

# Chaining Multiple Requests

This video tutorial demonstrates, how you may chain multiple requests in vREST:

To chain multiple requests (test cases), Variables can be extracted from Test Case's responses and can be used in subsequent HTTP requests (test cases). To extract the variable's values from test case's actual results, please follow the steps below:

- Expand the test case from which you want to extract variables
- Go to "Variable Extractor" sub tab
- In the "JSON Path / XML Path / Utility Method" column, you may provide
  - the JSON path of the API response if the response is in JSON format. For more information, please read [JSON Path Syntax](#).
  - the XML path of the API response if the response is in XML format. For more information, please read [XML Path Syntax](#).
  - utility method to extract data from any type of response. For more information, please read [Extracting variable via utility methods](#).

Let us take an example, how we can extract a variable from API response:

- Suppose we have an API which creates resource on server and returns the following JSON response:

```
{
  "id": "54a79b704cba8d5328d087f5",
  "resource_name": "testcase",
  "resource_url": "http://vrest.io/i/demo/m/RVD/create_resource",
  "resource_description": "This API creates a resource on the server",
  "meta": {
    "created_at": "2015-01-03T07:41:21.000Z"
  }
}
```

- Now, we want to save the id of newly created resource into variable say "resourceId" and creation time into variable say "resourceCreatedAt". We can extract these variables in the following way:

6. Validate whether create resource API works on providing valid input

PUT https://vrest.io/i/demo/m/RVD/create\_resource

Details Request Response Validation Results Variable Extractor

New Delete Help

Name	Path
resourceId	id
resourceCreatedAt	meta.created_at

Note:

- Variable extractor is only applicable for JSON responses.
- For defining JSON path, follow the [guide](#).
- From vREST v2.0.0, we have removed the JSON path prefix "\$." or "\$".
- These variables will be available in all subsequent test cases within a test run.

Few points regarding writing **Path** in the above table:

- Each individual property value can be extracted via JSON path notation e.g. `id` or `meta.created\_id`
- For more information, read [JSON Path syntax](#).
- Now we can use these extracted variables in subsequent requests. Note that once a variable is defined, it can be used in all subsequent requests within that test run only. If you want to override this variable, simply re-define the variable in any request.

Suppose we have an API which updates this newly created resource and it needs the ID of the resource to update. We can use the `{{resourceId}}` variable (extracted in previous step) in the URL as shown in the following figure:

7. Validate whether update resource API works

**PATCH** https://vrest.io/demo/m/RVD/update\_resource1/{{resource\_id}}

Details | Request | Response Validation | Results | Variable Extractor

URL	https://vrest.io/demo/m/RVD/update_resource1/{{resource_id}}		
Method	PATCH		
Description	Empty		
Project	Demo Project	Test Suites	Empty
Versions	Empty	Authorization	Empty
Tags	Empty	Wait for seconds	Empty



#### Notes

- Variable extraction feature is only applicable for JSON / XML responses at present using JSON / XML path.
- For defining JSON path, follow the [guide](#).
- For defining XML path, follow the [guide](#).
- If you want to extract variables from responses other than JSON/XML, then you need to write your own utility method to extract data from either response body or response headers. For more information on this, please read this [doc \(Extracting variable via Utility Methods\)](#).