

REPORT AWS - NODE JS - DYNAMODB

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

Previous Steps

- 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, with the 'Run collection' option highlighted by a red box.

The screenshot shows the Postman interface with the 'Runner' tab selected. The 'report' collection is selected in the left sidebar. In the main area, the 'RUN ORDER' section lists various API requests. The 'Run report' button at the bottom right of the runner panel is highlighted by a red box.

GET 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	
GET 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	
POST Create Report {{awsReportEndPoint}}/dev/report / Create Report	200 OK 290 ms 713 B
Pass Status code is 200	
Pass Validate json response data created	
GET 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	
DELETE Delete Report by Id {{awsReportEndPoint}}/dev/report/{{id}} / Delete Report by Id	200 OK 372 ms 487 B
Pass Status code is 200	
GET Get All Reports {{awsReportEndPoint}}/dev/reports / Get All Reports	200 OK 183 ms 1.779 KB
Pass Status code is 200	
PUT 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 application interface. On the left, there's a sidebar with icons for Collections, APIs, Environments, Mock Servers, and Monitors. The main area displays a collection named "report". Inside the "report" collection, there are several items:

- "aws report" contains endpoints: POST Create User, PUT Update User, DEL Delete User, GET Get Users, GET Generate report users CSV, GET Generate report users PDF, and GET Get User.
- "report" contains endpoints: GET Generate PDF Report, GET Generate CSV Report, POST Create Report, GET Get Report by Id, DEL Delete Report by Id, and GET Get All Reports.
- "report" also contains a "report" item which has its own GET endpoint: {{awsReportEndPoint}}/dev/reports.

 The "report" item under "report" is currently selected. Its details pane shows a GET request with the URL {{awsReportEndPoint}}/dev/reports. The "Params" tab is active, showing a single parameter "key" with a value "Value". The "Body" tab shows a JSON response with two objects. The first object has keys "alias", "species", "company", and "dateCreation". The second object has keys "alias", "species", "company", and "dateCreation". Both objects have nested properties like "name" and "team". The response body is as follows:


```

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 ]
    
```

 The status bar at the bottom indicates a 200 OK response with a time of 2.14 s and a size of 3.58 KB.

Screenshot of the Postman API testing tool interface showing a successful response for a PDF report generation endpoint.

The left sidebar shows "My Workspace" with collections like "report" containing various API endpoints such as "Generate PDF Report", "Generate CSV Report", "Create Report", etc.

The main window displays a "Generate PDF Report" request:

- Method: GET
- URL: {{awsPdfReportEndPoint}}/api/pdf_report
- Headers (6):
 - Content-Type: application/json
 - Authorization: Bearer {{token}}
 - Content-Type: application/json
 - Content-Type: application/json
 - Content-Type: application/json
 - Content-Type: application/json
- Query Params:

KEY	VALUE	DESCRIPTION
Key	Value	Description
- Body: (Empty)

The response details are as follows:

- Status: 200 OK
- Time: 267 ms
- Size: 2.21 KB
- Save Response

The response body contains a table titled "Report":

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