Creating and Deleting Resources

Kevin Dockx
ARCHITECT

@KevinDockx https://www.kevindockx.com

Coming Up

Creating resources

Supporting additional content-type values and input formatters

Deleting resources

Method safety and method idempotency

Method Safety and Method Idempotency

A method is considered safe when it doesn't change the resource representation

A method is considered idempotent when it can be called multiple times with the same result

Method Safety and Method Idempotency

HTTP Method	Safe?	Idempotent?
GET	yes	yes
OPTIONS	yes	yes
HEAD	yes	yes
POST	no	no
DELETE	no	yes
PUT	no	yes
PATCH	no	no

Method safety and idempotency help decide which method to use for which use case

Demo

Creating a Resource

Demo

Creating a Child Resource

Demo

Creating Child Resources Together with a Parent Resource

Demo

Creating a Collection of Resources

Demo

Working with Array Keys and Composite Keys

Demo

Handling POST to a Single Resource

Demo

Supporting Additional Content-Type Values and Input Formatters

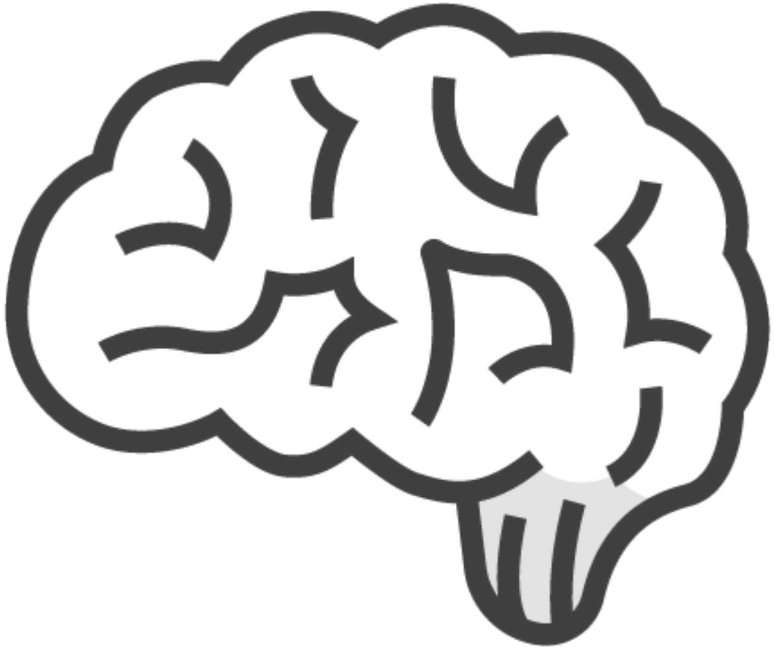
Demo

Deleting a Resource

Demo

Deleting a Resource with Child Resources

Deleting Collection Resources



Sending a DELETE request to a collection resource like http://host/api/authors is allowed

It's rarely implemented because it can be very destructive

Summary

A method is considered safe when it doesn't change the resource representation

A method is considered idempotent when it can be called multiple times with the same result

POST is used to create resources

- Unsafe
- Not idempotent
- 201 Created
- Location header

Summary

Use the content-type header to signify the media type of the request body

DELETE is used to delete resources

- Unsafe
- Idempotent
- 204 No content