

Francisco das Chagas Torres dos Santos

Setembro de 2022

Neste material, meu objetivo é ir além do gabarito das questões. Nos comentários, eu aprofundo mais no assunto, sem me distanciar do que a questão está pedindo. Isso me dá mais conhecimento sobre o assunto e me faz ver a questão abordada por diferentes ângulos.

Quando há bastantes questões sobre o assunto, opto por questões de provas do ano corrente e das bancas CESPE, FCC e FGV. No entanto, em alguns casos, é necessário pegar provas de anos anteriores e de outras bancas.

Devemos esquecer questões muito antigas, pois TI atualiza a todo instante e as bancas também podem mudar a pegada, a forma de cobrar o conteúdo.

Noções de arquitetura de sistemas

Swagger - Questões

1. Ano: 2022 Banca: FCC Órgão: TRT - 4ª REGIÃO (RS) Prova: FCC - 2022 - TRT - 4ª REGIÃO (RS) - Analista Judiciário - Especialidade: Tecnologia da Informação

Para documentar uma API acessível externamente a partir de um cenário de microsserviços, um Analista utilizou a especificação Swagger. Para cada operação RESTful na API, ele adicionou uma anotação A, juntamente com anotações B no método Java correspondente, para descrever a operação e suas respostas de erro esperadas. As anotações A e B são, respectivamente,

- a) @describeOperation e @errorResponse.
- b) @restOperation e @restResponse.
- c) @getInfoOperation e @getErrorResponse.
- d) @ApiOperation e @ApiResponse.
- e) @swaggerOperation e @swaggerResponse.

Resposta: Letra D

Comentários

Com @ApiResponse e @ApiResponses é possível especificar os códigos e as mensagens de retorno diretamente no controller. Estas configurações serão válidas para os endpoints onde estiverem definidos. Onde não estiver, será utilizado o padrão.

```
@ApiResponses(value = {  
    @ApiResponse(code = 200, message = "Retorna a lista de pessoa"),  
    @ApiResponse(code = 403, message = "Você não tem permissão para acessar este recurso"),  
    @ApiResponse(code = 500, message = "Foi gerada uma exceção"),
```

```
})  
@RequestMapping(value = "/pessoa", method = RequestMethod.GET, produces="application/json")  
public List<Pessoa> Get() {  
    return _pessoaRepository.findAll();  
}
```

O @ApiOperation descreve o endpoint.

```
@ApiOperation(value = "Retorna uma lista de pessoas")  
@ApiResponse(value = {  
    @ApiResponse(code = 200, message = "Retorna a lista de pessoa"),  
    @ApiResponse(code = 403, message = "Você não tem permissão para acessar este recurso"),  
    @ApiResponse(code = 500, message = "Foi gerada uma exceção"),  
})  
@RequestMapping(value = "/pessoa", method = RequestMethod.GET, produces="application/json")  
public List<Pessoa> Get() {  
    return _pessoaRepository.findAll();  
}
```

Fonte: <https://www.treinaweb.com.br/blog/documentando-uma-api-spring-boot-com-o-swagger>

2. Ano: 2022 Banca: FCC Órgão: TRT - 4ª REGIÃO (RS) Prova: FCC - 2022 - TRT - 4ª REGIÃO (RS) - Técnico Judiciário - Área Apoio Especializado - Especialidade: Tecnologia da Informação

Em um código Swagger escrito em JSON, a URL Base é formada por

- a) host, baseForms e schemes.
- b) consumes, host e produces.
- c) host, basePath e schemes.
- d) produces, schemes e baseForms.
- e) application, get e basePath.

Resposta: Letra C

Comentários

As APIs REST têm uma URL base a qual os caminhos do endpoint são anexados. Ela é definida por schemes, host e basePath.

```
host: petstore.swagger.io  
basePath: /v2  
schemes:  
- https
```

Todos os endpoints/caminhos da API são relativos a URL base.

GET <https://petstore.swagger.io/v2/pets/findByStatus?status=available>

Temos:

- operation: GET
- scheme: https
- host; petstore.swagger.io
- basePath: /v2
- path: *petsfindByStatus*
- query parameter: status=available

Os **schemes** são protocolos de transferências usados pela API. O Swagger suporta os schemes http, https e WebSocket -ws e wss. Os schemes podem ser especificados em uma lista YAML conforme abaixo:

```
schemes:  
- http  
- https
```

Ou através da sintaxe literal de matriz

```
schemes: [htt, https]
```

O nome do domínio ou endereço IP (IPv4) é o **host**. As portas padrões dos schemas são 80 para HTTP e 443 para HTTPS. Se a porta for diferente da padrão, incluir no host. Em host vai ter apenas o host, sem http(s):// ou subcaminhos. Exemplos:

- api.example.com
- example.com:8089
- 93.184.216.34
- 93.184.216.34:8089

Incorreto:

- http://api.example.com
- example.com/api/v1

Se o host não for especificado, supõe-se que seja o mesmo em que a documentação da API está sendo veiculada.

O **basePath** começa com /. Exemplos válidos:

- /
- /v2
- /api/v2

Vários hosts são suportados no OpenAPI 3.0, o OpenAPI 2.0 suporta apenas um host por especificação de API ou dois se você considerar HTTP e HTTPS como hosts diferentes.

Para mais, consulte:

- <https://swagger.io/docs/specification/2-0/api-host-and-base-path/>
- <https://swagger.io/docs/specification/api-host-and-base-path/>

3. Ano: 2022 Banca: FCC Órgão: TRT-23 MT Prova: Analista Judiciário Área Apoio Especializado Especialidade Tecnologia da Informação

No componente controller de uma aplicação criada com recursos Spring Boot foram incluídas as anotações abaixo a um método

REST.

```
@ApiOperation(value = "Obter a lista de processos no sistema", response = Iterable.class, tags = "getProcessos")
```

```
@ApiResponses(value = {  
    @ApiResponse(code = 200, message = "Sucesso|OK"),  
    @ApiResponse(code = 401, message = "Não autorizado"),  
    @ApiResponse(code = 403, message = "Proibido"),  
    @ApiResponse(code = 404, message = "Não encontrado") })
```

```
@RequestMapping(value = "/getProcessos")  
public List<Processos> getProcessos() {  
    return processos;  
}
```

@ApiOperation e @ApiResponses são anotações

- a) MapStruct.
- b) Hibernate.
- c) Spring Eureka.
- d) Spring Zuul.
- e) Swagger

Letra E

Comentários

A resposta desta questão é Swagger e para entender as anotações, veja o comentário da questão 1. Aqui vamos, comentar brevemente as demais alternativas.

- a) MapStrut é um mapeador Java Bean com anotações: @Mapper e @Mapping.
- b) Hibernate é um ORM, ou seja, framework para persistência de dados. É a implementação do JPA. Algumas anotações: @Entity, @Id, @IdClass, @EmbeddId, @Transient e outras.
- c) Spring Eureka é para registrar e descobrir serviços (serviço de discovery) de arquitetura de Microservices, há o Eureka Client e Eureka Server. Anotação @EnableEurekaServe para tornar

a aplicação um Eureka Server. @EnableDiscoveryClient para fazer o registro da aplicação no Eureka.

- d) Spring Zuul é uma solução de roteamento dinâmico que possibilita resiliência e segurança para as aplicações. Anotações @EnableZuulProxy

4. Ano: 2022 Banca: IESES Órgão: CREA-SC Prova: IESES - 2022 - CREA-SC - Analista de Sistemas

Com relação ao Swagger, é correto afirmar, EXCETO:

- a) É uma aplicação que auxilia os desenvolvedores a definir, criar, documentar e consumir APIs REST.
- b) Fornece ferramentas para auxiliar na definição do arquivo de configuração e interagir com API através das definições do arquivo de configuração.
- c) É composto de um arquivo de configuração, que pode ser definido em YAML ou JSON.
- d) É uma aplicação desenvolvida pela Microsoft que auxilia os desenvolvedores a definir, criar, documentar e consumir APIs REST.

Letra E
