

Postman

How to run postman Requests/tests from command line

Pre-requisites

1. Install node.js
2. Install newman
3. export collection and run from command line

* **Newman** is a command line collection runner for Postman

Step 1 : check if node.js is already installed

cmd

node -v

npm -v

Step 2 : Install node.js (*Check in Next slide*)

<https://nodejs.org/en/download/>

Check if node and npm are installed

Step 3 : Install Newman

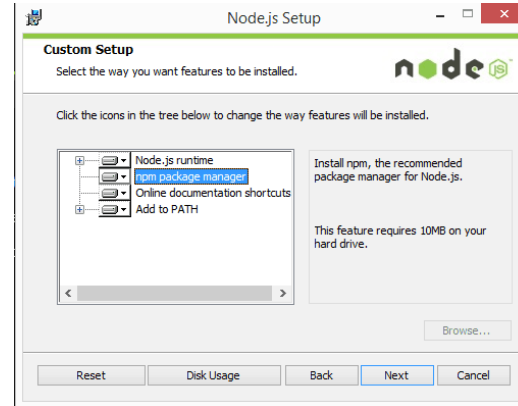
npm install -g newman

Install Node.js and NPM on Windows

1. **Download the Windows installer**
from the <https://nodejs.org/en/download/>
2. **Run the installer** (the .msi file you downloaded in the previous step.)
3. **Follow the prompts in the installer** (Accept the license agreement, click the NEXT button a bunch of times and accept the default installation settings).
4. **Restart your computer.** You won't be able to run Node.js until you restart your computer.
5. **Test:** Check if node.js & npm installed
Open command Prompt execute below commands....

node -v

npm -v



```
Microsoft Windows [Version 10.0.17134.285]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\admin>node -v
v8.11.3

C:\Users\admin>npm -v
5.6.0

Update available 5.6.0 -> 6.4.1
Run npm i npm to update
```

How to run postman Requests/tests from command line

Method-1

Step 1 : Export collection as json file

Step 2 : On cmd goto location of collection json file

Step 3 : Run command

newman run [collectionfilename.json](#)

newman run mycollection.json -n 10 # runs the collection 10 times

Method-2 (with reports)

newman run [collectionfilename.json](#) -r html,xml,junit

Method-3

Step 1 : Share collection → get link

Step 2 : Open Command Prompt

Step 3 : Run command

newman run <https://www.getpostman.com/collections/d3133df708dff40cac81>

Method-4 (with reports)

newman run <https://www.getpostman.com/collections/d3133df708dff40cac81> -r html,xml,junit

How to add & Run SOAP requests in Postman

Wizdler plugin for chrome to get Request Body from WSDL

Example1: Countries Info (SOAP)

- WSDL:
- <http://webservices.oorsprong.org/websamples.countryinfo/CountryInfoService.wso?WSDL>

POST

http://webservices.oorsprong.org/websamples.countryinfo/CountryInfoService.wso?WSDL

WSDL URL

Params

Send

Authorization Headers (1) Body Pre-request Script Tests

☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary XML (text/xml)

```
1 <Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
2   <Body>
3     <CapitalCity xmlns="http://www.oorsprong.org/websamples.countryinfo">
4       <sCountryISOCode>IN</sCountryISOCode>
5     </CapitalCity>
6   </Body>
7 </Envelope>
```

Request

Body Cookies Headers (8) Test Results

Pretty Raw Preview XML

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
3   <soap:Body>
4     <m:CapitalCityResponse xmlns:m="http://www.oorsprong.org/websamples.countryinfo">
5       <m:CapitalCityResult>New Delhi</m:CapitalCityResult>
6     </m:CapitalCityResponse>
7   </soap:Body>
8 </soap:Envelope>
```

Response

Countries info
Example

Status: 200 OK

Example2: Number Conversion Service (SOAP)

- The Number Conversion Web Service, implemented with DataFlex, provides functions that convert numbers into words or dollar amounts.
- The following operations are available. For a formal definition, please review the Service Description.
- **NumberToWords**
 - Returns the word corresponding to the positive number passed as parameter. Limited to quadrillions.
- **NumberToDollars**
 - Returns the non-zero dollar amount of the passed number.
- WSDL: <https://www.dataaccess.com/webservicesserver/NumberConversion.wso?WSDL>

POST NumberConversion No Environment

NumberConversion

POST https://www.dataaccess.com/webservicesserver/numberconversion.wso?WSDL Send

Params Authorization Headers (1) Body Pre-request Script Tests

form-data x-www-form-urlencoded raw binary XML (text/xml)

```
1 <Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
2   <Body>
3     <NumberToDollars xmlns="http://www.dataaccess.com/webservicesserver/">
4       <dNum>10.45</dNum>
5     </NumberToDollars>
6   </Body>
7 </Envelope>
```

Request

Status: 200 OK Time: 263 ms Size: 806 B Save

Body Cookies Headers (13) Test Results

Pretty Raw Preview XML

Number Conversion Example

Response

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
3   <soap:Body>
4     <m:NumberToDollarsResponse xmlns:m="http://www.dataaccess.com/webservicesserver/">
5       <m:NumberToDollarsResult>ten dollars and forty five cents</m:NumberToDollarsResult>
6     </m:NumberToDollarsResponse>
7   </soap:Body>
8 </soap:Envelope>
```

Example3: Text Casing Service (SOAP)

- The Text Casing Web Service, implemented with DataFlex, provides functions to change text casing in different ways.
- The following operations are available. For a formal definition, please review the Service Description.
- **InvertStringCase**
 - Uppercase letters that are lowercase and lowercase letters that are uppercase.
- **InvertCaseFirstAdjustStringToPrevious**
 - Invert the casing of the first letter and adjust the string to conform with the previous casing of first letter.
- **InvertCaseFirstAdjustStringToCurrent**
 - Invert the casing of the first letter and adjust the string to conform with the current casing of first letter.
- **AllUppercaseWithToken**
 - Change string to uppercase adding token between characters.
- **AllLowercaseWithToken**
 - Change string to lowercase adding token between characters.
- **UppercaseWordsWithToken**
 - Change string to uppercase adding token between characters of each word.
- **LowercaseWordsWithToken**
 - Change string to lowercase adding token between characters of each word.
- WSDL: <https://www.dataaccess.com/webservicesserver/TextCasing.wso?WSDL>

POST ▼ https://www.dataaccess.com/webservicesserver/TextCasing.wso?WSDL Send ▼

Params ● Authorization Headers (1) Body Pre-request Script Tests

form-data x-www-form-urlencoded raw binary XML (text/xml) ▼

```
1 <Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
2   <Body>
3     <InvertStringCase xmlns="http://www.dataaccess.com/webservicesserver/">
4       <sAString>WelCome</sAString>
5     </InvertStringCase>
6   </Body>
7 </Envelope>
```

Request

Text Case Conversion Example

Body Cookies Headers (13) Test Results Status: 200 OK Time: 268 ms Size: 789 B Save

Pretty Raw Preview XML ≡

Response

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
3   <soap:Body>
4     <m:InvertStringCaseResponse xmlns:m="http://www.dataaccess.com/webservicesserver/">
5       <m:InvertStringCaseResult>wELcOME</m:InvertStringCaseResult>
6     </m:InvertStringCaseResponse>
7   </soap:Body>
8 </soap:Envelope>
```