

Application Programming Interfaces (APIs)

Code-Along

Ryan Day (IC)

10/20/2017

Agenda

- 1:00 - 1:45
 - Understanding APIs
 - Browsing and viewing REST APIs
 - Tools for using APIs
- 1:45 - 1:50
 - Break
- 1:50 - 2:30
 - Coding with APIs

Introductions



Lab 1

**View an API in the
browser**

API/Web Service Styles: SOAP vs. REST

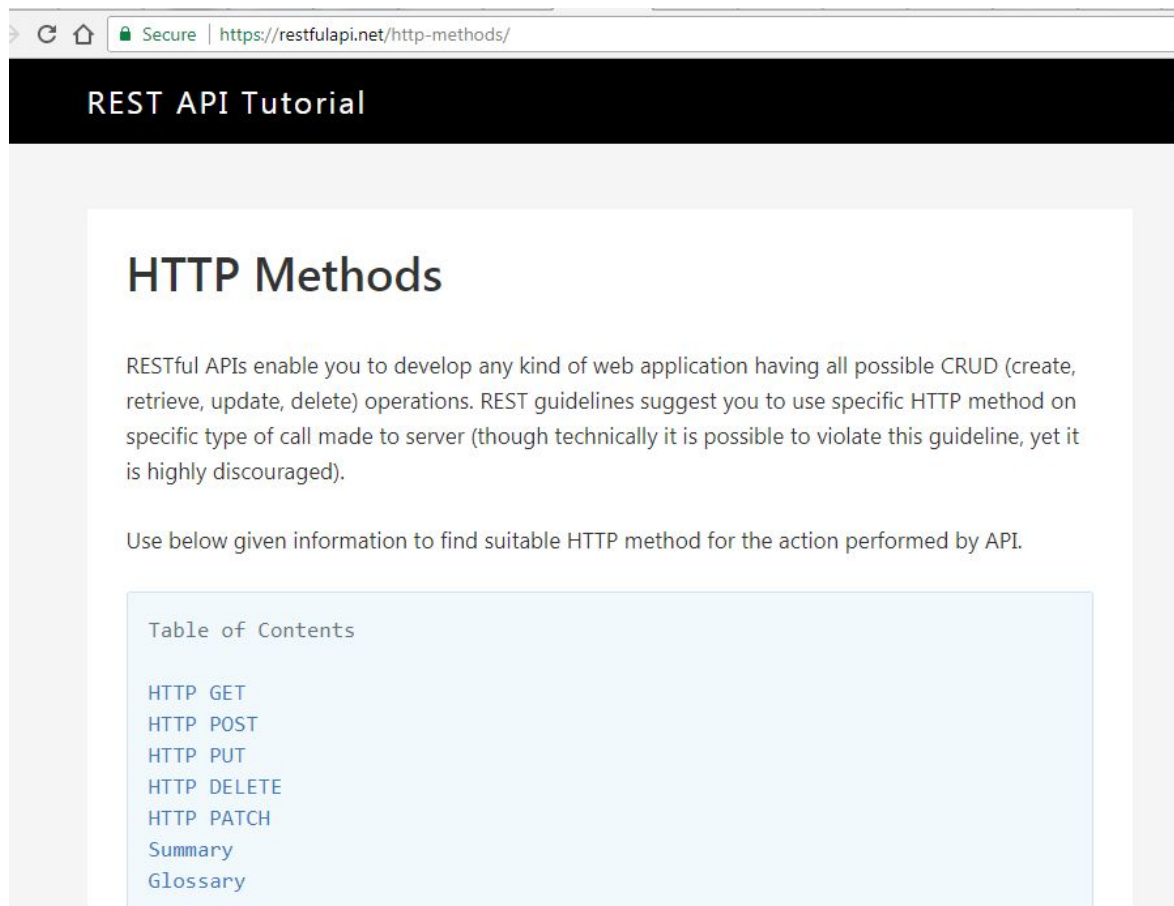
#	SOAP	REST
1	A XML-based message protocol	An architectural style protocol
2	Uses WSDL for communication between consumer and provider	Uses XML or JSON to send and receive data
3	Invokes services by calling RPC method	Simply calls services via URL path
4	Does not return human readable result	Result is readable which is just plain XML or JSON
5	Transfer is over HTTP. Also uses other protocols such as SMTP, FTP, etc.	Transfer is over HTTP only
6	JavaScript can call SOAP, but it is difficult to implement	Easy to call from JavaScript

<http://www.differencebetween.net/technology/internet/difference-between-rest-and-soap/>

Understanding REST



HTTP Methods (verbs)



<https://restfulapi.net/http-methods/>

HTTP Methods Pre-Test

Before you get started, make a wild guess on what the HTTP methods are used for!

I would like to read data from an API.

1 point

- ☐ HTTP GET
- ☐ HTTP PUT
- ☐ HTTP POST
- ☐ HTTP PATCH
- ☐ HTTP DELETE

I would like to create a new record with an API.

1 point

https://docs.google.com/forms/d/e/1FAIpQLSdQkaY3IBGljN2oNjjrgxKwur2E7zK2gonp8Uw8p4RoH4fZSQ/viewform?c=0&w=1&usp=mail_form_link

Understanding REST



HTTP Response Codes (status codes)

REST API Tutorial

HTTP Status Codes

REST APIs use the **Status-Line** part of an HTTP response message to inform clients of their request's overarching result. RFC 2616 defines the **Status-Line** syntax as shown below:

```
Status-Line = HTTP-Version SP Status-Code SP Reason-Phrase CRLF
```

HTTP defines forty standard status codes that can be used to convey the results of a client's request. The status codes are divided into the five categories presented below.

CATEGORY	DESCRIPTION
1xx: Informational	Communicates transfer protocol-level information.
2xx: Success	Indicates that the client's request was accepted successfully.
3xx:	Indicates that the client must take some additional action in order to

<https://restfulapi.net/http-status-codes/>

Understanding JSON



JS Timing

JS Cookies

JS AJAX

AJAX Intro

AJAX XMLHttpRequest

AJAX Request

AJAX Response

AJAX XMLHttpRequest File

AJAX PHP

AJAX ASP

AJAX Database

AJAX Applications

AJAX Examples

JS JSON

JSON Intro

JSON Syntax

JSON - Introduction

◀ Previous

JSON: **J**ava**S**cript **O**bject **N**otation.

JSON is a syntax for storing and exchanging data.

JSON is text, written with JavaScript object notation.

Exchanging Data

Lab 2

Exploring the Prototype City Pairs API

Lab 3



Tour of the GSA APIs

API Tools Demonstration

GSA Approved Tools

Postman

File Edit View Collection History Help

Runner Import Builder Team Library

Filter

History Collections

All Me Team

Postman Echo 37 requests

API Code-Along 8 requests

POST Per Diem API - Valid

GET Per Diem API - Invalid

GET eMuseum - Valid

GET Job Search - Valid

GET SF Tool - Valid

GET City Pairs - Valid

GET JSON placeholder

GET GEAR IT Standards - with ID

Per Diem API - Valid City Pairs - Valid

No Environment

Examples (0)

GET https://api.gsa.gov/travel/citypairs/v0/airfares?award_year=2017&origin_airport_abbrev=ABQ&api_key=DEM... Params Send Save

Authorization Headers Body Pre-request Script Tests Cookies Code

Type No Auth

Body Cookies Headers (22) Tests Status: 200 OK Time: 1171 ms Size: 85.44 KB

Pretty Raw Preview JSON Save Response

```
1 {
2   "message": "This is a prototype API. These results are for demonstration purposes only.",
3   "result": [
4     {
5       "ID": 24169,
6       "ITEM_NUM": "6",
7       "AWARD_YEAR": "2017",
8       "ORIGIN_AIRPORT_ABBREV": "ABQ",
9       "DESTINATION_AIRPORT_ABBREV": "ATL",
10      "ORIGIN_CITY_NAME": "ALBUQUERQUE",
11      "ORIGIN_STATE": "NM",
12      "ORIGIN_COUNTRY": "USA",
13      "DESTINATION_CITY_NAME": "ATLANTA",
14      "DESTINATION_STATE": "GA",
15      "DESTINATION_COUNTRY": "USA",
16      "AIRLINE_ABBREV": "AA",
17      "AWARDED_SERV": "C",
18      "PAX_COUNT": "3344",
19      "YCA_FARE": 250,
```


SoapUI 5.3.0

File Project Suite Case Step Tools Desktop Help

Empty SOAP REST Import Save All Forum Trial Preferences Proxy

Search Forum Online Help

Navigator

- Projects
 - Prototype City Pairs
 - https://api.gsa.gov
 - {id} [/travel/citypairs/v0/airfares/{id}]
 - airfares [/travel/citypairs/v0/airfares]
 - Method 1
 - Request 1
 - https://cityPairsPrototypeAPI.app.cloud.gov
 - https://cityPairsPrototypeAPI.app.cloud.gov
 - Load test non-API.DATA.GOV
 - https://api.gsa.gov TestSuite
 - REST Project 1
 - REST Project 2
 - REST cloud sandbox

Request 1

Method: GET Endpoint: https://api.gsa.gov Resource: /travel/citypairs/v0/airfares Parameters: ?award_year=2017&origin_airport_abbrev=ABQ&destination_airport_abbrev=MCI&api_key

Name	Value	Style	Level
award_year	2017	QUERY	RESOU...
origin_airport_abb...	ABQ	QUERY	RESOU...
destination_airpor...	MCI	QUERY	RESOU...
api_key	DEMO_K...	QUERY	RESOU...

Raw Request

```
1 {
2   "message": "This is a prototype API. These results are for demonstration purposes."
3   "result": [ {
4     "ID": 24182,
5     "ITEM_NUM": "19",
6     "AWARD_YEAR": "2017",
7     "ORIGIN_AIRPORT_ABBREV": "ABQ",
8     "DESTINATION_AIRPORT_ABBREV": "MCI",
9     "ORIGIN_CITY_NAME": "ALBUQUERQUE",
10    "ORIGIN_STATE": "NM",
11    "ORIGIN_COUNTRY": "USA",
12    "DESTINATION_CITY_NAME": "KANSAS CITY",
13    "DESTINATION_STATE": "MO",
14    "DESTINATION_COUNTRY": "USA",
15    "AIRLINE_ABBREV": "WN",
16    "AWARDED_SERV": "N",
17    "PAX_COUNT": "2026",
18    "YCA_FARE": 171,
19    "XCA_FARE": 126,
20    "BUSINESS_FARE": 0,
21    "ORIGIN_AIRPORT_LOCATION": "",
22    "DESTINATION_AIRPORT_LOCATION": "",
23    "ORIGIN_CITY_STATE_AIRPORT": "ALBUQUERQUENM-ABQ",
24    "DESTINATION_CITY_STATE_AIRPORT": "KANSAS CITYMO-MCI",
25    "EFFECTIVE_DATE": "2016-10-01T00:00:00.000Z",
26    "EXPIRATION_DATE": "2017-09-30T00:00:00.000Z"
27  } ]
28 }
```

Required: ☐ Sets if parameter is required
Type:
Options:

Request Properties Request Params

Property	Value
Name	Request 1
Description	

Properties

response time: 2458ms (742 bytes)

SoapUI log http log jetty log error log wsrm log memory log

Headers (23) Attachments (0) SSL Info (2 certs) Representations (1) Schema (conflicts) JMS (0)

1:1

Lab 4

Solving Business Problems with APIs

BREAK



Back at 1:50

Lab 5

**Use HTML and
JavaScript to consume
APIs.**

Lab 6

**Using Google Scripts to
pull API data.**

Wrapping Up



Q&A and Next Steps

Join the API Working Group

Join our mailing list and attend quarterly meetings of the API Working Group.

Email Joe Castle (joseph.castle@gsa.gov) to be added to the list.

Thanks for your time!

Ryan Day

Corporate IT Services (IC)

816-823-5322

ryan.day@gsa.gov

Extra Slides

GSA API Standards



**Best practices for API
providers**

GSA API Standards

This document captures GSA's recommended best practices, conventions, and standards for Application Programming Interfaces (APIs). We encourage GSA development groups to use these standards when developing APIs for GSA.

Index

[About These Standards](#)
[Overall Considerations](#)
[Developers Are Your End Users](#)
[Design Considerations](#)
[Future Topics](#)
[Public Domain](#)

About These Standards

These standards are forked from the [18F API Standards](#). They are also influenced by several other sources, including the [White House API Standards](#) and best practices from the private sector.

The standards are a roadmap not a roadblock

These standards are intended to streamline the process for GSA organizations to publish new APIs by providing practical and pragmatic advice. We believe these standards will benefit GSA API development and provide consistency. Projects can benefit if they start considering these in their design even if they do not implement them all.

Lab 7 (extra)

**Brainstorming for new
APIs**