For this taSK We are using API of Open Weather Map's public API and created our own microservice

Using the Fetch API to fetch Open Weather Map's public API for JSON data for US cities

Command line weather app using Node.js and [OpenWeatherMap](http://openweathermap.org/).

This library provides an interface to query the current weather and temperature using a zipcode location and the OpenWeatherMap [API](http://openweathermap.org/current).

* Create an app, package it in a container and publish to a Docker registry:-
* Creation of Dockerfile

|  |
| --- |
| FROM node:7 |
|  |
|  | RUN apt-get update && apt install -y git && \ |
|  | git clone <https://github.com/thedevopsguru/caas_task_microservice> /home/ec2-user/test\_repo |
|  | WORKDIR /app |
|  | COPY package.json app.js config.json printer.js weather.js ./ |
|  | RUN npm install |
|  | COPY . /app |
|  | EXPOSE 8081 |
|  | RUN node app.js & |

**Docker Build Image and run**

docker build -t microservice\_weather .



docker run -it microservice\_weather





**Run app in locolhost using this command:-**

node app.js F 90210 20001 10001



**Publish your image to a container registry AWS**

aws ecr create-repository --repository-name microservice\_weather

aws ecr get-login

**Tag Image of container**

docker tag final\_weather:1.0 641825664234.dkr.ecr.ap-south-1.amazonaws.com/final\_weather:latest

**Push container**

docker push 641825664234.dkr.ecr.ap-south-1.amazonaws.com/final\_weather:latest

**Deploying the containers**

In order to deploy the containers I have created a small yml file for the microservice\_weather service and the Container to be deployed.

####### deployment.yml###########

apiVersion: v1

kind: Pod for docker image deployment to Kuberenetes

metadata:

name: final\_weather

spec:

replicas: 1

template:

metadata:

labels:

app: final\_weather

spec:

containers:

- name: final\_weather

image: 641825664234.dkr.ecr.ap-south-1.amazonaws.com/final\_weather:latest

ports:

- containerPort: 8081

imagePullPolicy: Always

kubectl create -f /tmp/ deployment.yml

**Run your container on the cluster**



kubectl run microservice\_weather --image=641825664234.dkr.ecr.ap-south-1.amazonaws.com/final\_weather:latest --port=8081



kubectl get deployment microservice\_weather





Kubernates updates the pod and pulls the container with tag final\_weather:1.0 from ECR.



curl [http://13.232.82.65:8081](http://13.232.82.65:8081/)



Kubernates Installation of worker and master nodes:-



**On Master node:**

sudo yum install -y kubelet kubeadm kubectl

Enable & start kubelet service:

sudo systemctl enable kubelet && sudo systemctl start kubelet

**On worker node:**

sudo yum install -y kubelet kubeadm kubectl

