguiu retornar uma resposta na requisição do cliente.

503 Service Unavailable — Erro normalmente utilizado para informar que o servidor está fora do ar, em manutenção ou sobrecarregado.

AVALIAÇÃO DOCENTE

O que você está achando das minhas aulas neste conteúdo?

[Clique aqui](#) ou escaneie o QRCode ao lado para avaliar minha aula.

Sinta-se à vontade para fornecer uma avaliação sempre que achar necessário.





DEVinHouse

Parcerias para desenvolver a sua carreira

OBRIGADO!



<LAB365>