**Practical No: 7**

**Date: 17/04/2023**

**Aim: Running Location Service**

**Description:**

**Give Overview of Location Service**

The Location Service is a Docker-based microservice that provides geolocation capabilities to an application. It can be used to store and retrieve location data for different members of a team. The service is built on top of the Flask framework and uses a PostgreSQL database for data storage. The Location Service provides RESTful APIs that can be accessed using HTTP requests. These APIs can be used to create, read, update, and delete location data. The service can be easily deployed as a Docker container, making it easy to integrate into any application stack that uses containerization. Overall, the Location Service is a powerful tool for any application that requires location-based functionality.

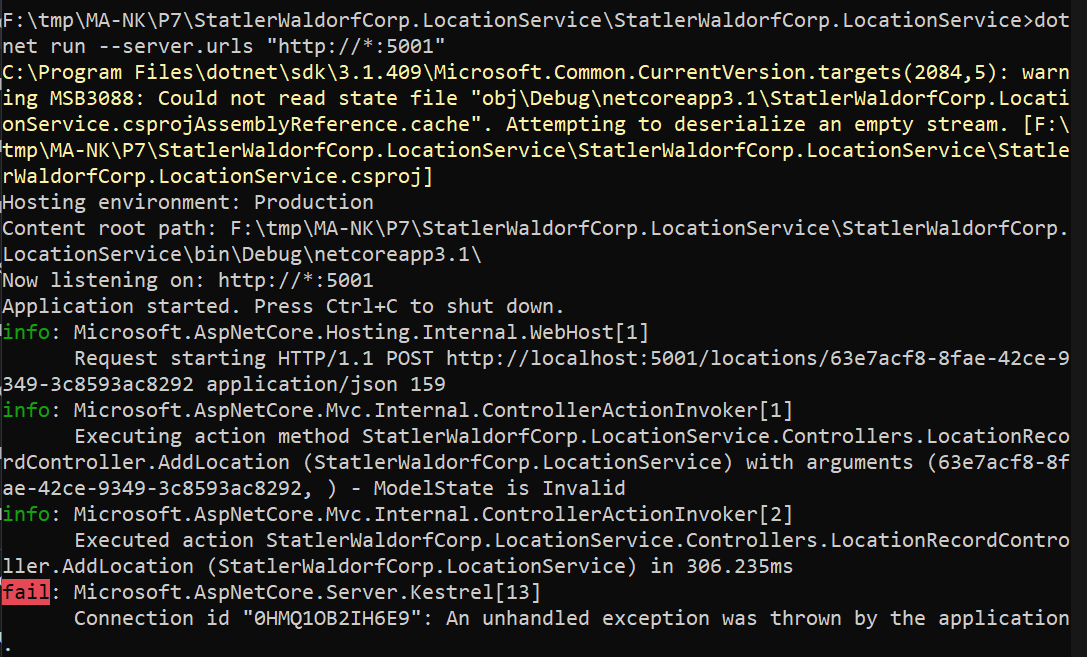
**7 a) Running Location Service Locally using .Net core**

**Code & Output:**

**We need to open 3 command prompts**

**On command prompt 1 start location service (go inside locationservices folder first)**

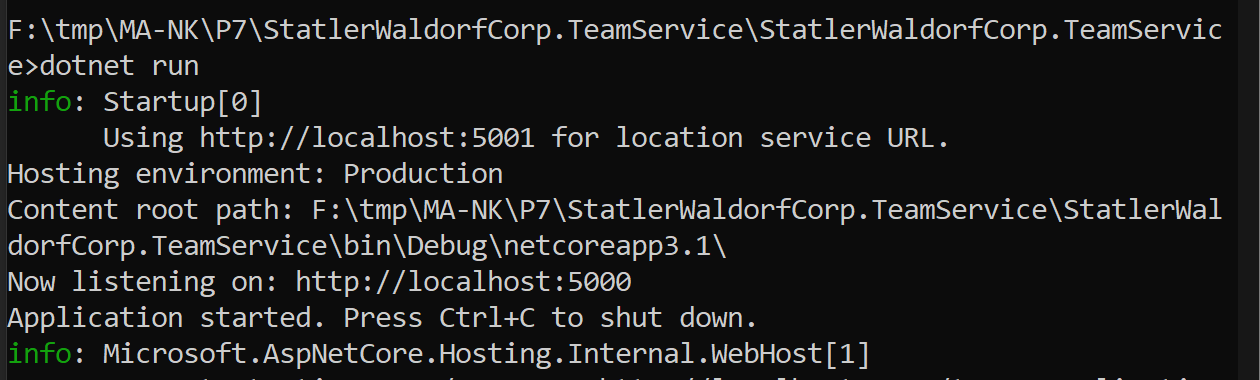
**Command :**

**dotnet run --server.urls "http://\*:5001"**

**On command prompt 2 start location service (go inside teamservices folder first)**

**Command :**

**dotnet run**

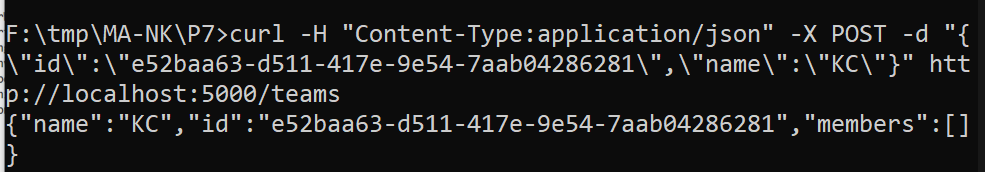
****

**On command prompt 3 run all following commands**

**Command to Add new team**

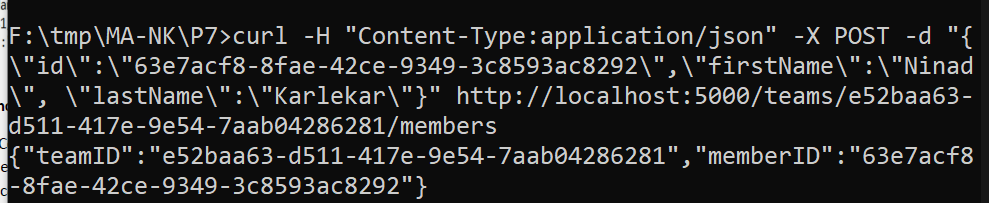
**Commands:**

**curl -H "Content-Type:application/json" -X POST -d "{\"id\":\"e52baa63-d511-417e-9e54-7aab04286281\", \"name\":\"KC\"}"** [**http://localhost:5000/teams**](http://localhost:5000/teams)

****

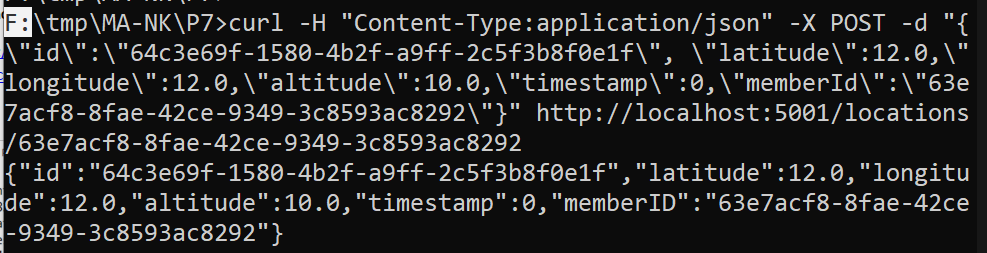
**Command to add new member to team**

**curl -H "Content-Type:application/json" -X POST -d "{\"id\":\"63e7acf8-8fae-42ce-9349-3c8593ac8292\",\"firstName\":\"Ninad\", \"lastName\":\"Karlekar\"}"** [**http://localhost:5000/teams/e52baa63-d511-417e-9e54-7aab04286281/members**](http://localhost:5000/teams/e52baa63-d511-417e-9e54-7aab04286281/members)

****

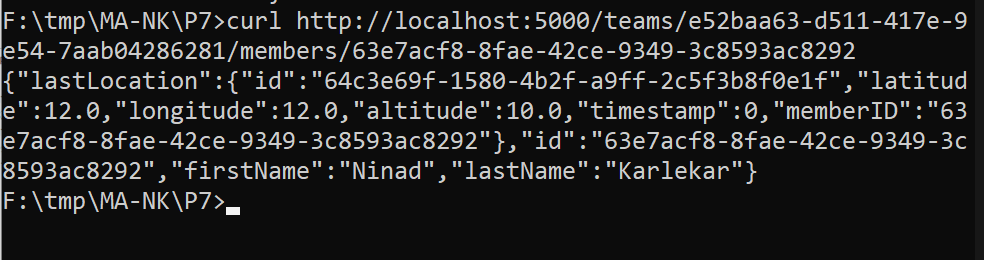
**Command to add new location to member**

**curl -H "Content-Type:application/json" -X POST -d "{\"id\":\"64c3e69f-1580-4b2f-a9ff-2c5f3b8f0e1f\",\"latitude\":12.0,\"longitude\":12.0,\"altitude\":10.0,\"timestamp\":0,\"memberId\":\"63e7acf8-8fae-42ce-9349-3c8593ac8292\"}"** [**http://localhost:5001/locations/63e7acf8-8fae-42ce-9349-3c8593ac8292**](http://localhost:5001/locations/63e7acf8-8fae-42ce-9349-3c8593ac8292)

****

**Command To confirm it is accessible from teams (that is from port 5000) it shows last location**

**curl** [**http://localhost:5000/teams/e52baa63-d511-417e-9e54-7aab04286281/members/63e7acf8-8fae-42ce-9349-3c8593ac8292**](http://localhost:5000/teams/e52baa63-d511-417e-9e54-7aab04286281/members/63e7acf8-8fae-42ce-9349-3c8593ac8292)

****

**7 b) Running Location Service in Docker**

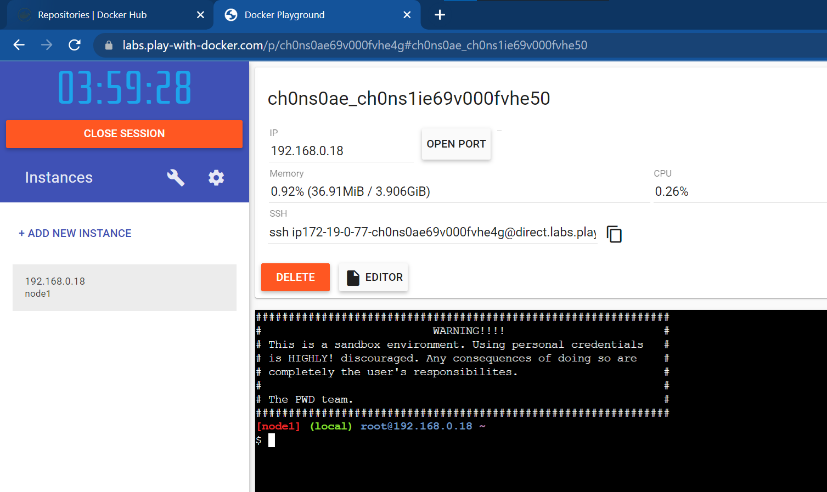
**Code & Output:**

**(create docker hub login first to use it in play with docker)**

**Now login in to Play-With-Docker**

**Click on Start**

**Click on Add New Instance**

****

**Start typing following commands**

Command : To run teamservice

docker run -d -p 5000:5000 -e PORT=5000 \

-e LOCATION\_\_URL=http://localhost:5001 \

dotnetcoreservices/teamservice:location

output: (you can observe that it has started port 5000 on top)

Text

Description automatically generated

Command: to run location service

docker run -d -p 5001:5001 -e PORT=5001 \

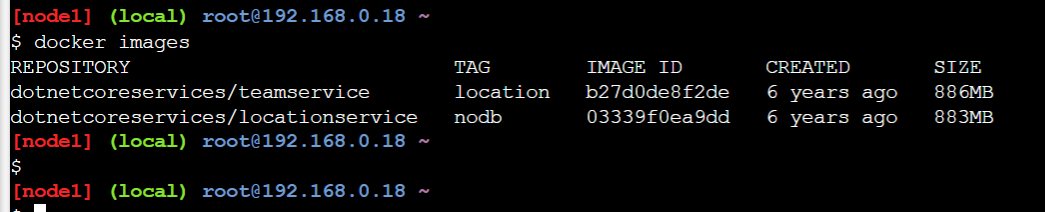
dotnetcoreservices/locationservice:nodb

output: (now it has started one more port that is 5001 for location service)



Command : to check running images in docker

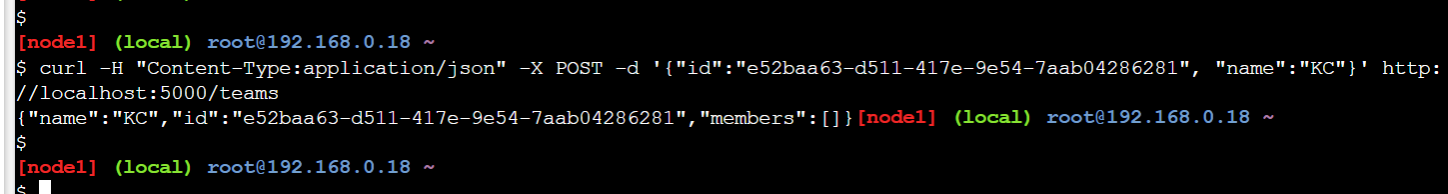
docker images



Command: to create new team

curl -H "Content-Type:application/json" -X POST -d \

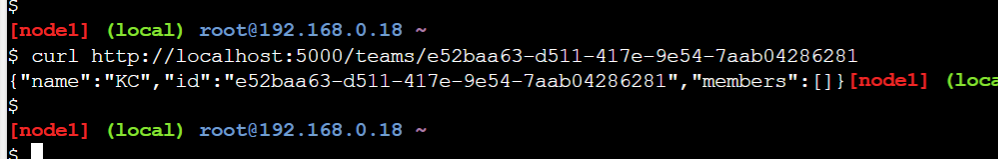
'{"id":"e52baa63-d511-417e-9e54-7aab04286281", "name":"KC"}' <http://localhost:5000/teams>



Command :To confirm that team is added

curl http://localhost:5000/teams/e52baa63-d511-417e-9e54-7aab04286281

Output

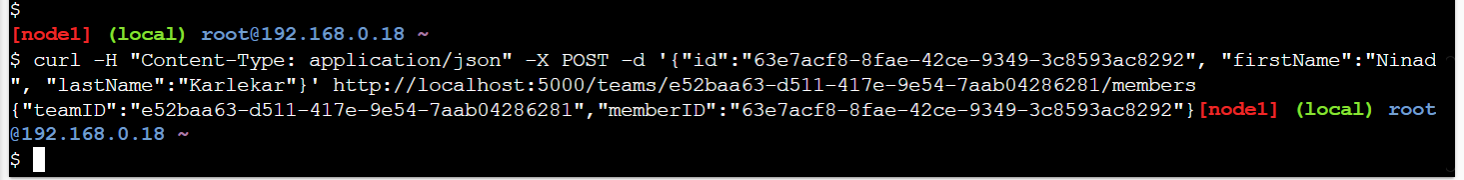


Command : to add new member to teamab

curl -H "Content-Type:application/json" -X POST -d \

'{"id":"63e7acf8-8fae-42ce-9349-3c8593ac8292", "firstName":"Ninad", "lastName":"Karlekar"}' http://localhost:5000/teams/e52baa63-d511-417e-9e54-7aab04286281/members

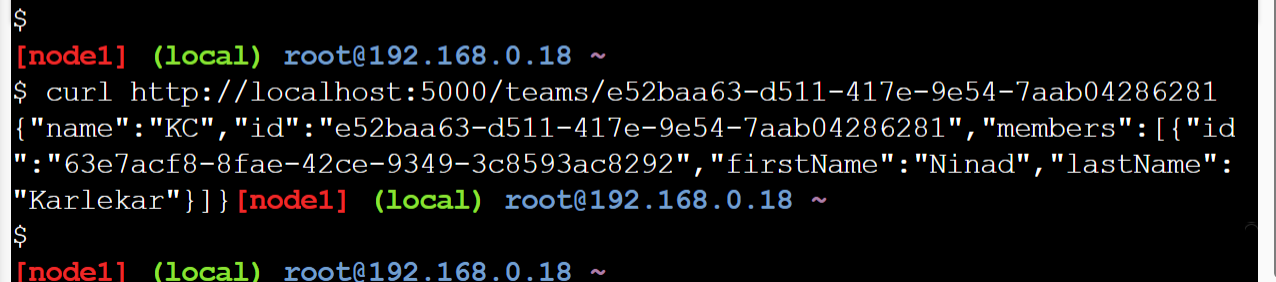
Output:



Command :To confirm member added

curl http://localhost:5000/teams/e52baa63-d511-417e-9e54-7aab04286281

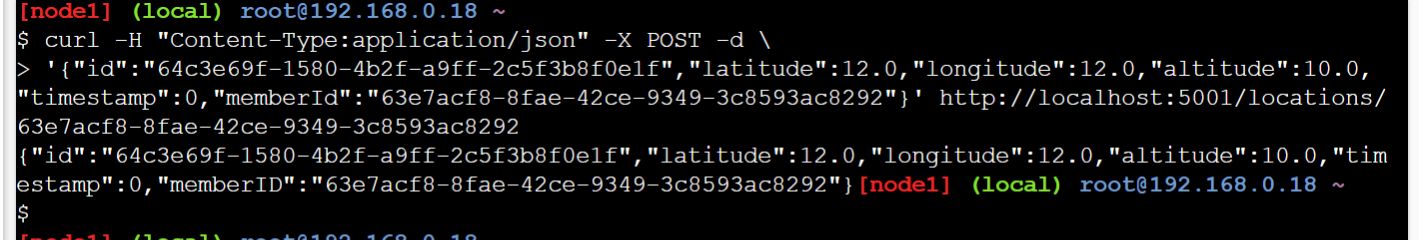
output:



Command : To add location for member

curl -H "Content-Type:application/json" -X POST -d \

'{"id":"64c3e69f-1580-4b2f-a9ff-2c5f3b8f0e1f","latitude":12.0,"longitude":12.0,"altitude":10.0, "timestamp":0,"memberId":"63e7acf8-8fae-42ce-9349-3c8593ac8292"}' <http://localhost:5001/locations/63e7acf8-8fae-42ce-9349-3c8593ac8292>



Command : To confirm location is added in member

curl http://localhost:5001/locations/63e7acf8-8fae-42ce-9349-3c8593ac8292

output:

