



delphiedintorni.it
 *delphi italian community*



REST API testing with Postman

Part 1 - Introduction



PAOLO ROSSI
WINTech ITALIA **CTO**

SENCHA & EMB





GITHUB PROJECTS



github.com/paolo-rossi



Delphi JWT

JSON Web Token Library for REST



Linux Daemon

Library to build real Linux daemons



Delphi Neon

JSON Serialization Library for REST



OpenAPI-Delphi

OpenAPI 3.0 library for Delphi



WiRL Project

JAX-RS Like REST Library for Delphi

AGENDA

- REST APIs (REST Services)
- The HTTP protocol
- The Postman main application
- Creating simple GET requests
- Creating requests with body
 - POST, PUT, PATCH, etc...
- Inspecting responses (Postman Console)
- Managing Authorization
- Working with Collections
- Capturing HTTP requests (Proxy & Interceptor)



delphiedintorni.it
 *delphi italian community*



REST API testing with Postman

Part 1 - Introduction

TOOLS

- Postman (main GUI app)
 - <https://www.postman.com/>
- HTTPie (client HTTP CLI)
 - <https://httpie.org/>
- Newman (the Postman CLI)
 - <https://github.com/postmanlabs/newman>
- Need a REST server for the demo?
 - <https://github.com/delphi-blocks/WiRL>

REST SERVICES (API)

- HTTP servers
 - Learn the HTTP protocol
 - 83% of web traffic is API traffic
- Using REST architectural style
 - Content negotiation
 - HTTP verbs
 - Resources, not commands
 - Uniform Resource Identification (URI)
 - Resource format (JSON mostly)
 - Auth with tokens (JWT mostly)

HTTP PROTOCOL

- Hypertext Transfer Protocol
 - RFC 2616
- Very simple, text based
- Is what makes the World Wide Web work
- Elements
 - Hyperlink
 - Request, response
 - HTTP Headers
 - URI and query parameters
 - HTTP verbs (GET, POST, PUT, DELETE, etc...)



Demo:

HTTP protocol with HTTPie

- Using browser version
- Downloading & installing npm version
- Public site demo
- Redirection demo

USE CASE SCENARIO

- Test an endpoint
- Test API endpoints
- Building the client against an existing API
- Building the server following a specification
- Building the server from scratch
- Building the server and the client (from scratch)

POSTMAN FEATURES

- Not only RESTful APIs
 - GraphQL APIs
 - SOAP APIs
- Automated Testing
- Design with Mock Servers
- Documentation
- API Monitors
- Workspaces

POSTMAN APP

- Download and install
- Sign up and login
- Postman UI
 - Header
 - Sidebar
 - Footer
 - Request Builder



Demo:

The Postman UI

- Discover Postman UI
 - Header
 - Sidebar
 - Footer
 - Request Builder

REQUEST BUILDER

- Name and description (yes, it's important!)
- HTTP verb
- URL
- Params (description)
- Authorization
- Headers (description)
- Body *
- Pre-request Scripts **
- Tests **
- Send & Save

CREATING REQUESTS

- URL
 - Host + Port + Path + Parameters
 - Resource + Subresource + PathParam
- Verbs
 - GET, POST, PUT, PATCH, DELETE, OPTIONS
- Headers
 - Accept fields
 - Authorization (basic, bearer, etc...)
- Body
 - For POST, PUT, PATCH



Demo:

Building GET requests

- Name and description
- HTTP verb
- URL
- Params (description)
- Authorization
- Headers (description)

“BODY” REQUESTS

- POST (PUT, PATCH, etc...) requests have a body
- Headers
 - Content-Type
 - Content-Encoding
- Body
 - form-data
 - x-www-form-urlencoded
 - Raw
 - Text, JS, JSON, XML
 - Binary
 - GraphQL



Demo:

Building POST requests

- Headers
- Body formats

RESPONSES

- Response Info
 - Status
 - Time
 - Size
- Response Body
 - Pretty
 - Raw
 - Preview
 - Visualize
 - Custom Visualizer
- Headers and Cookies

POSTMAN CONSOLE

- Visualize Raw requests and responses
 - Before they are processed by Postman
- Useful to inspect HTTP traffic at lower level
 - Encoding
 - Redirections
- Compare requests and responses with other tools
- Discover proxy and certificates used
- Error logs from test or pre-request scripts
- `console.log()` from inside scripts



Demo:

Inspecting responses

- Info
- Body
- Headers & Cookies

AUTHORIZATION

- UI to easily set your auth fields
 - Authorization header auto-generated
 - When sending the request
- Inheriting the Authorization settings
 - More later on Collections
- Login with Basic
- Bearer Token



Demo:

Managing Authorization

- Login with Basic
- Bearer Token

COLLECTIONS

- Group requests
 - Keeping your workspace organized
 - Collaboration with teammates
 - Generate API documentation
 - Automate request runs
- Create new collection
- Save requests to collections
- Create request from a collection
- Authorization setting
- Export collections



Demo:

Working with collections

- Create new collection
- Save requests to collections
- Create request from a collection
- Export collections

CAPTURING REQUESTS

- **Interceptor**
 - Install the Chrome extension
 - Install the Interceptor Bridge
 - Connect the Interceptor
- **Proxy**
 - Capture requests from mobile devices

WHAT'S NEXT

- Variables
- Environments
- Mock Servers
- Tests
- Running tests
- Newman

GRAZIE PER
LA PARTECIPAZIONE

DOMANDE?



**REST API testing
with Postman**



PAOLO ROSSI



blog.paolorossi.net



paolo@paolorossi.net



twitter.com/awebguy



github.com/paolo-rossi



linkedin.com/in/paolo-rossi-pc



REST API testing with Postman