

# 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

<b>GET</b> Generate PDF Report {{awsPdfReportEndPoint}}/api/pdf_report / Generate PDF Report	200 OK 319 ms 2.382 KB
Pass Status code is 200	
Pass Content-Type header is application/json	
<b>GET</b> Generate CSV Report {{awsReportEndPoint}}/dev/csv / Generate CSV Report	200 OK 535 ms 1.036 KB
Pass Status code is 200	
Pass Content-Type header is text/csv	
<b>POST</b> Create Report {{awsReportEndPoint}}/dev/report / Create Report	200 OK 290 ms 713 B
Pass Status code is 200	
Pass Validate json response data created	
<b>GET</b> Get Report by Id {{awsReportEndPoint}}/dev/report/{{id}} / Get Report by Id	200 OK 338 ms 713 B
Pass Status code is 200	
Pass Get report by id	
<b>DELETE</b> Delete Report by Id {{awsReportEndPoint}}/dev/report/{{id}} / Delete Report by Id	200 OK 372 ms 487 B
Pass Status code is 200	
<b>GET</b> Get All Reports {{awsReportEndPoint}}/dev/reports / Get All Reports	200 OK 183 ms 1.779 KB
Pass Status code is 200	
<b>PUT</b> Update Report {{awsReportEndPoint}}/dev/report/{{id}} / Update Report	200 OK 272 ms 672 B
Pass Status code is 200	
Pass Validate json response data created	

## Some Examples of api responses:

The screenshot shows the Postman interface with the following details:

- Left Sidebar:** My Workspace, Collections, APIs, Environments, Mock Servers, Monitors, History.
- Top Bar:** New, Import, GET Generate PDF Rep..., GET Generate CSV Re..., POST Create Report, GET Get All Reports, report, localization.
- Current Collection:** report / Get All Reports
- Selected Endpoint:** GET {{awsReportEndPoint}}/dev/reports
- Params Tab:** Params, Authorization, Headers (6), Body, Pre-request Script, Tests, Settings.
- Body Tab:** PRETTY, Raw, Preview, Visualize, JSON. The JSON response is:

```

1  {
2   "alias": "Alias",
3   "species": "species",
4   "company": {
5     "name": "company name",
6     "team": "company team"
7   },
8   "dateCreation": 1636485873228,
9   "id": "NRTEOLbp4LAshf-7EieL1Nv29XTbnLM4QLVytXh",
10  "name": "Name"
11 },
12 [
13   {
14     "alias": "Alias",
15     "species": "species",
16     "company": {
17       "name": "company name",
18       "team": "company team"
19     },
20     "dateCreation": 1636485870531
21 ]

```
- Status Bar:** Status: 200 OK Time: 2.14 s Size: 3.58 KB Save Response.

Screenshot of the Postman API testing tool interface showing a collection named "report".

The "Generate PDF Report" endpoint is selected, displaying the following details:

- Method:** GET
- URL:** {{awsPdfReportEndPoint}}/api/pdf\_report
- Params:** Authorization, Headers (6), Body, Pre-request Script, Tests, Settings
- Query Params:** KEY: Key, VALUE: Value, DESCRIPTION: Description, Bulk Edit

The response body shows a table titled "Report" with the following data:

Name	Alias	Species	Company Name	Company Team
Cesar Vegas	cesarvegas	special	The factory of users	Dandevils
Name example	Alias example	species example	company name example	company team example

Test results indicate a successful 200 OK response.