

BRUNNA CROCHES

PARTE 3

API | REST | SPA



Australian Pelican | Artist Image :Broinowski, Gracius Joseph | Dates:1837-1913

Guia iniciante:
API

SUMMARY

REST | API | SPA



O QUE É JSON?

3.4

QUESTÕES
RESOLVIDAS DE
REST

3.5

3.4 O que é JSON?

JSON

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

Mas **JSON significa JavaScript Object Notation**, ou seja, só posso usar com JavaScript correto? Na verdade não e alguns ainda caem nesta armadilha.

O JSON além de ser um formato leve para troca de dados é também muito simples de ler.

Mas quando dizemos que algo é simples, é interessante compará-lo com algo mais complexo para entendermos tal simplicidade não é? Neste caso podemos comparar o JSON com o [formato XML](#).

JSON x XML

XML	JSON
<pre><note> <to>Tove</to> <from>Jani</from> <heading>Reminder</h eading> <body>Don't forget me this weekend! </body> </note></pre>	<pre>{ "id":1, "nome":"Alexandre Gama", "endereco":"R. Qualquer" }</pre>

Bom, é notável a diferença. Visualmente o segundo trecho (em JSON) é mais fácil de ler. Podemos listar algumas outras vantagens:

Vantagens do JSON:

- Leitura mais simples
- Analisador(parsing) mais fácil
- JSON suporta objetos! Sim, ele é tipado!
- Velocidade maior na execução e transporte de dados
- Arquivo com tamanho reduzido
- Quem utiliza? Google, Facebook, Yahoo!, Twitter...

Qual biblioteca usar?

Existem diversas bibliotecas para trabalharmos com [JSON](#) e [Java](#).

Usaremos no nosso estudo o `json.jar` que você pode baixar tranquilamente neste link.

Questões REST

HTML Content

Arquitetura REST

Na prática, o que significa dizer que REST é stateless?

Significa que cada comunicação é independente, cada requisição é autossuficiente, enviando todos os dados de uma só vez, salvando as informações da requisição para reduzir o consumo de dados

Significa que cada comunicação depende especificamente da primeira requisição enviada, que geralmente contém informações sobre a autenticação do cliente

Significa que cada comunicação é independente, cada requisição é autossuficiente, enviando todos os dados de uma só vez, sem salvar informações da requisição

Stateless significa "requisição sem estado", ou seja, é uma requisição feita para o web service que de forma alguma irá trazer um status code na resposta

HTML Content

Verbo inválido

Qual verbo ou método HTTP listado abaixo não é válido para ser utilizado em um web service que implementa a arquitetura REST?

SEARCH

POST

PATCH

GET

HTML Content

Conceitos sobre a arquitetura REST

Tendo em vista que uma requisição feita a um web service RESTful pode retornar o resultado em diversos formatos, como JSON, XML, texto plano, dentre outros, qual o motivo do formato JSON ser atualmente o mais adotado?

Por não permitir alterações na sua estrutura, tendo campos fixos, o que facilita a implementação

Por seu formato leve, legível e sua fácil interpretação por diversas tecnologias

Por ser orientado a objetos, facilitando sua implementação

Por ser o mais indicado em conjunto aos métodos HTTP

HTML Content

Diferenciando os termos

Qual é a diferença entre os termos REST e RESTful?

Não existe diferença, são sinônimos

Um web service REST retorna sempre JSON, enquanto RESTful pode retornar JSON e XML

REST é a arquitetura, já RESTful é uma aplicação que implementa a arquitetura REST

REST utiliza apenas 2 verbos HTTP (GET e POST), já RESTful utiliza todos os verbos HTTP

HTML Content

Conceito incorreto

Qual das alternativas abaixo apresenta um conceito INCORRETO a respeito da arquitetura REST?

Todas alternativas são incorretas a respeito da arquitetura REST

Deve manter sempre na sessão do navegador informações da requisição anterior

Não é obrigatório que cada requisição seja independente, podendo recuperar informações da requisição anterior para dar sequência a ação (autenticação, por exemplo)

Não pode ser implementada em ambiente seguro com o Protocolo HTTPS

HTML Content

Ações da aplicação

A arquitetura REST utiliza os verbos ou métodos HTTP para informar ao web service qual ação será tomada na aplicação. Com base nessa informação, qual das opções abaixo representa, respectivamente, a sequência correta para recuperar um recurso do web service, de forma que o usuário veja os dados, e em seguida, solicite sua remoção?

FIND e DELETE

GET e DESTROY

GET e DELETE

GET, REMOVE

HTML Content

Migrar um webservice

Você foi contratado pela empresa de domínio mybusiness.com o intuito de migrar um web service para a arquitetura REST. Após aceitar esse desafio, em uma reunião com o gerente de projetos, você ouve a seguinte afirmação: "Essa migração será rápida, pois para um web service ser RESTful, basta retornar JSON".

Com base na afirmação do gerente de projetos, podemos dizer que:

A afirmação está correta, pois a mudança chave para tornar um web service RESTful é padronizar o retorno em JSON, facilitando o trabalho de qualquer aplicação que consumirá esse web service

A afirmação está correta, pois um web service RESTful segue todos os padrões impostos pela arquitetura REST, o que se limita exatamente ao retorno padronizado em JSON

A afirmação não está correta. Embora o tipo de retorno JSON como padrão seja uma característica OBRIGATÓRIA da arquitetura REST, não basta apenas retornar neste formato, e sim informar a ação desejada na URI do web service e manter informações na sessão, otimizando o tempo de resposta

Não está correta, pois um web service RESTful segue todos os padrões impostos pela arquitetura REST, inclusive NÃO definindo exclusividade do retorno em JSON

Método de remoção

Qual método HTTP é indicado para realizar a remoção de um recurso em um web service que implementa a arquitetura REST?

POST

DELETE

REMOVE

DEL

Métodos HTTP

Em um web service já em funcionamento, para buscar informações a respeito do perfil (profile) do usuário de identificação 1, enviamos uma requisição GET para a URL `http://www.mybusiness.com/api/users/1/profile`.

Tendo em vista que o método GET não envia informações no corpo da requisição, como o web service faz para identificar de qual usuário estamos buscando pelo perfil?

O web service identifica através da query string enviada na URL, onde é mapeado nas rotas do web service, que a mesma possui um parâmetro de valor 1 entre os recursos users e profile

O web service identifica através da URI e sempre recolhe toda as informações de valores inteiros, convertendo para a rota identificada com essa característica

Não é possível realizar essa consulta com o método ou verbo HTTP GET, pois para isso é necessário o método ou verbo HTTP LIST, que envia a identificação do usuário no corpo da requisição

Não é possível realizar essa consulta com o método ou verbo HTTP GET, pois para isso é necessário o método ou verbo HTTP SEARCH, que identifica todo valor inteiro da URI e converte para a identificação do recurso

HTML Content

Trabalhando com métodos

Qual método HTTP é indicado para realizar a atualização completa de um recurso em um web service que implementa a arquitetura REST?

POST

PUT

REFRESH

PATCH

HTML Content

Conceitos REST

Qual das afirmações abaixo NÃO é verdadeira com base no conceito da arquitetura REST?

Todo web service RESTful implementa a arquitetura REST

Todo web service é REST

Comunica através do protocolo HTTP

Utiliza os métodos HTTP para indicar a ação desejada

HTML Content

Trabalhando com métodos ou verbos HTTP

Das alternativas abaixo, qual delas não é considerada um método ou verbo HTTP?

DELETE

GET

RETRIEVE

PUT

HTML Content

Diferenciando os termos

É correto dizer que os termos REST e RESTful possuem o mesmo significado?

Não, pois REST é a arquitetura, já RESTful é uma aplicação que implementa a arquitetura REST

Não, o termo REST utiliza apenas 2 verbos HTTP (GET e POST), enquanto RESTful utiliza todos os verbos HTTP

Sim, são sinônimos, porém, o termo RESTful é mais conhecido na linguagem Java

Sim, o termos possuem o mesmo significado, porém RESTful é mais

conhecido no âmbito de aplicações WEB

HTML Content

Conceitos sobre a arquitetura REST

Com relação ao conceito da arquitetura REST, podemos afirmar que:

Possui o padrão de retornar os dados sempre no formato JSON

É uma arquitetura criada para ser utilizada somente em aplicações desenvolvidas em JavaScript

É uma forma de implementar web services, uma arquitetura que utiliza o protocolo HTTP e seus verbos

Possui uma única URI específica para identificar as diversas ações executadas pela aplicação

ABOUT ME



Brunna Croches

Developer Full Stack

Brunna Croches é Dev FullStack, advogada e empreendedora.

Apaixonada por tech, vem adquirindo vasto conhecimento na área.

Desenvolveu projetos ricos em diversidade, buscando captar as próximas tendências e necessidades do mercado.

Neste e-book você aprenderá ou recapitulará de forma simplificada e otimizados conceitos de programação feito por ela.

let's share

CONTACT



Brunna Croches

Developer Full Stack



brunnacroches.dev



linkedin.com/brunnacroches



github.com/brunnacroches



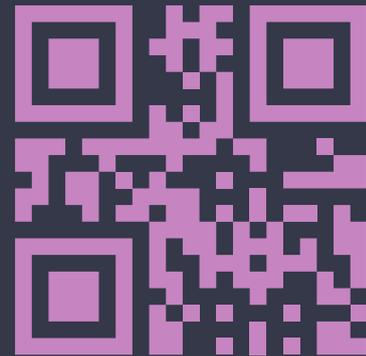
@brunnacroches.dev



discord.com/brunnacroches



brunnacroches@gmail.com



let's share