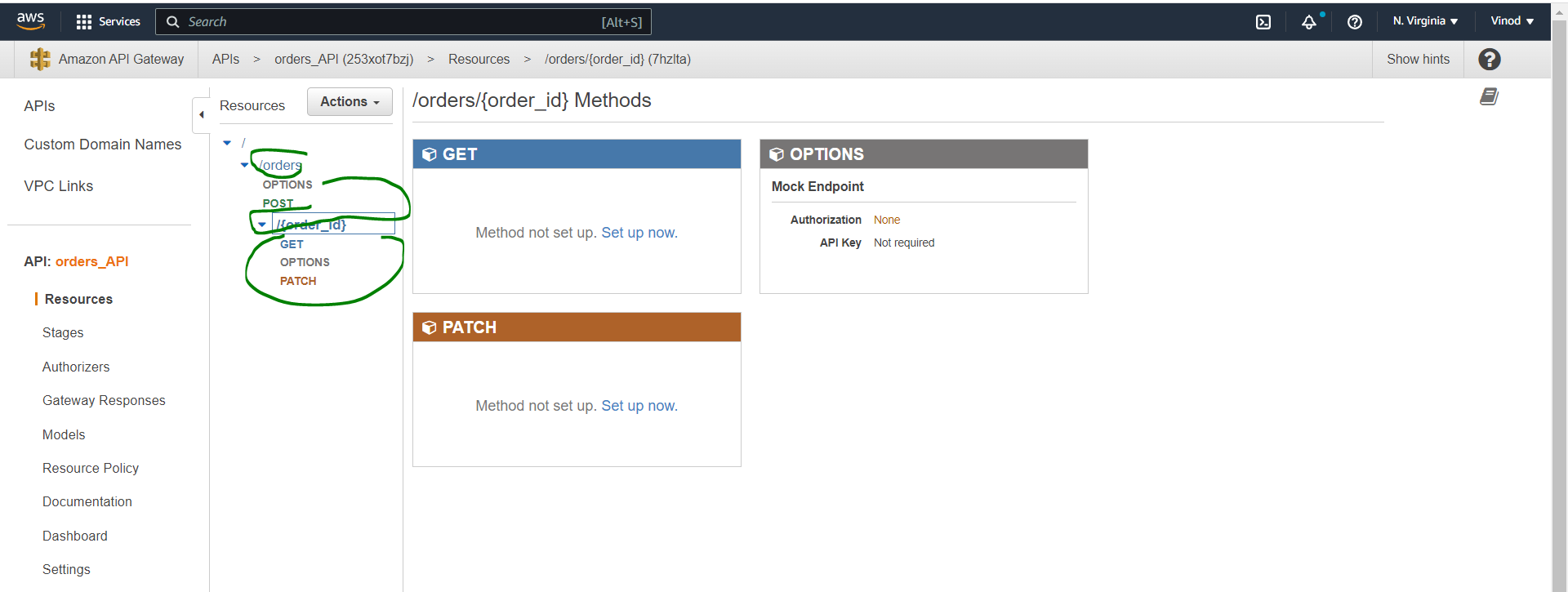
Under orders, create a resource with name order\_id and a variable path {order\_id}.

The GET method was created under this new resource. The purpose of this method is to retrieve the order details of a particular order from the ‘orders’ dynamodb table.

The PATCH method was also created under this resource. The purpose of this method is to update the details of a particular order. More on this in the next document.



We will not be using Lambda proxy integration meaning we need to use the Integration Request and Integration Response modules to modify the data going into and coming out of the lambda function invoked by this method.

In the *Method Request* section, apart from setting the API Key Required to true, there is nothing else to be done. There is no query string parameter, header or body for us in this request to be validated. We only need the order\_id and that is a resource i.e. part of the main URL itself.

In the *Integration Request* section, we are not using lambda proxy integration. We use the Mapping Templates section to extract the order\_id that is part of the request URL. Without this mapping template, when you print out the ‘event’ in lambda, you will get a blank event.

#set($inputRoot = $input.path('$'))

{"order\_id": "$input.params('order\_id')"}

In the *Integration Response* section, we use the Mapping Templates section to extract the order\_details in the format we want.

#set($inputRoot = $input.path('$'))

$input.json('$.body')

Also because we have enabled CORS, under ‘Header Mappings’ section, we will add ’\*’ for Access-Control-Allow-Origin.

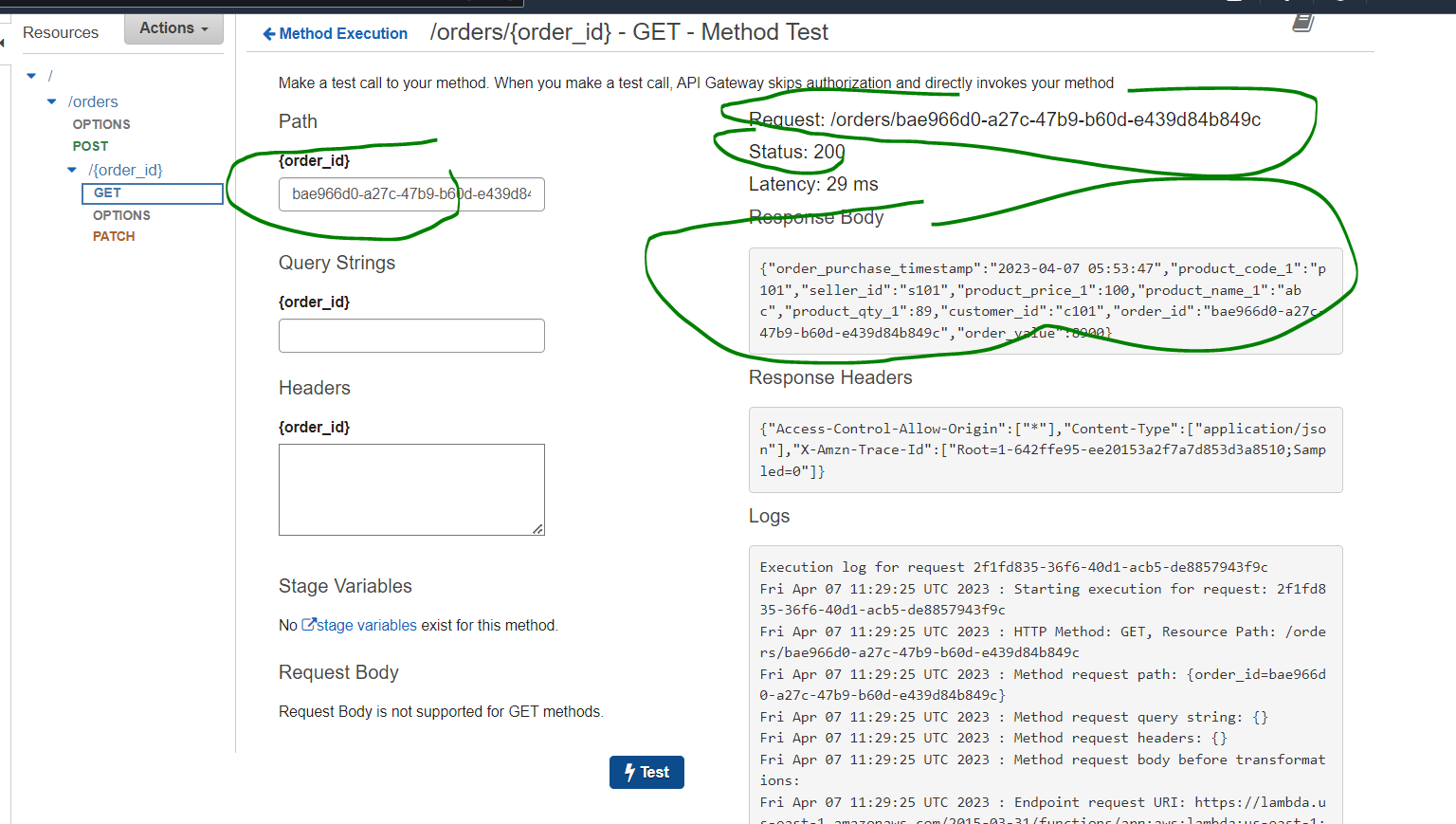
In the *Method Response* section, nothing to be done. The header Access-Control-Allow-Origin would have been added when we enabled CORS and the value for this was already set as ‘\*’ in the Integration Response section.

*Lambda function invoked by the GET method:*

The lambda function invoked by this method will retrieve the order details from the orders dynamodb table. Refer the “Code” folder for the get\_item code.

**TESTING:**

*From within API Gateway:*



*From POSTMAN:*

