

REPORT AWS - NODE JS - DYNAMODB

For the project was used [NodeJs Serverless Stack](#). In site serverless view the installation instructions for serverless

Previous Steps

- Clone this repo
- Have a aws account(free) or create
- Install or have an aws cli view site for installation instructions:

<https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

- Install or have an node js environment view site for installation instructions:

<https://www.journaldev.com/7402/node-js-environment-setup-node-js-installation>

View Documentation

View the file `manual.pdf` located in documentation folder

Import Postman collection

Import the file `report.postman_collection.json` located in documentation folder in postman

Commands for installation

aws configure

Configure your aws credentials

npm install

Execute the command in the root folder of the project for install the node packages located in `package.json` file

sls deploy

Deploy serverless stack to AWS cloud.

npm test

Execute jests tests

```

callback
  ✓ get report (90ms)

pdf_report tests
  ✓ pdf_report is object
  ✓ pdf download mock

14 passing (357ms)

```

Configure postman endpoint

```

Service Information
service: serverless-aws-nodejs
stage: dev
region: us-east-1
stack: serverless-aws-nodejs-dev
resources: 48
api keys:
  None
endpoints:
  POST - https://ppoziu3l24.execute-api.us-east-1.amazonaws.com/dev/report
  PUT - https://ppoziu3l24.execute-api.us-east-1.amazonaws.com/dev/report/{id}
  GET - https://ppoziu3l24.execute-api.us-east-1.amazonaws.com/dev/report/{id}
  GET - https://ppoziu3l24.execute-api.us-east-1.amazonaws.com/dev/reports
  GET - https://ppoziu3l24.execute-api.us-east-1.amazonaws.com/dev/csv
  DELETE - https://ppoziu3l24.execute-api.us-east-1.amazonaws.com/dev/report/{id}
  GET - https://sts3dnagf9.execute-api.us-east-1.amazonaws.com/api/pdf_report
functions:
  create: serverless-aws-nodejs-dev-create
  edit: serverless-aws-nodejs-dev-edit
  get: serverless-aws-nodejs-dev-get
  all: serverless-aws-nodejs-dev-all
  csv: serverless-aws-nodejs-dev-csv
  pdf_report: serverless-aws-nodejs-dev-pdf_report
  delete: serverless-aws-nodejs-dev-delete
layers:
  None

Monitor APIs by route with the Serverless Dashboard: run "serverless"
mac@MacBook-Pro-de-MAC:~/serverless-dynamodb-nodejs-starter% 

```

- take the url part of the endpoint similar to the following: <https://xxxxx.execute-api.us-east-1.amazonaws.com>
- copy the endpoint string before /dev/report then set the variable awsReportEndPoint in postman
- copy the endpoint string before /pdf_report then set the variable awsPdfReportEndPoint in postman

Services endpoints

Consume in postman any of the following services:

- Create Report
- Update Report
- Delete Report by Id
- Get Report by Id
- Get All Reports

- Generate CSV Report
- Generate PDF Report

Run integration tests

Press right click on the collection in postman, in the menu that is displayed select run collection:

The screenshot shows the Postman interface with the 'report' collection selected in the left sidebar. A context menu is open over the collection name 'report'. The 'Run collection' option is highlighted with a red box. Other options in the menu include 'Edit', 'Share collection', 'Add request', 'Add folder', 'Monitor collection', 'Create a fork', 'Create Pull Request', 'Merge changes', 'View documentation', 'Rename', 'Duplicate', 'Export', 'Manage roles', 'Remove from workspace', and 'Delete'.

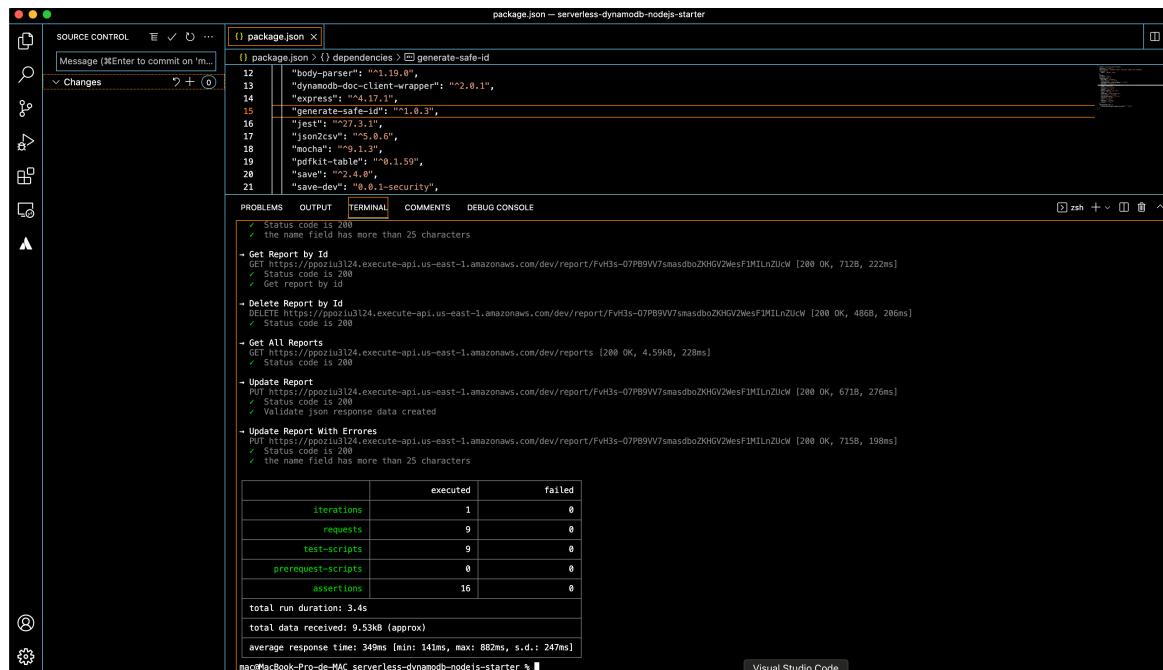
The screenshot shows the Postman interface with the 'Runner' tab selected. The 'report' collection is selected in the left sidebar. In the main area, a 'RUN ORDER' table lists various API requests. The 'Run report' button at the bottom right of the table is highlighted with a red box. The table includes columns for method, URL, and description. The 'Iterations' field is set to 1. Other settings include 'Delay' (0 ms), 'Data' (Select File), and several checkboxes for response handling and cookie management.

<input checked="" type="checkbox"/>	GET	Generate PDF Report
<input checked="" type="checkbox"/>	GET	Generate CSV Report
<input checked="" type="checkbox"/>	POST	Create Report
<input checked="" type="checkbox"/>	GET	Get Report by Id
<input checked="" type="checkbox"/>	DEL	Delete Report by Id
<input checked="" type="checkbox"/>	GET	Get All Reports
<input checked="" type="checkbox"/>	PUT	Update Report

Run integration test in console with newman

Run the next command replace the vars `awsPdfReportEndPoint` and `awsReportEndPoint` with the endpoint url

```
newman run documentation/report.postman_collection.json --global-var
"awsPdfReportEndPoint=https://xxxx.execute-api.us-east-1.amazonaws.com" --
-global-var "awsReportEndPoint=https://xxxx.execute-api.us-east-
1.amazonaws.com"
```



The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the results of a Newman run against a Postman collection. The output includes various API requests and their responses, along with a summary table at the bottom.

	executed	failed
iterations	1	0
requests	9	0
test-scripts	9	0
prerequest-scripts	0	0
assertions	16	0

total run duration: 3.4s
total data received: 9.53kB (approx)
average response time: 349ms [min: 141ms, max: 882ms, s.d.: 247ms]

Some Examples of api responses:

The screenshot shows the Postman application interface with two separate API requests displayed.

Request 1: GET /dev/reports

- Method:** GET
- URL:** {{awsReportEndPoint}}/dev/reports
- Params:** Authorization, Headers (6), Body, Pre-request Script, Tests, Settings
- Query Params:** KEY, VALUE, DESCRIPTION, Bulk Edit
- Body:** Pretty, Raw, Preview, Visualize, JSON
- Response Status:** 200 OK, Time: 2.14 s, Size: 3.58 KB, Save Response

```

1 [
2   {
3     "alias": "Alias",
4     "species": "species",
5     "company": [
6       {
7         "name": "company name",
8         "team": "company team"
9       },
10      {
11        "dateCreation": 1636485873228,
12        "id": "NtRE1QDpALAshf-7EielLnx29XTbnLM4QLVYtxh",
13        "name": "Name"
14      },
15      {
16        "alias": "Alias",
17        "species": "species",
18        "company": [
19          {
20            "name": "company name",
21            "team": "company team"
22          },
23          {
24            "dateCreation": 1636485878531,
25            ...
26          }
27        ],
28        ...
29      }
30    ]
  
```

Request 2: GET /api/pdf_report

- Method:** GET
- URL:** {{awsPdfReportEndPoint}}/api/pdf_report
- Params:** Authorization, Headers (6), Body, Pre-request Script, Tests, Settings
- Query Params:** KEY, VALUE, DESCRIPTION, Bulk Edit
- Body:** Body, Cookies, Headers (5), Test Results
- Response Status:** 200 OK, Time: 267 ms, Size: 2.21 KB, Save Response

Name	Alias	Species	Company Name	Company Team
Cesar Vargas	cesarv	species	The factory of users	Dandeele
Name example	Alias example	species example	company name example	company team example