Cloud Project Report

SRIJA BANDA

Task 1: Word Count on AWS EC2/LightSail using PySpark

Objective

To deploy PySpark on an AWS EC2 or LightSail instance and run a simple word count script that reads a file from Amazon S3 and writes the output back to S3.

Step-by-Step Execution

1. Launch EC2/LightSail Instance

- Chose Amazon Linux 2
- Connected via SSH using AWS Console

2. Install Java, Python, and PySpark

```
sudo yum install java-11 -y

export JAVA_HOME=$(dirname $(dirname $(readlink -f $(which java))))

java --version


sudo yum install python3-pip -y

pip3 install pyspark
```

3. Set Up S3 Access in PySpark Script

- Created an S3 bucket with input_file.txt
- Wrote word_count.py script using PySpark
- Included AWS credentials and input/output S3 paths in the script

4. Execute the PySpark Job

```
spark-submit word_count.py
```

- Output saved to S3 bucket under specified output folder

5. Validate Spark UI

- Accessed Spark UI on port 4040 (if necessary) to validate job execution (optional)

6. Screenshot Evidence

- SSH terminal with PySpark job output
- S3 bucket showing input and output files

Task 2: Dockerized Node.js App Deployment on AWS LightSail

Objective

To build and deploy a simple Node.js web application using Docker on AWS LightSail, and publish the image to Docker Hub.

Step-by-Step Execution

1. Create Node.js App

mkdir docker-node-app && cd docker-node-app

npm init -y

npm install express

Create server.js:

```
const express = require('express');

const app = express();

const port = 3000;


app.get('/', (req, res) => {

  res.send('Hello from Dockerized Node.js App on LightSail!');

});


app.listen(port, () => {

  console.log(`Server running on http://localhost:${port}`);

});
```

2. Create Dockerfile

```
FROM node:16
```

```
WORKDIR /app

COPY package*.json ./

RUN npm install

COPY . .

EXPOSE 3000

CMD ["node", "server.js"]
```

3. Install Docker on LightSail

```
sudo yum update -y

sudo yum install docker -y

sudo service docker start

sudo usermod -aG docker ec2-user

# Logout and reconnect
```

4. Build and Run Docker Container

```
docker build -t my-node-app .

docker run -d -p 80:3000 my-node-app
```

5. Open Port 80 in LightSail Networking

- Navigated to instance's Networking tab
- Added firewall rule:
    - Port: 80
    - Protocol: TCP
    - Source: All

6. View in Browser

Visited:

http://<lightSail-public-ip>/

- App returned: Hello from Dockerized Node.js App on LightSail!

7. Push Image to Docker Hub

```
docker login

docker tag my-node-app srijabanda/my-node-app:latest
```

docker push srijabanda/my-node-app:latest

- Docker Repo: https://hub.docker.com/r/srijabanda/my-node-app

8. Screenshot Evidence

- Docker build and run output
- Live app in browser
- LightSail firewall rule
- Docker Hub repo

Conclusion

Both tasks demonstrate end-to-end deployment of a data processing job (Task 1) and a containerized web application (Task 2) using AWS infrastructure, PySpark, Docker, and Node.js. All steps were verified with live outputs and linked resources.