Pragyan 2023

Digital Twin Workshop

Hands on session

March 2023

www.brillio.com

Contents

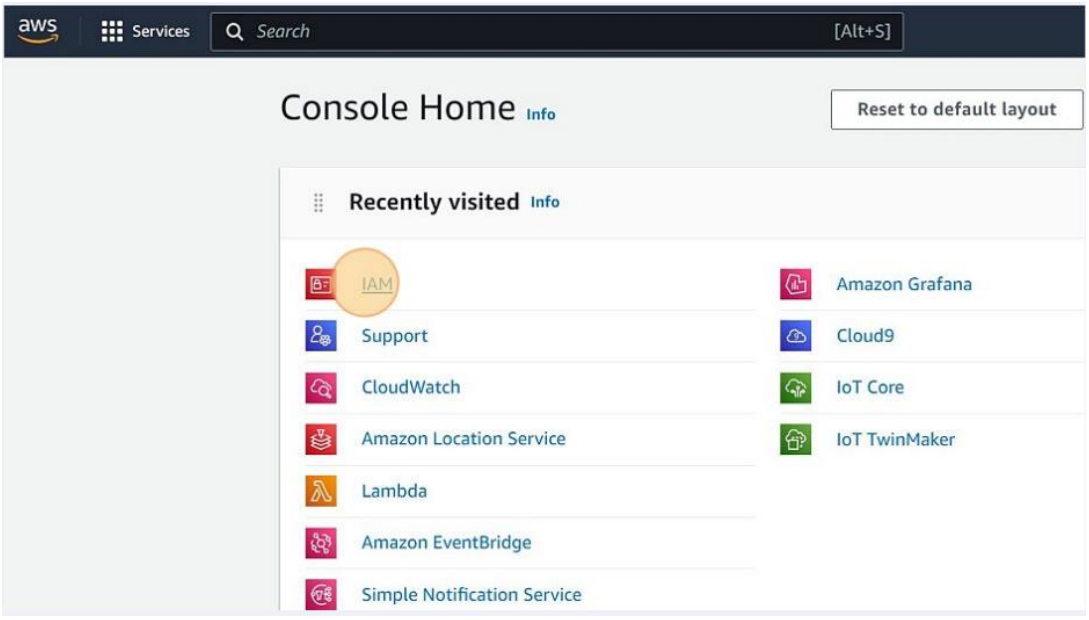
Hands-on on AWS Digital Twin (IOT TwinMaker)

1.1 Prerequisites

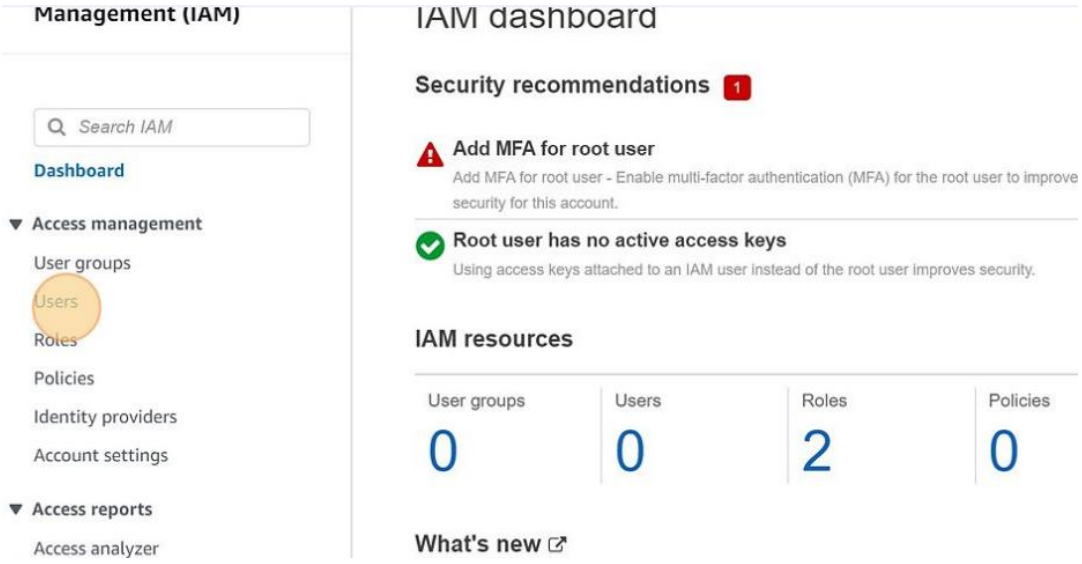
- Download hands on pre-requisite files from Github link: https://github.com/kumarvaibhav7/Workshop-Assets
- Login Cloud Computing Services - Amazon Web Services (AWS) to with your AWS Root Account.

1.2 Creating IAM User

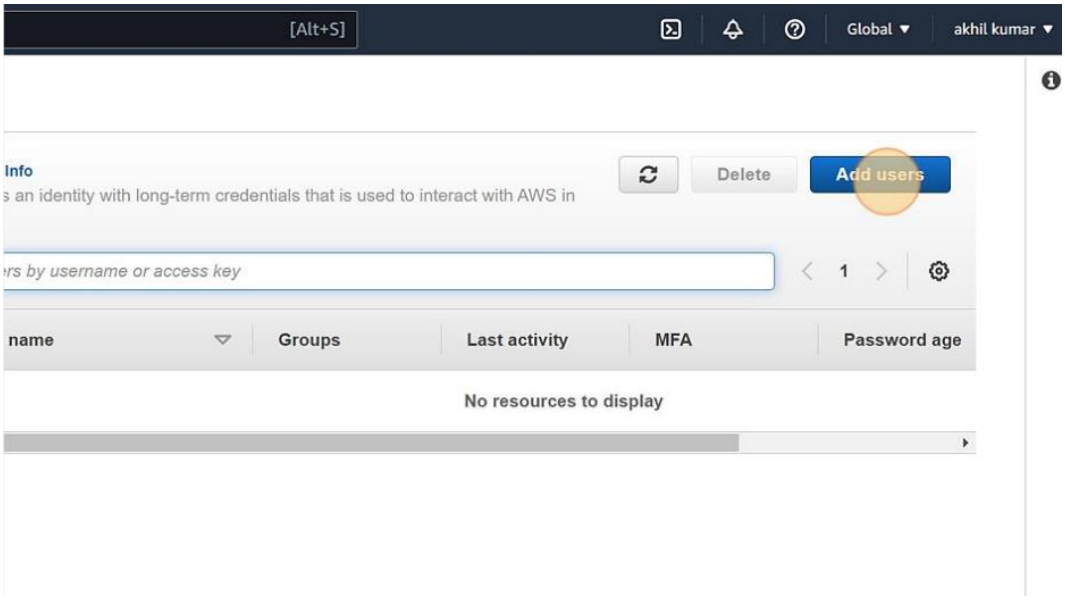
- Click “IAM”



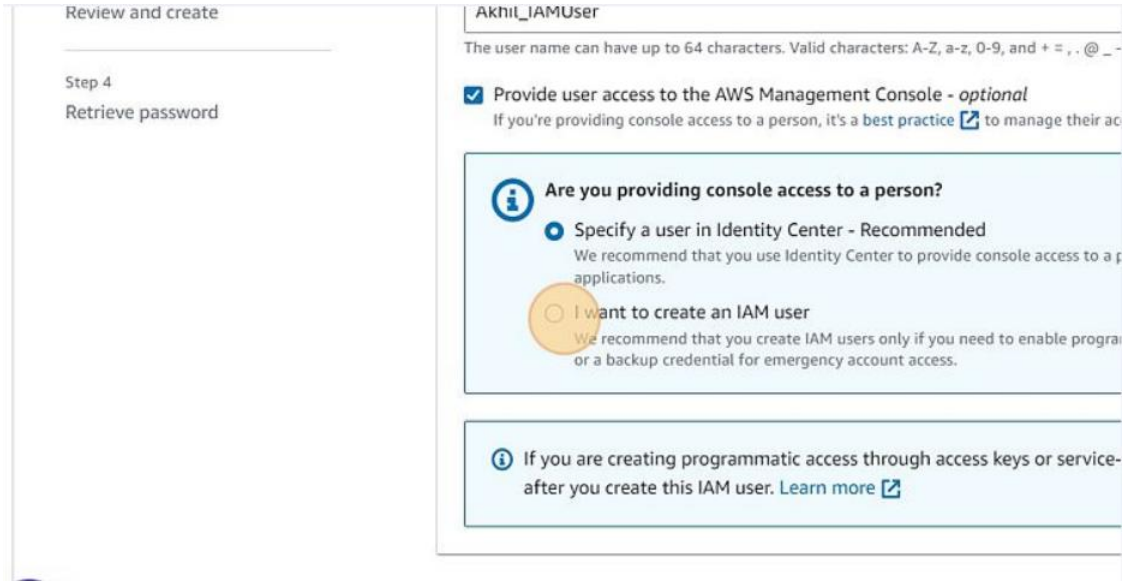
- Click "Users"



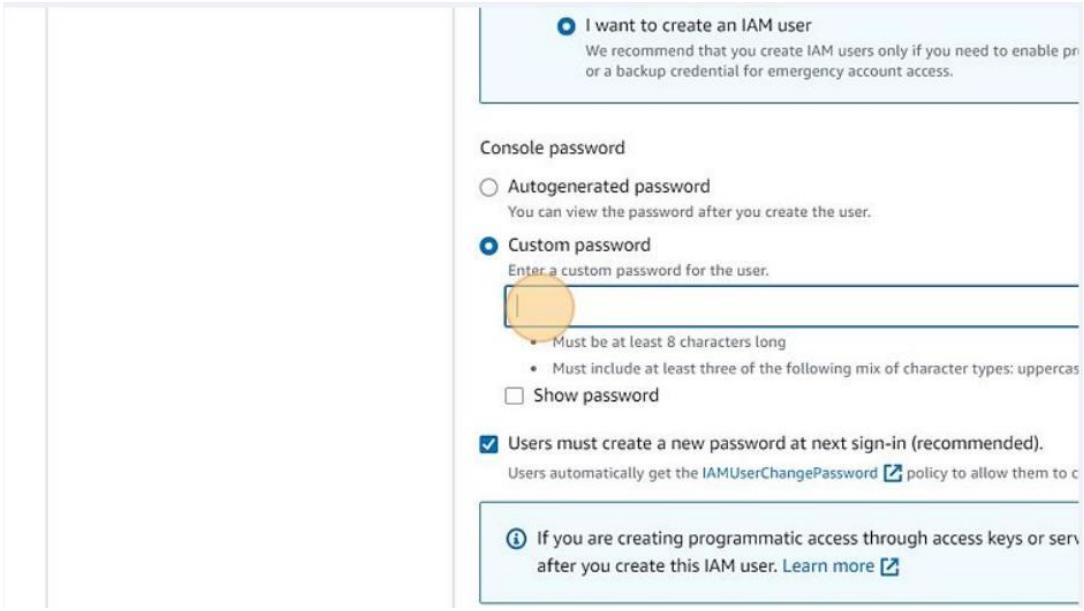
- Click "Add users"



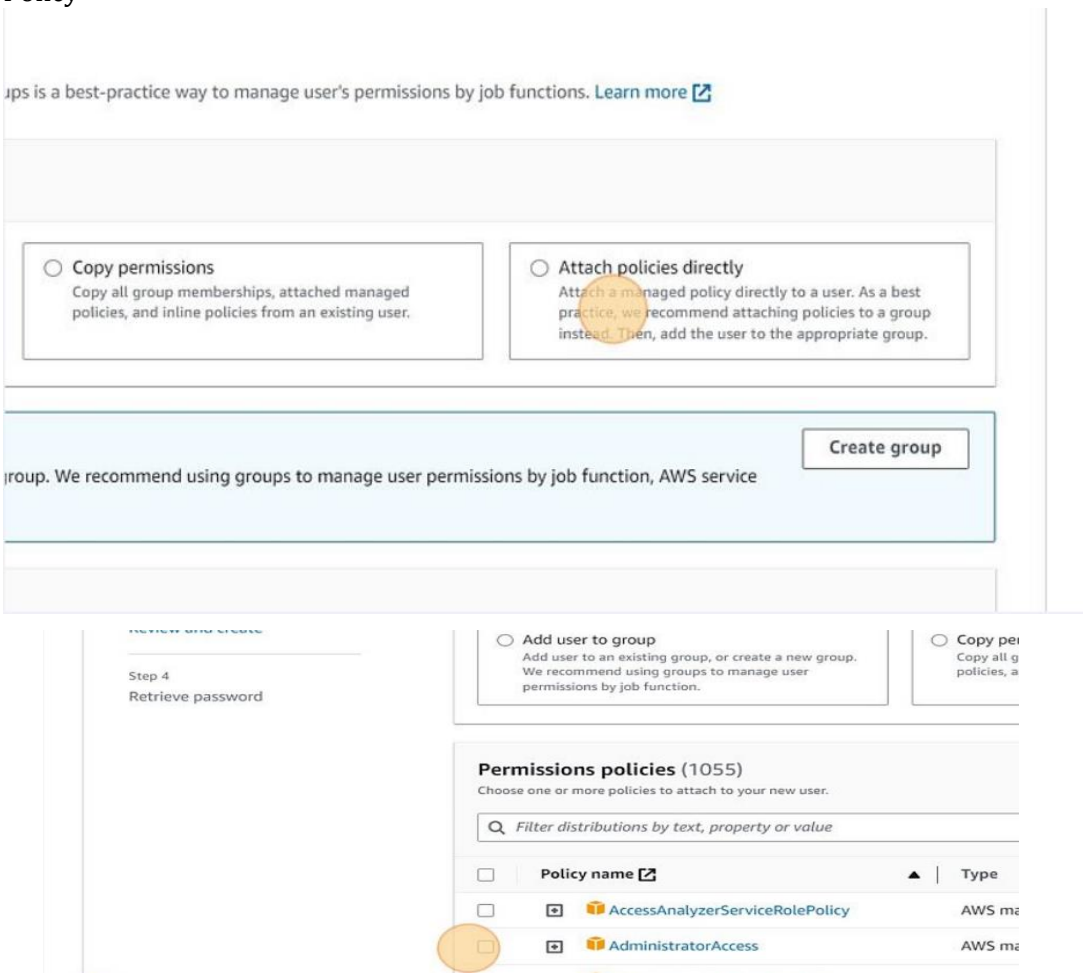
- Enter Username of Your Choice and Click on the checkbox “Provide user access to the AWS Management Console” and after that click on bullet point “I want to create an IAM User”



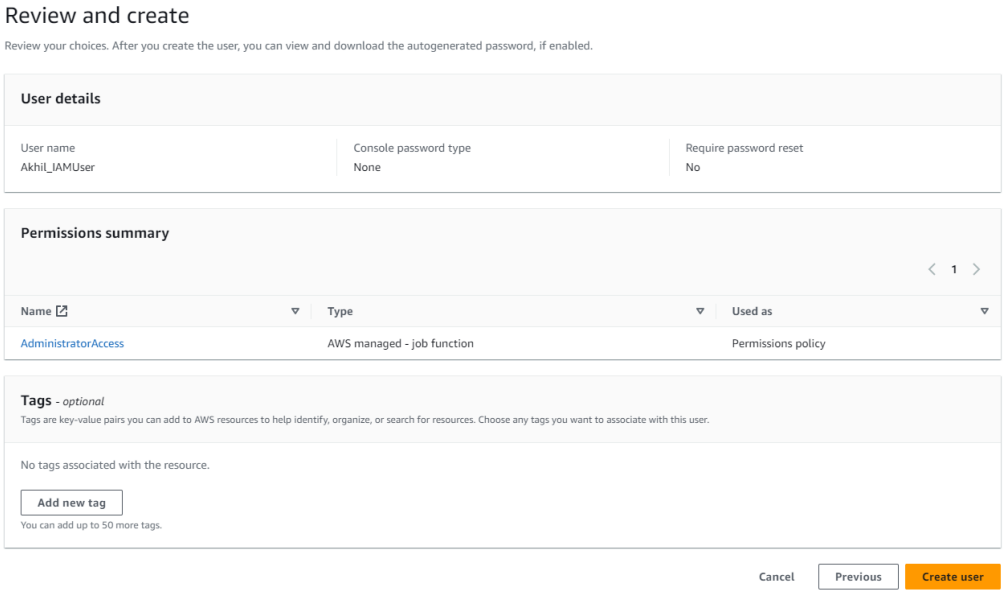
- Click on the “Custom Password” and give the password of your choice



- Click Next & Click on the Option Attach Policies Directly and attach Administrator Access Policy



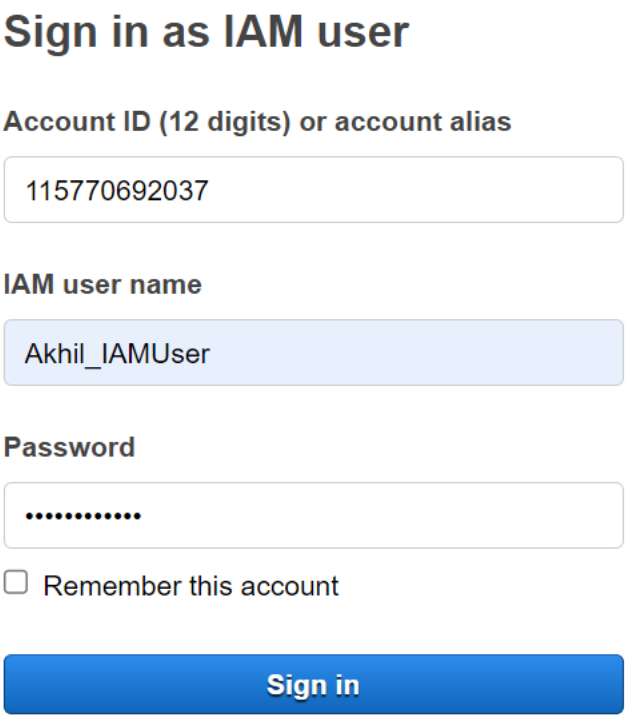
- Click on “Next” and review the policy attached and click on “Review and Create”



- Once the user is created note the console sign-on url and download the csv file

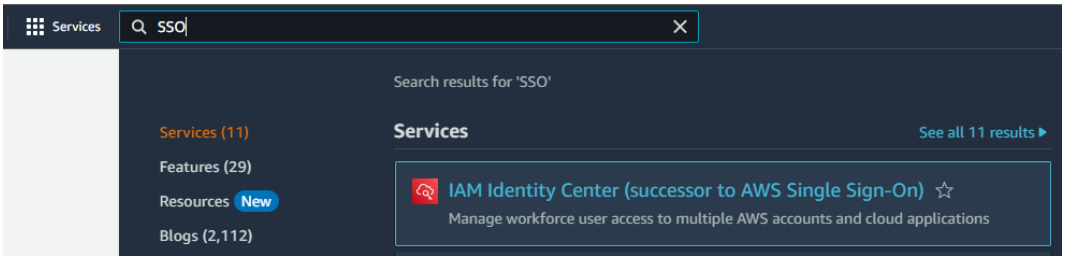
1.2.1 Sign in with IAM User

- You can login with the console url you copied previously and enter the username and password

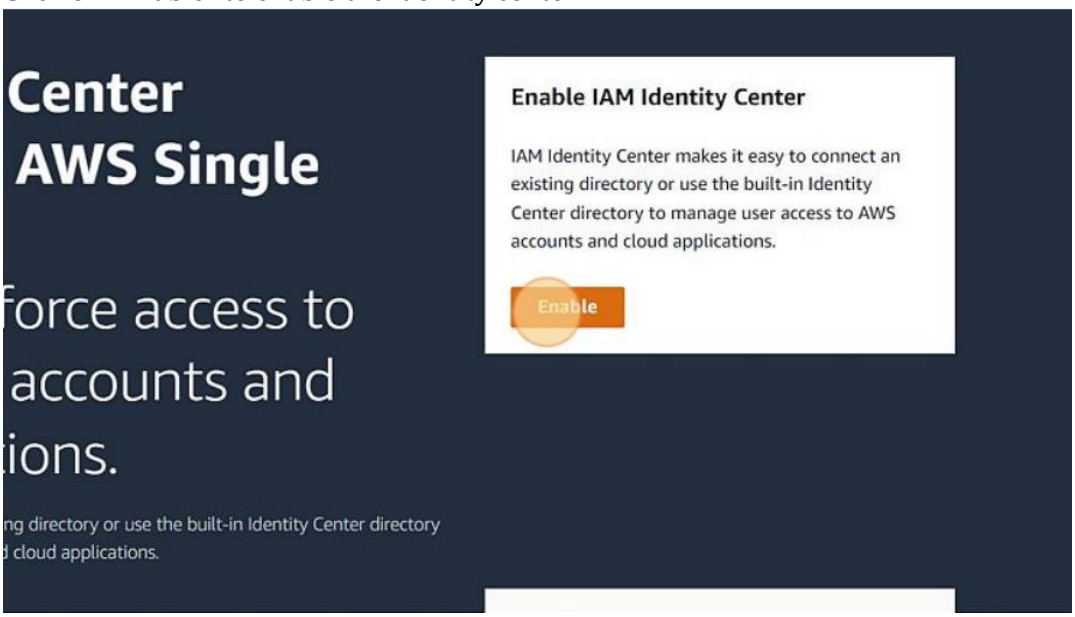


1.2.2 Creation SSO User

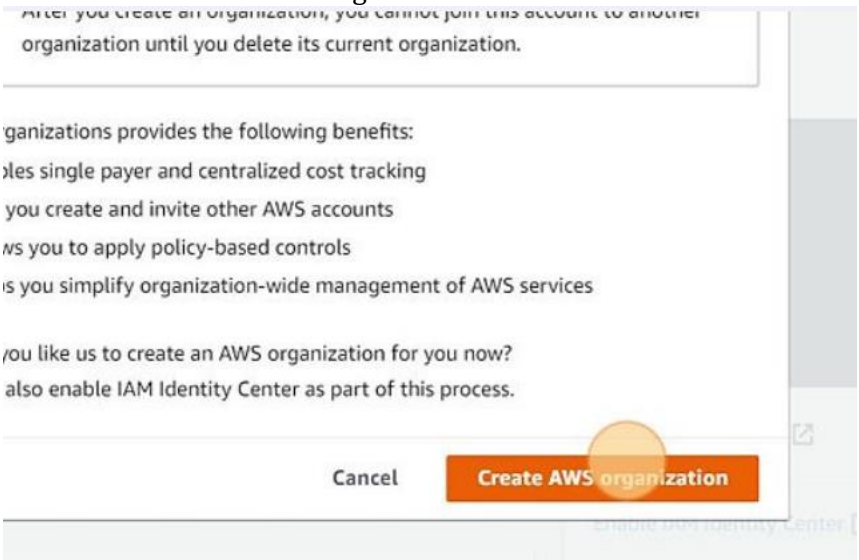
- Click on the search bar and search for SSO and click on “IAM Identity Center”



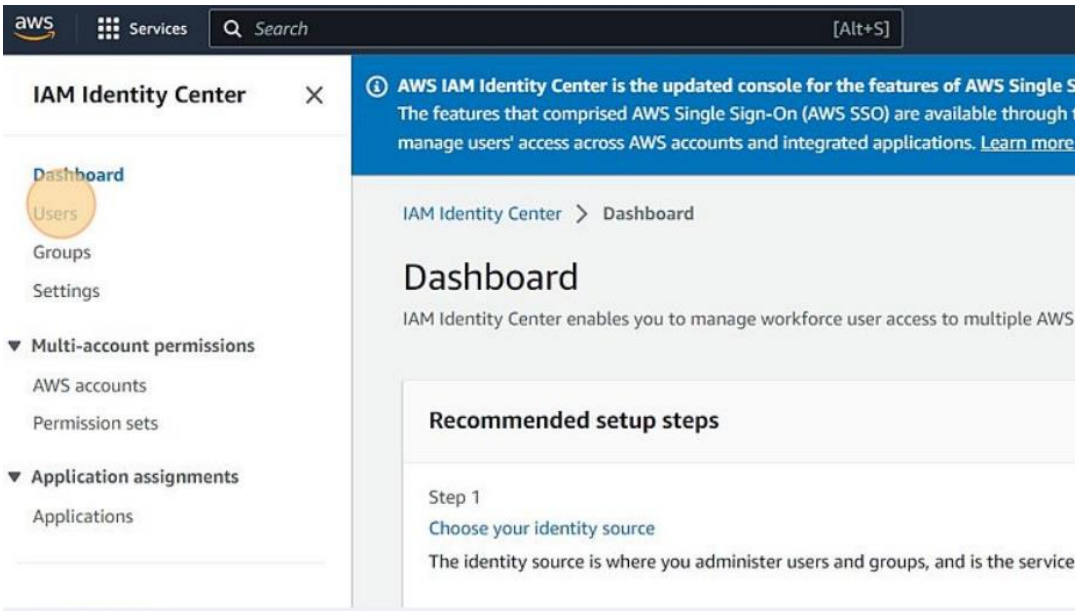
- Click on “Enable” to enable the identity center



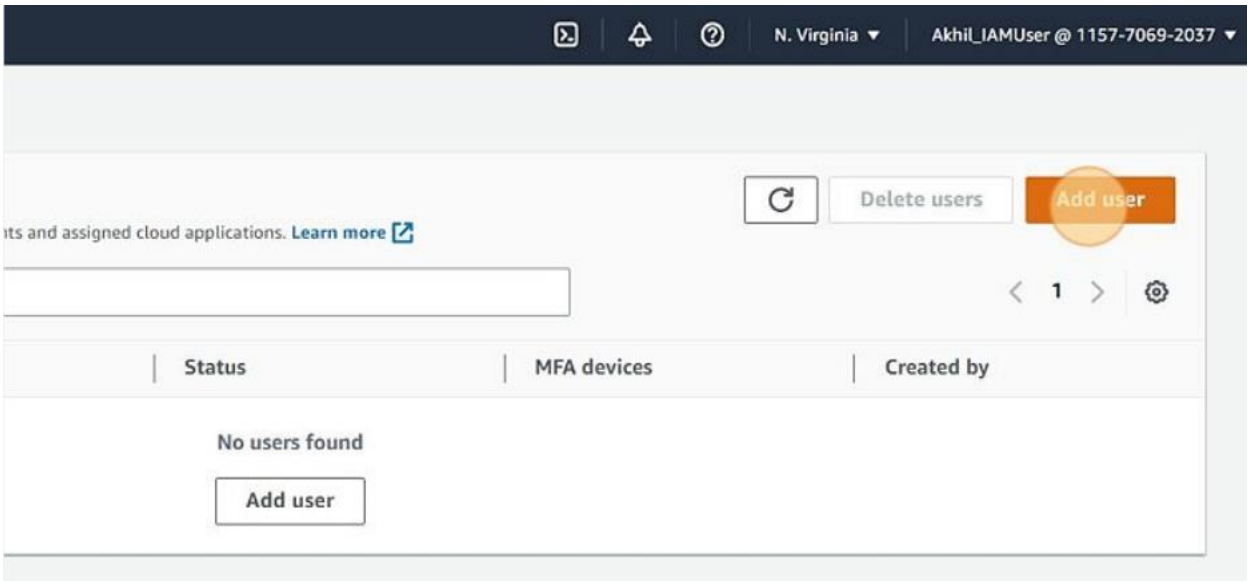
- Click on the “Create AWS Organization”



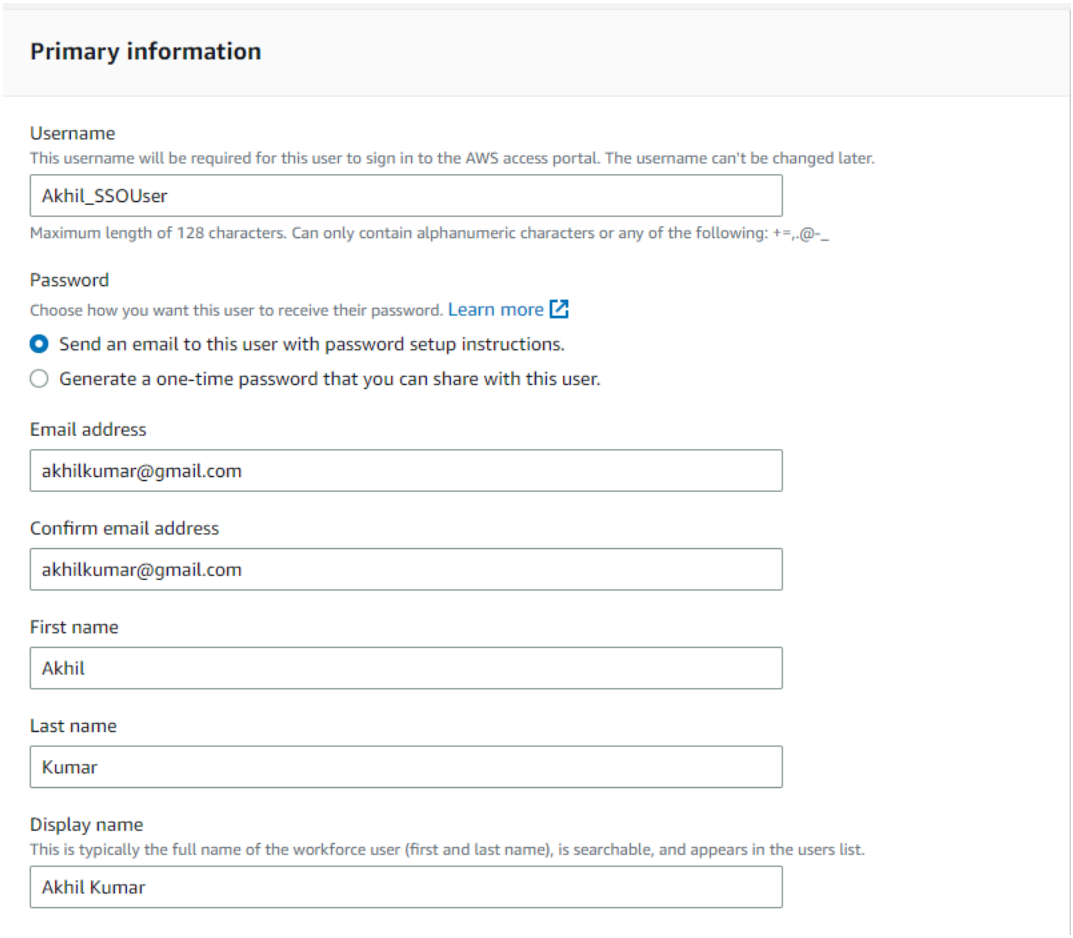
- Click "Users"



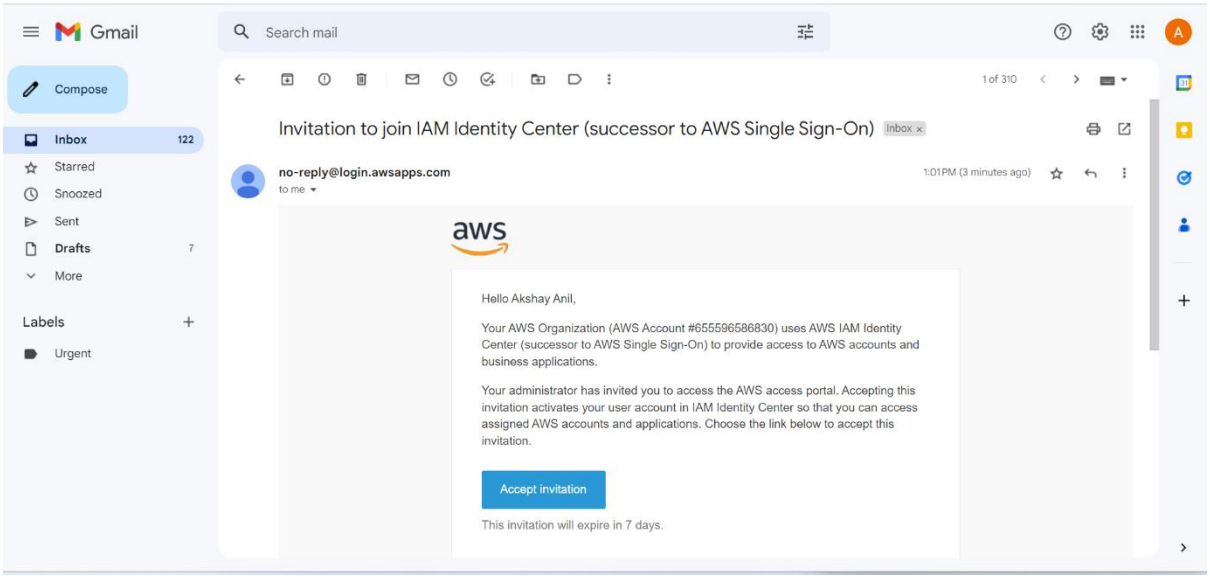
- Click "Add user"



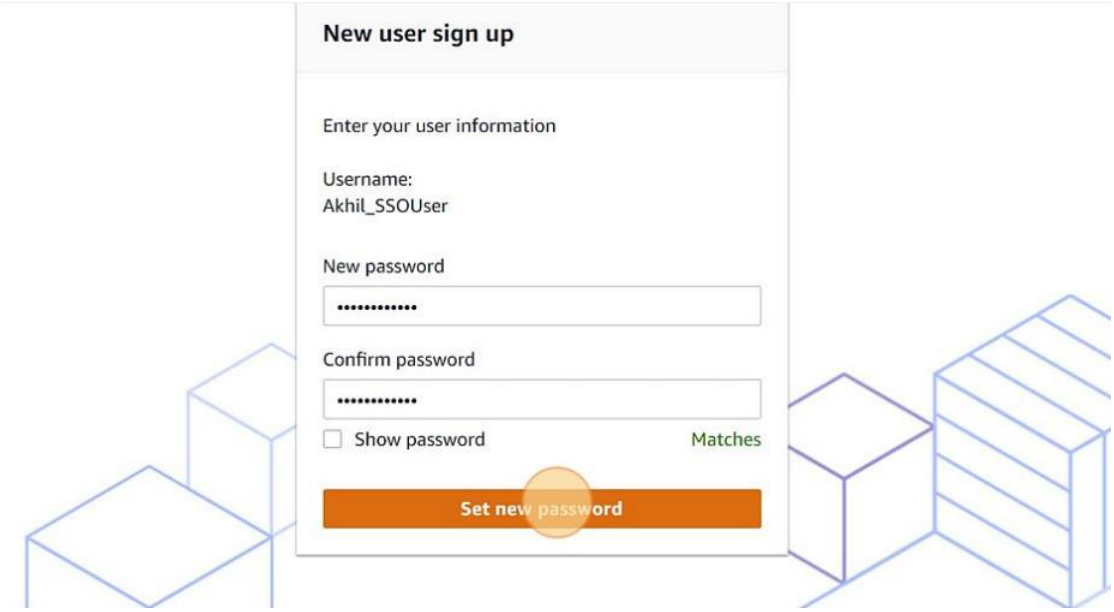
- Fill in all the necessary details for creating the SSO User



- Click Next and Create the user and check your email to accept the invitation

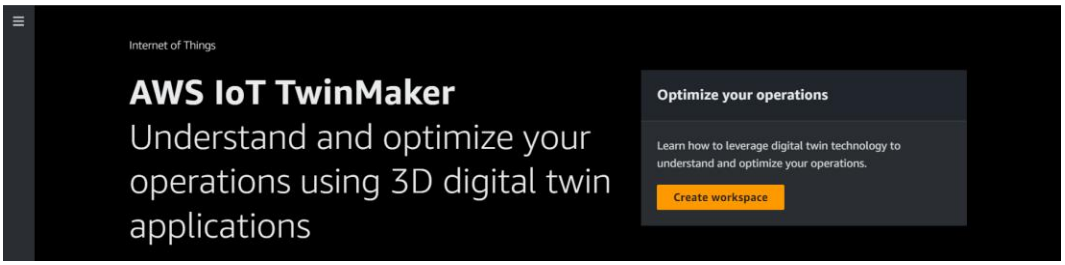


- After clicking on the accept invitation you have to set the password for the SSO User

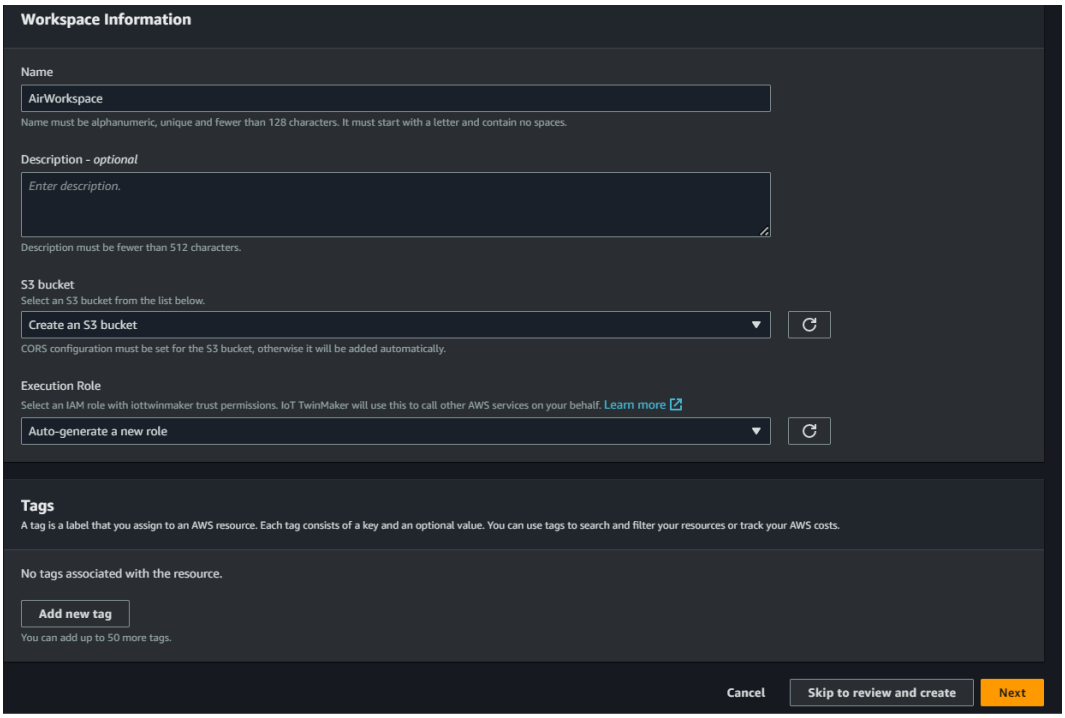


1.3 Creating Digital Twin Instance

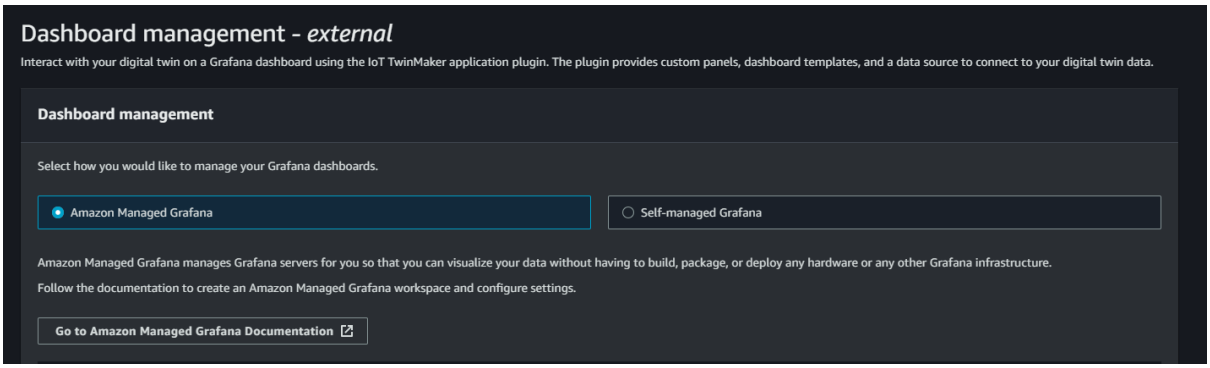
- Search for IOT Twin maker And open the service - AWS IOT Twin Maker and Click on "Create Workspace"



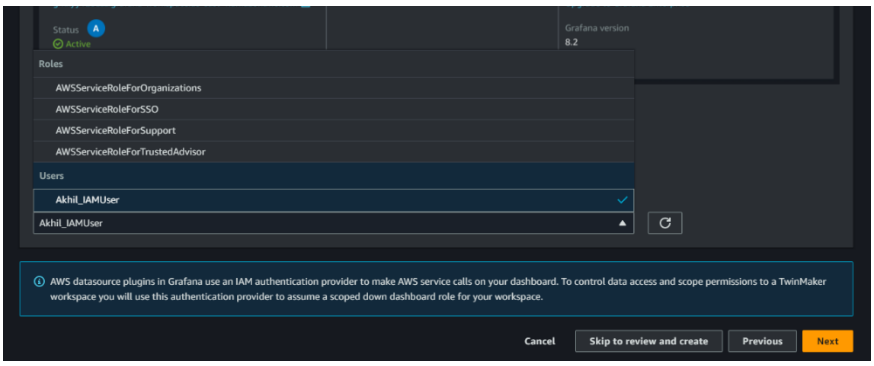
- Enter workspace name as "AirWorkspace"
- While Selecting S3 Bucket – Select "Create a S3 Bucket"
- For Execution Role "Auto Generate a new role"



- Click "Next" and ensure that "Aws Managed Grafana" is selected in dashboard management

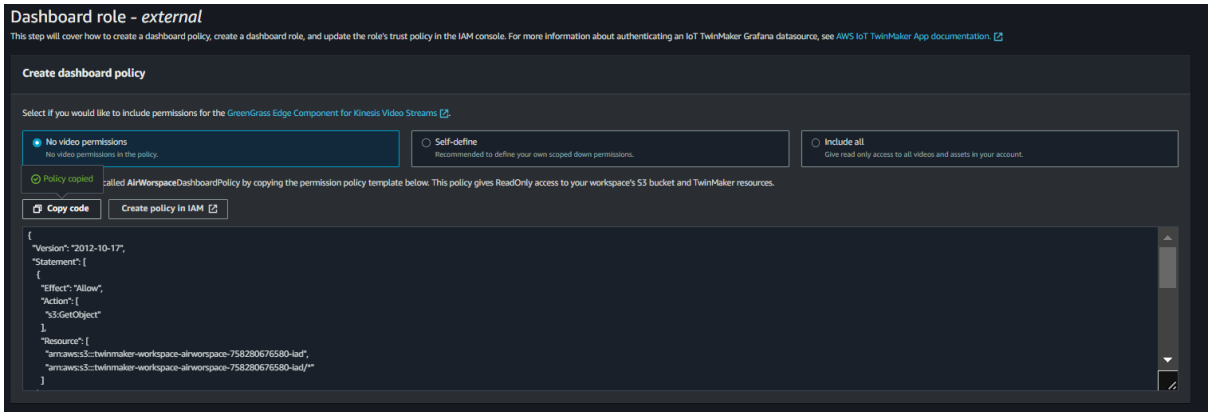


- Under grafana authentication account choose the IAM User and click "Next"

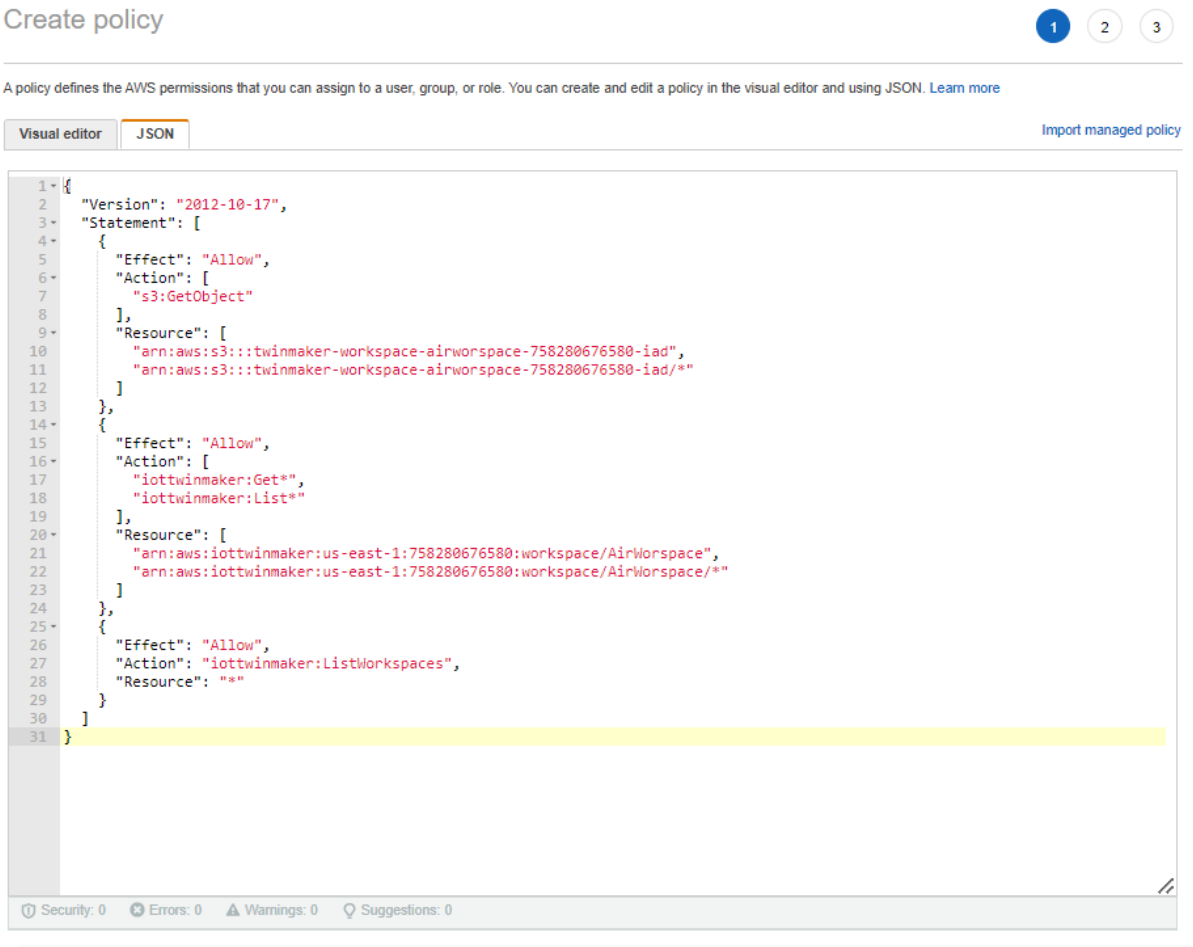


1.3.1 Create Dashboard Policy

- Click copy code and then Click on “Create Policy In IAM”

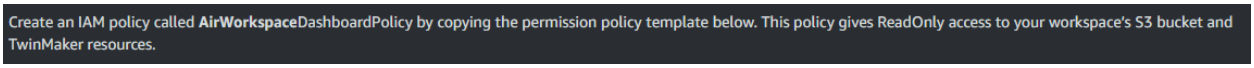


- IAM Console would pop-up and select JSON and paste the code you have copied previously

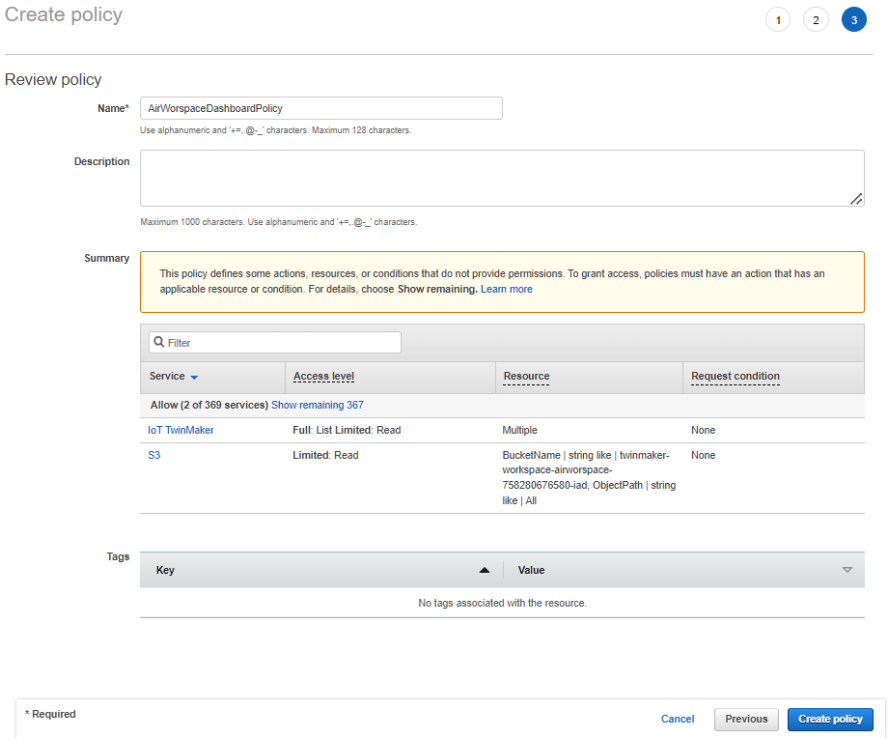


- Click on “Next Tags” and again click on “Next Review”

- Provide the name which is available in Twinmaker console

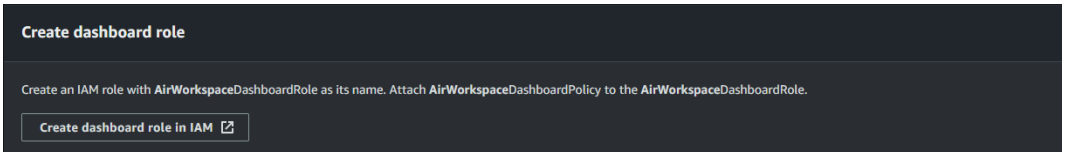


- Verify the Name and click "Create Policy"



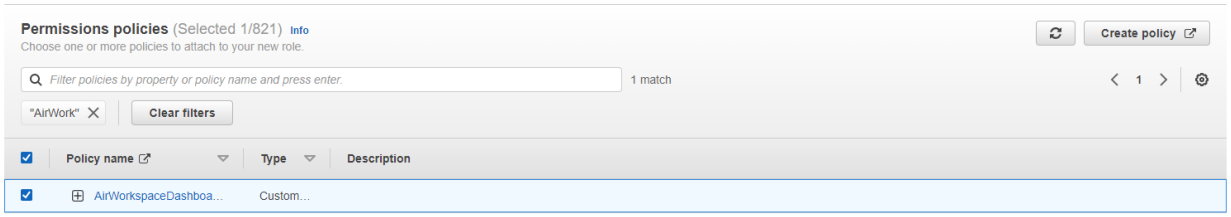
1.3.2 Create Dashboard Role

- Click on the "Create dashboard role in IAM"

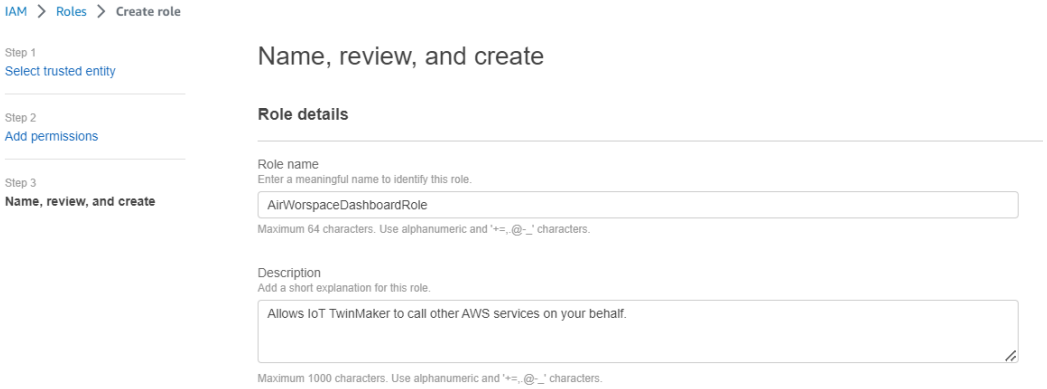


- Create a role Under "IOT Twin Maker Service" and click "Next".

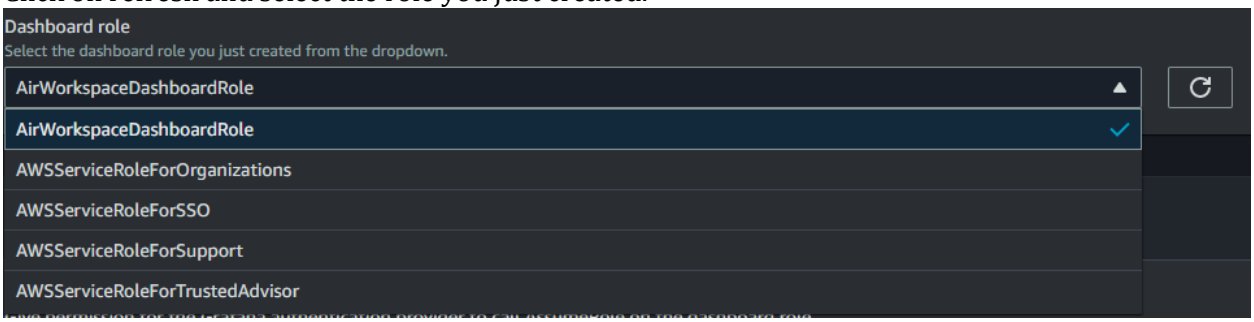
- Search for the dashboard policy (AirWorkspaceDashboardPolicy) and attach it and click "Next".



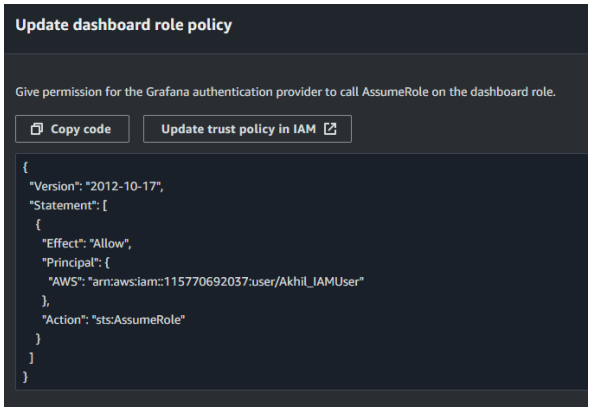
- Give the role name as defined in the console as "AirWorkspaceDashboardRole" and click "Create Role"



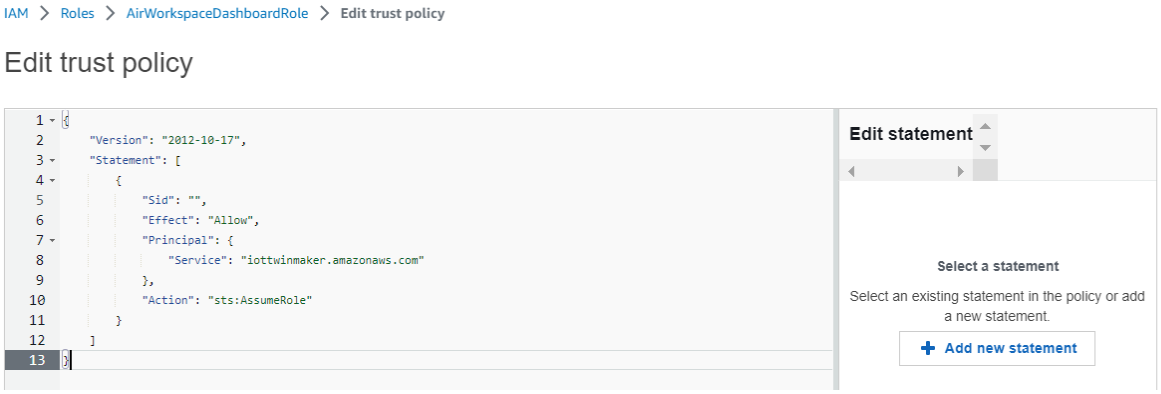
- Click on refresh and select the role you just created.



- Click on copy code and then click on "Update trust policy in IAM"



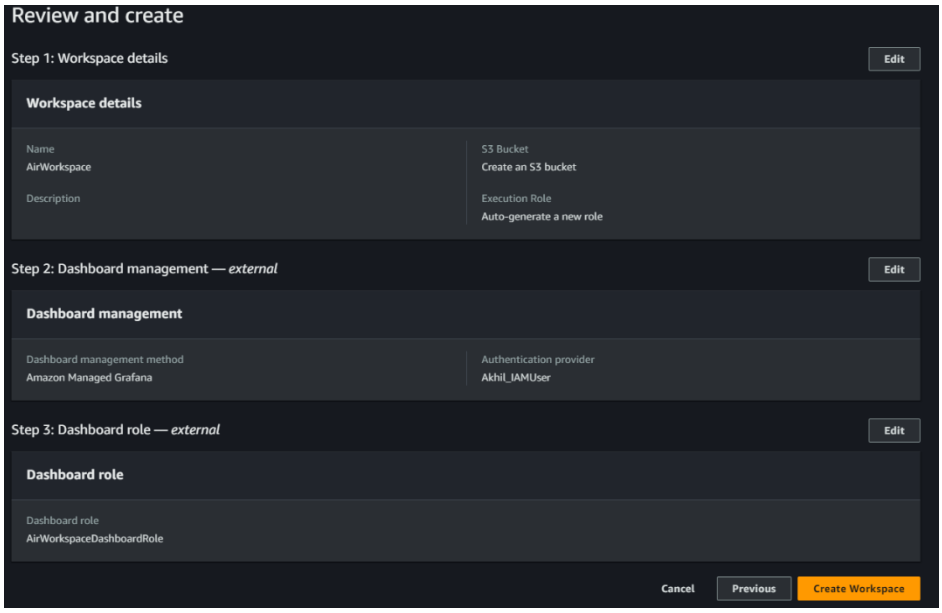
- The new tab which is opened will give you the option to edit the trust relationship of your dashboard role .



- Clear the prewritten code and update the code with the one you copied and click on update policy

1.3.3 Review & Create Workspace

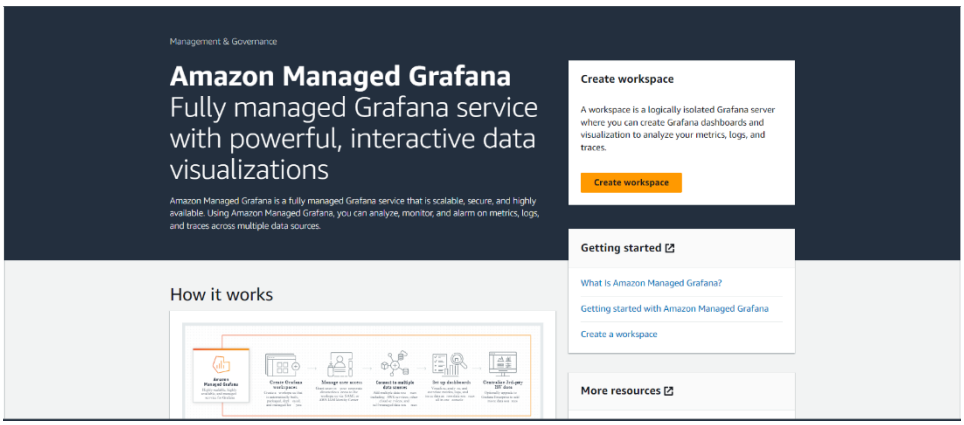
- Verify all the three steps and click on "Create Workspace"



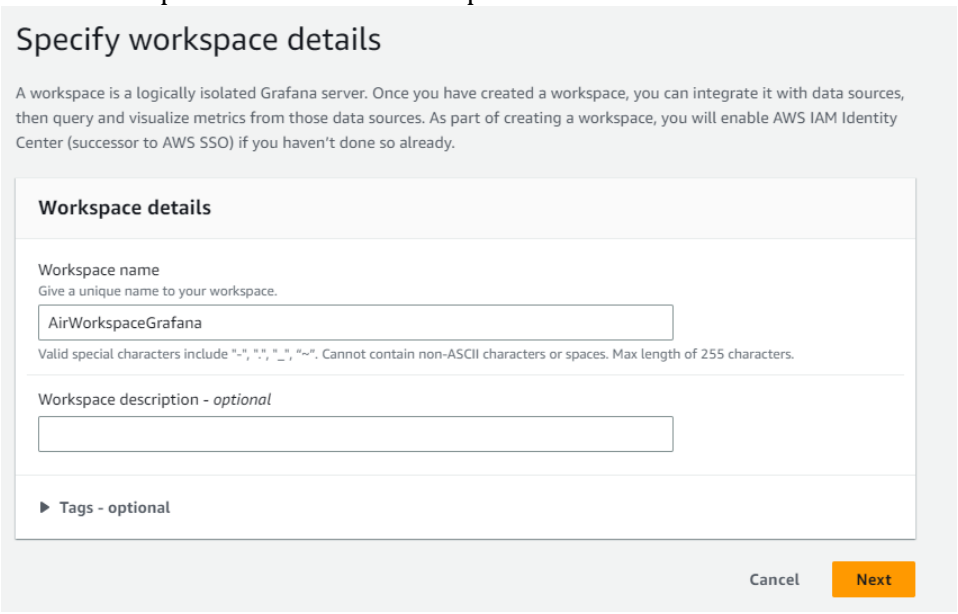
1.4 Configuring Visualization

1.4.1 Creating Workspace

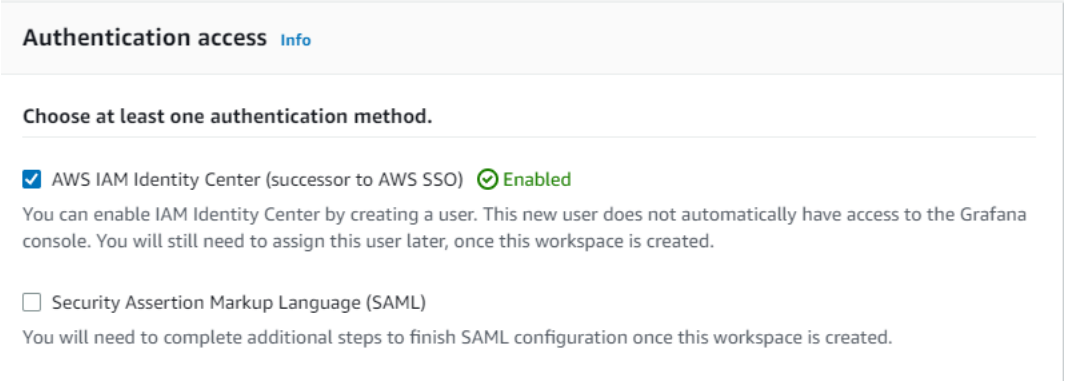
- Search for "Amazon Grafana" and click on "Create Workspace"



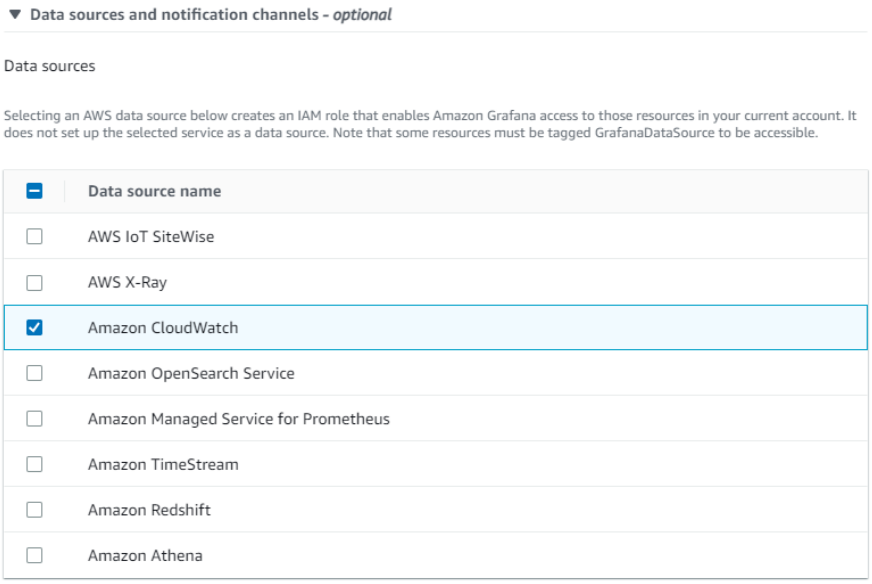
- Enter Workspace name as "AirWorkspaceGrafana" and click "Next"



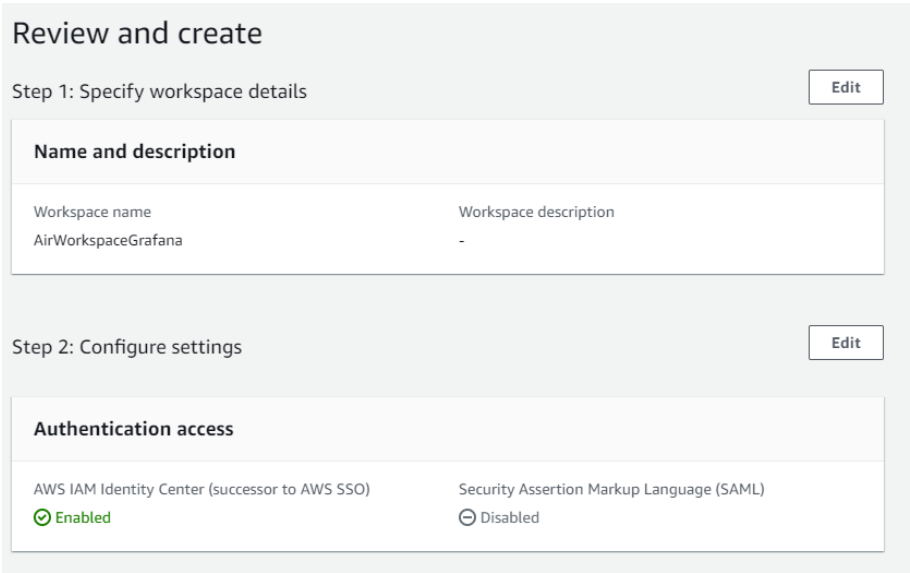
- Select Authentication Access as "Aws IAM Identity Center" and wait for it to get enabled and click "Next"

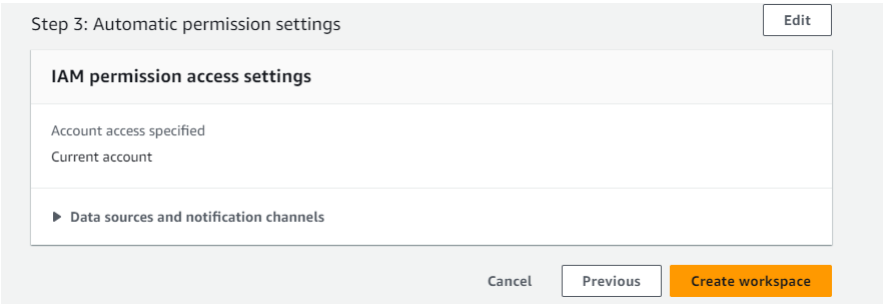


- Under data sources and notification channels select “Amazon CloudWatch” and click “Next”

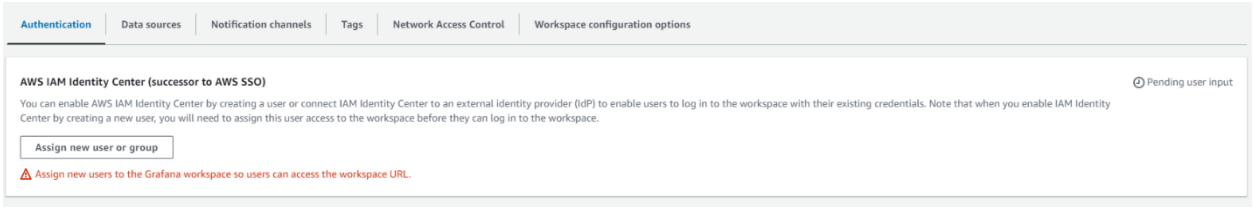


- Verify the workspace details and authentication access and click on “Create Workspace”

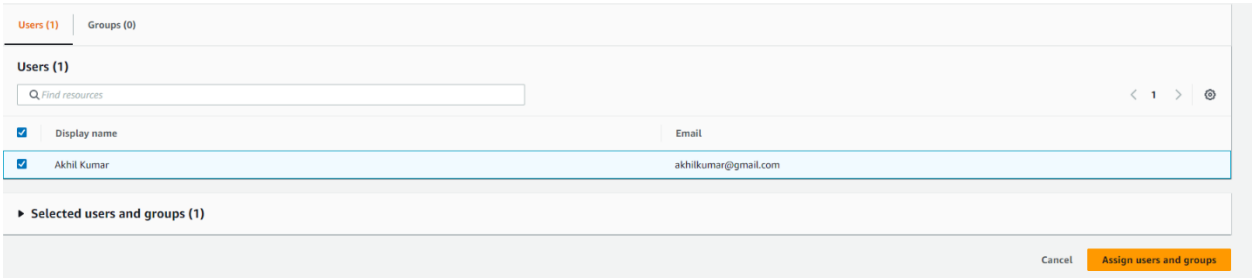




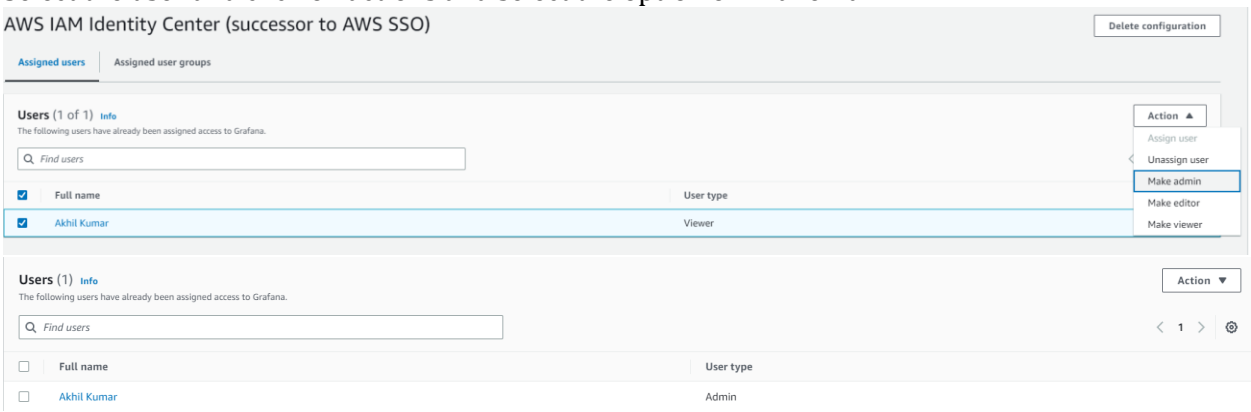
- Click on “Assign new user or group” under Authentication



- Select the SSO user which we created in 1.2.2 and click on “Assign users and groups”

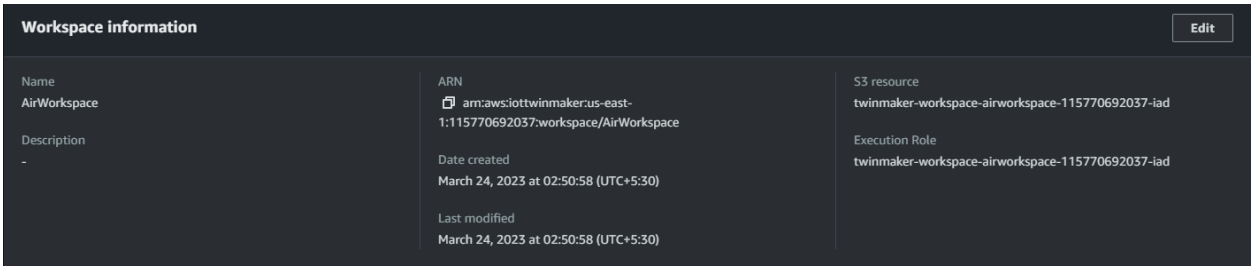


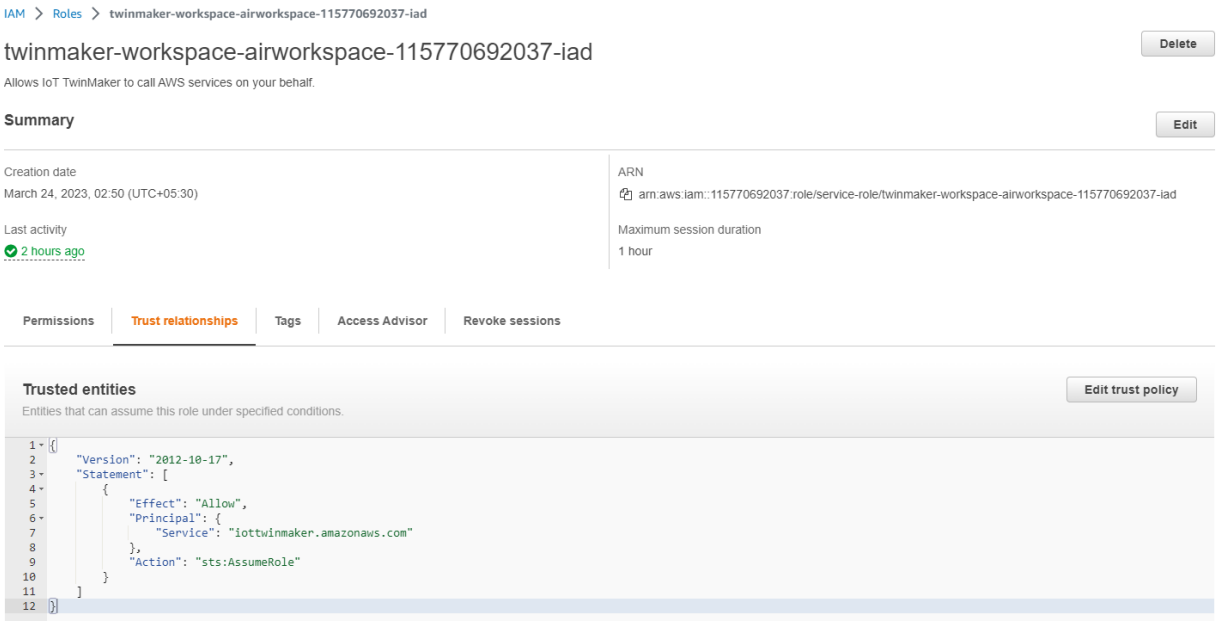
- Select the user and click on actions and select the option of “Make Admin”



1.4.2 Connecting Grafana and IOT-Twinmaker

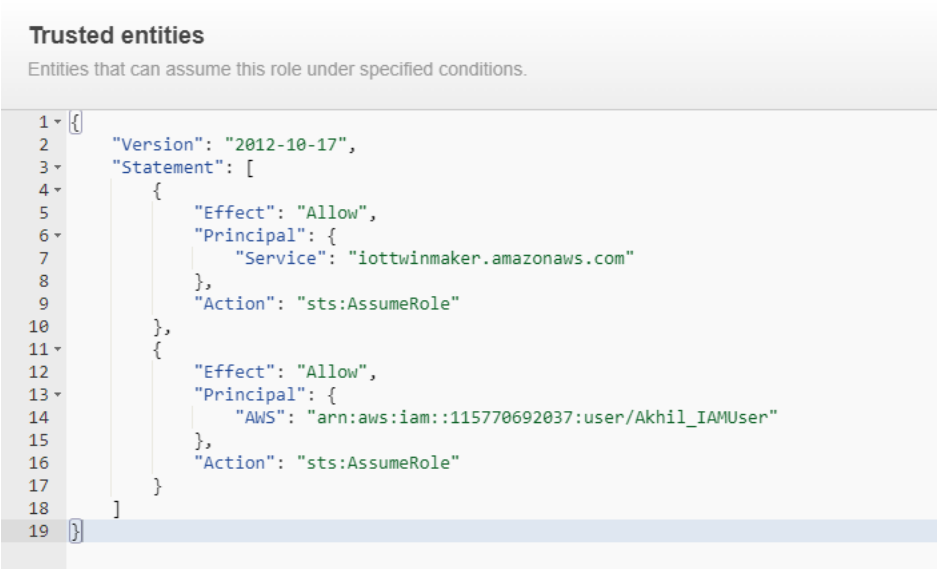
- Go to Twin maker and copy the execution role and open it in IAM console and open the trust relationship in it.



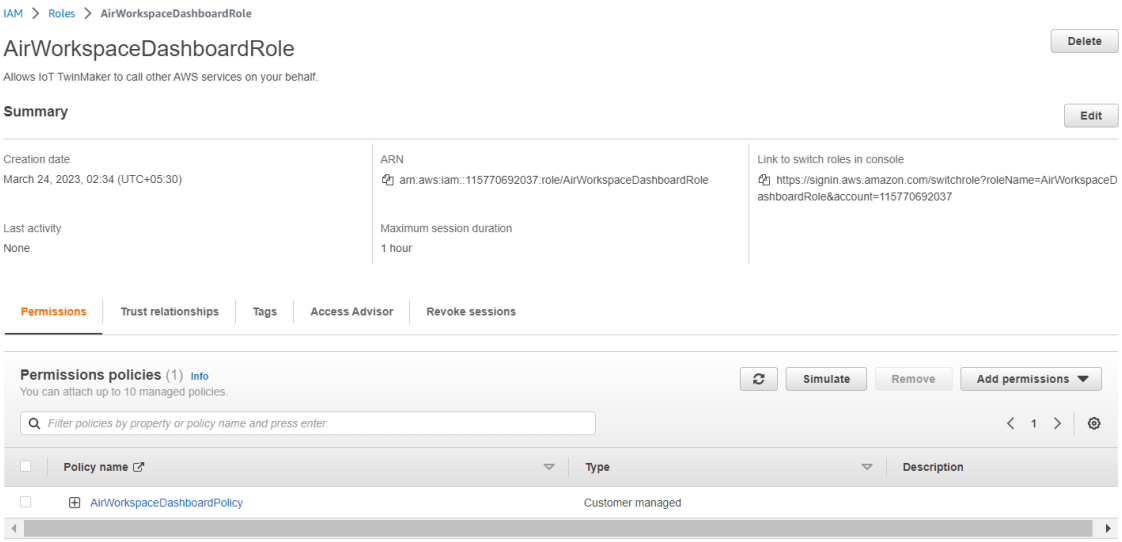


- Click on Edit Trust Policy and update it with the one provided below. Replace the brackets with your account id and IAM username

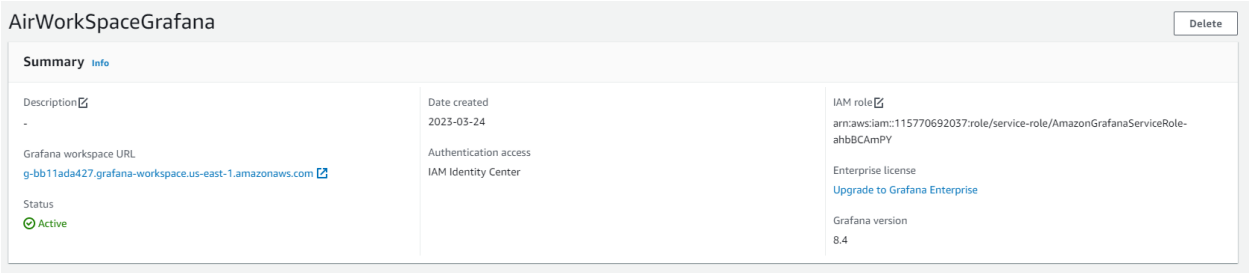
```
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Principal": {
"Service": "iottwinmaker.amazonaws.com"
},
"Action": "sts:AssumeRole"
},
{
"Effect": "Allow",
"Principal": {
"AWS": "arn:aws:iam::{Account ID}:user/{IAM Username} "
},
"Action": "sts:AssumeRole"
}
]
}
```



- Now open the dashboard role you created in 1.3.2 while creating the twinmaker workspace.

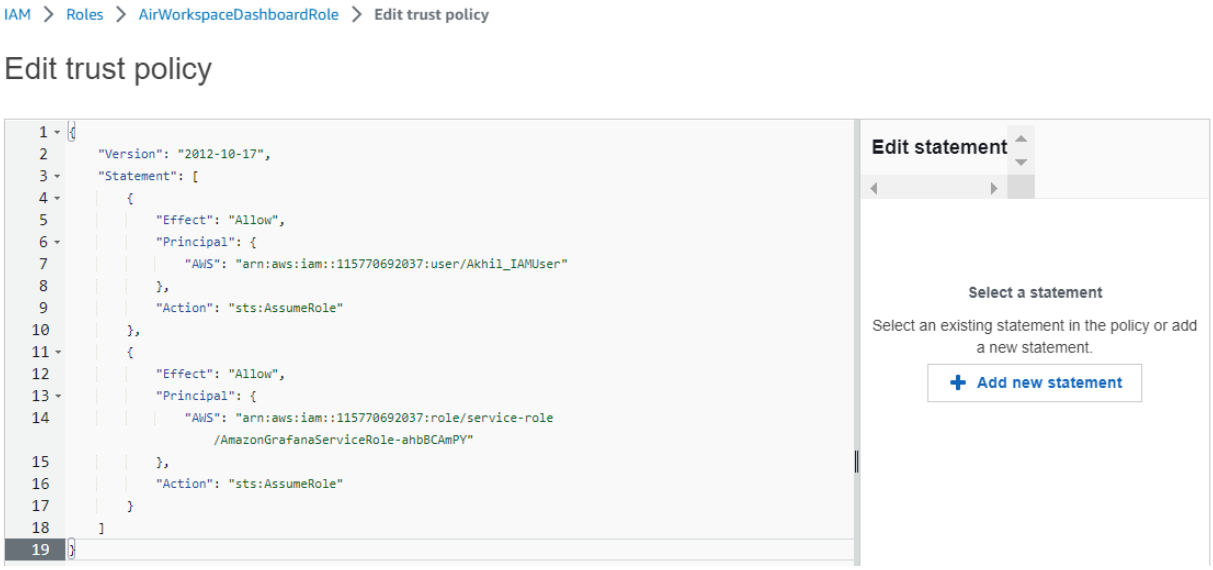


- Go to Grafana dashboard and copy the ARN provided



- Go to Dashboard Role and edit the trust relationship and replace the bracket with arn of grafana

```
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Principal": {
"AWS": "arn:aws:iam::115770692037:user/Akhil_IAMUser"
},
"Action": "sts:AssumeRole"
},
{
"Effect": "Allow",
"Principal": {
"AWS": "{PASTE IT HERE}"
},
"Action": "sts:AssumeRole"
}
]
}
```

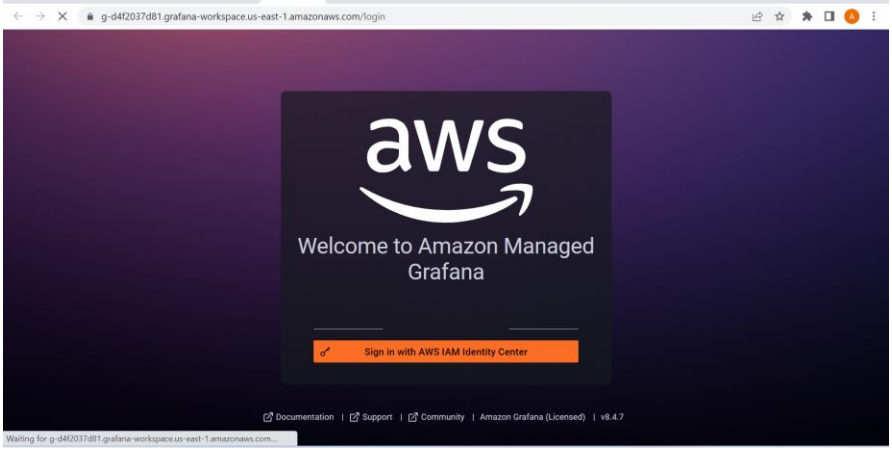


- Click “Update Policy”

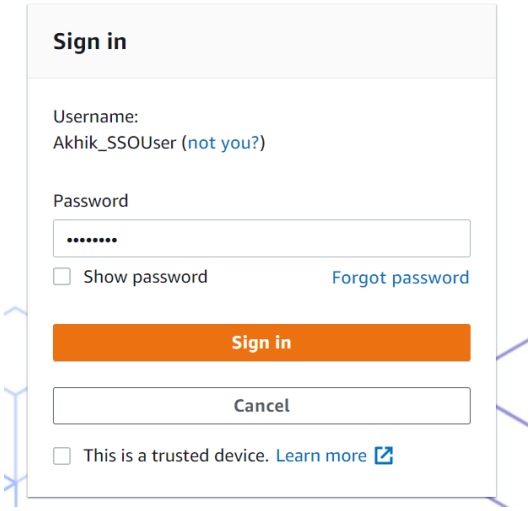
- Open the grafana dashboard using grafana workspace url

  Grafana workspace URL
  g-bb11ada427.grafana-workspace.us-east-1.amazonaws.com

- Open the url and click on “Sign in with AWS IAM Identity Center”

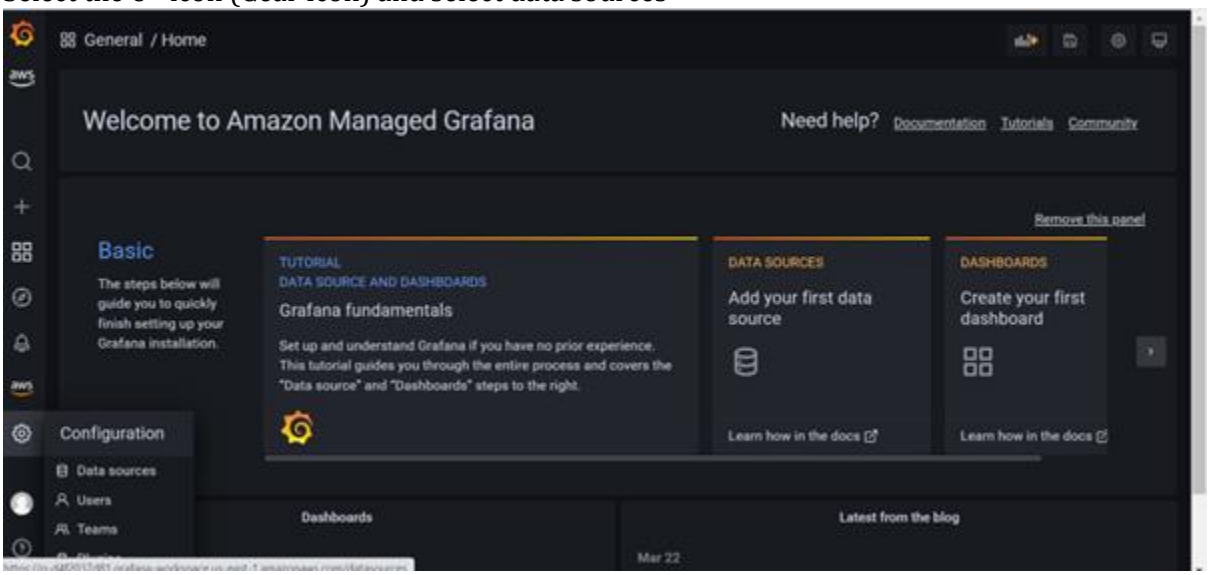


- Enter the Credentials and sign in

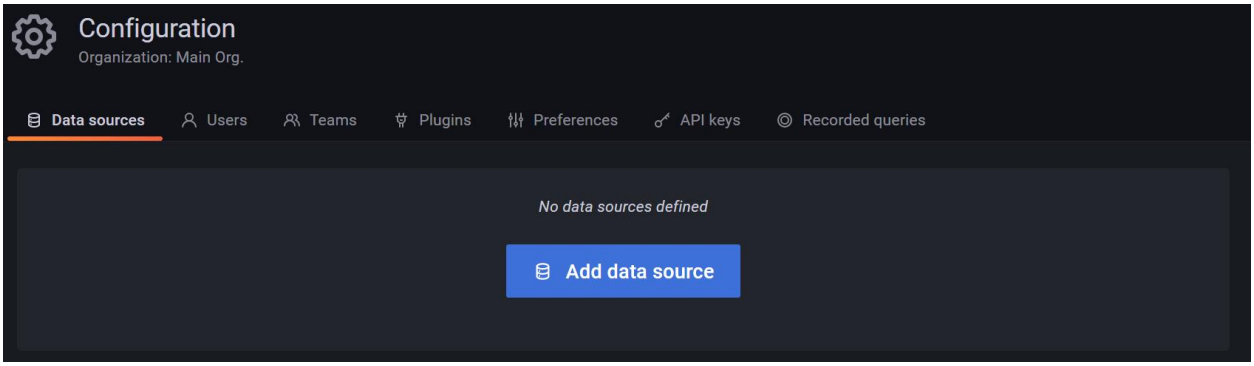


- Click “Sign In”

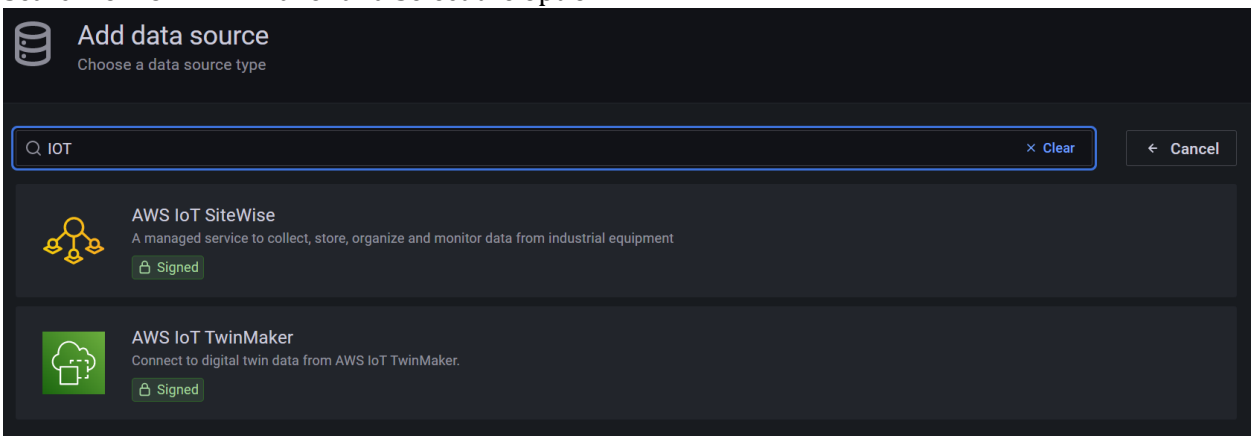
- Select the 6th icon (Gear Icon) and select data sources



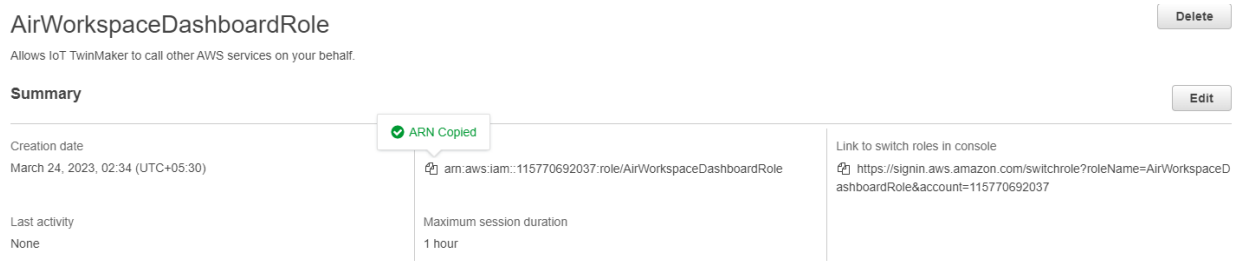
- Click on Add Data Source



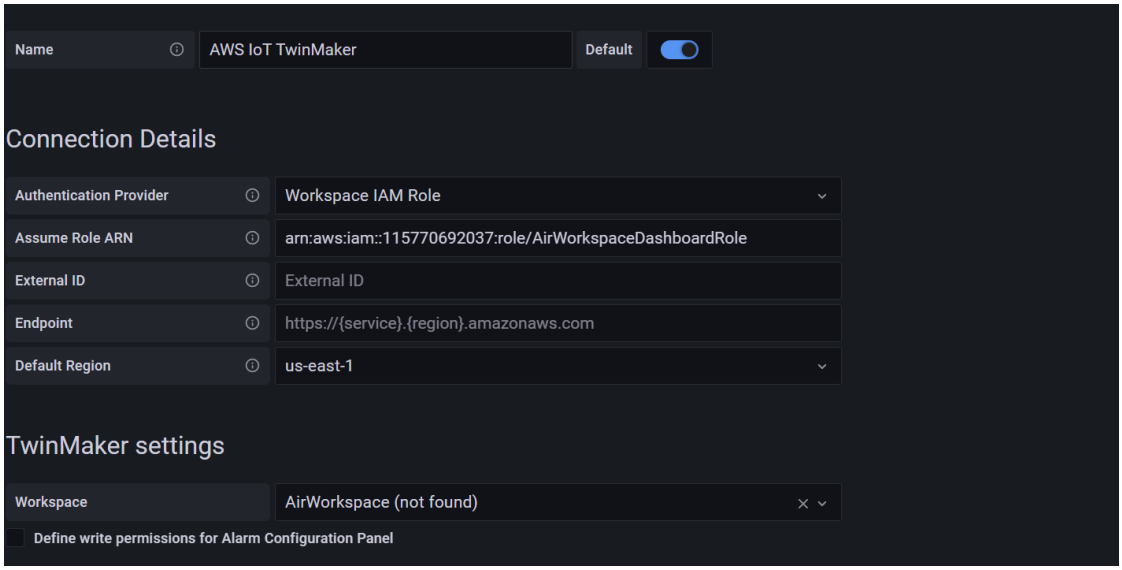
- Search for IOT Twinmaker and Select the option



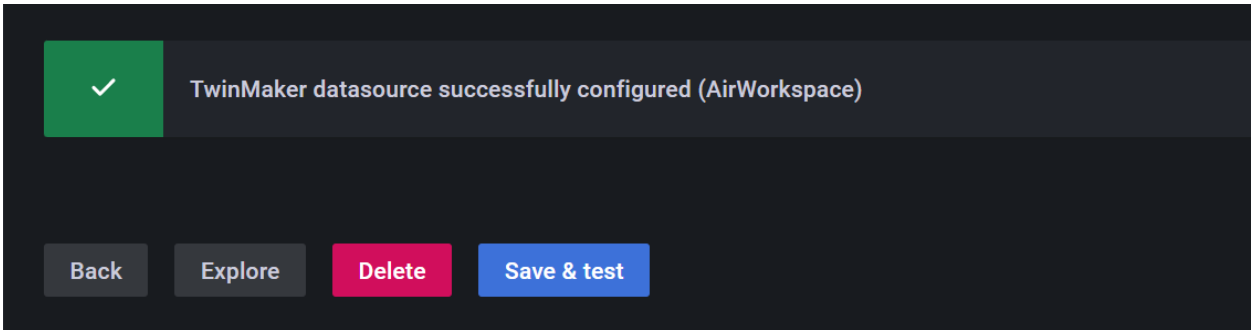
- Go to iam console and search for your dashboard role



- Paste the arn copied in “Assume Role Arn” and give workspace name as “AirWorkspace”



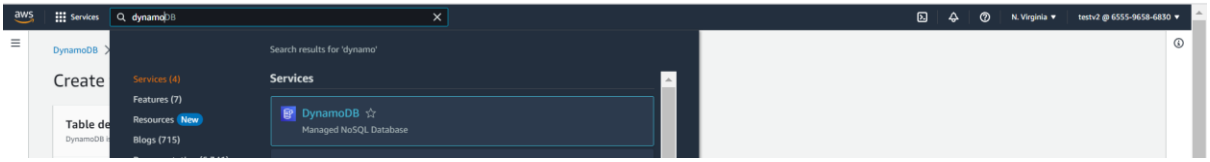
- Click on “Save & Test” & your connection is successfully added as a data source



1.5 Creating NoSql Database

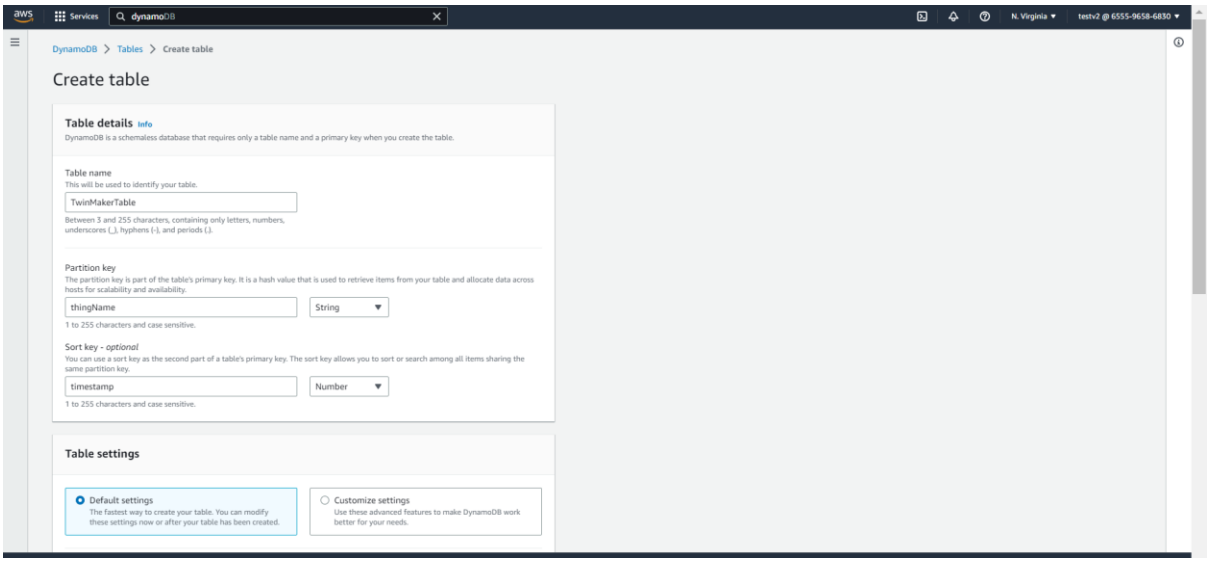
1.5.1 Creating Schema

- Go To DynamoDB Console

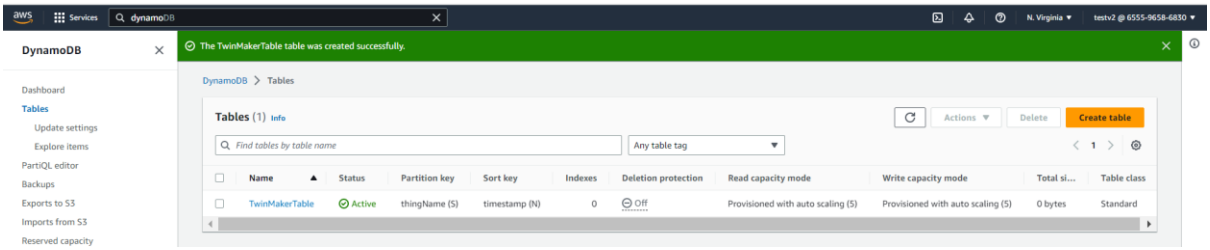


- Create a DynamoDB table named TwinMakerTable that contains the key thingName of type String as partition key and the key timestamp of type Number as Sort key.

  Note: names are case-sensitive as the same names are used to create schemas

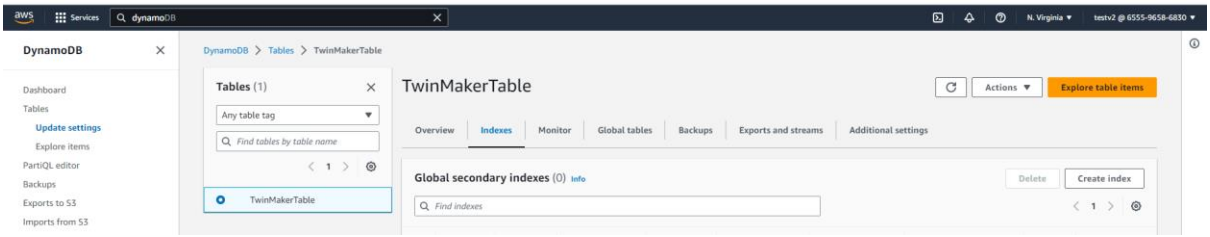


- Leave the rest of fields as it as – Click in Create

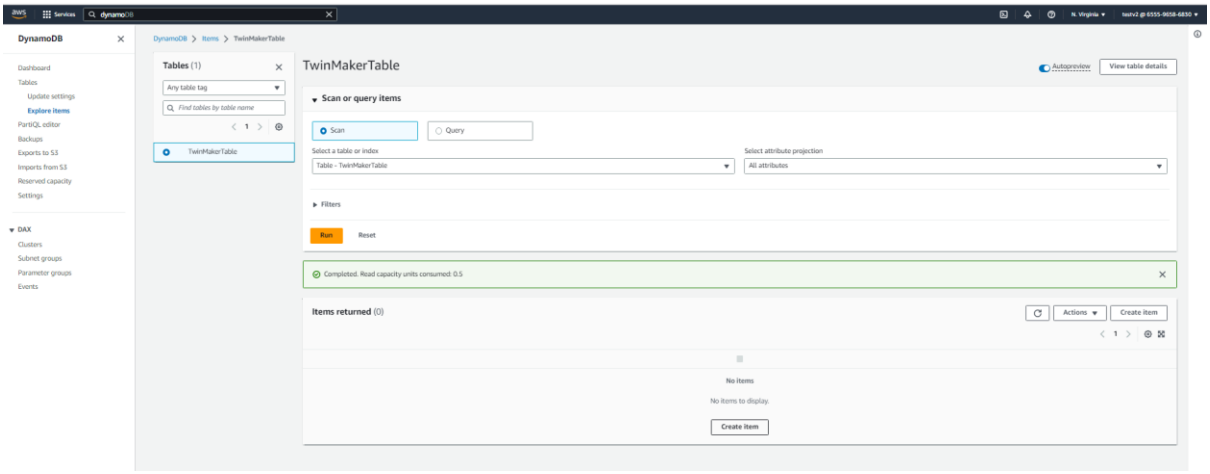


1.5.2 Creating table items

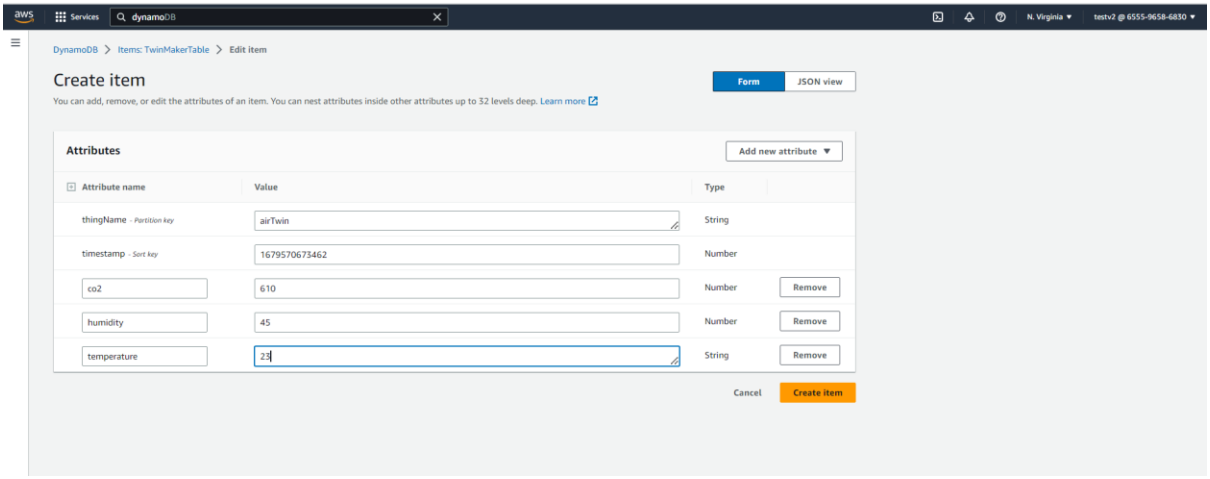
- Create 5 items in the table, corresponding to 5 measurements of a sensor named airTwin
- Click in explore table items



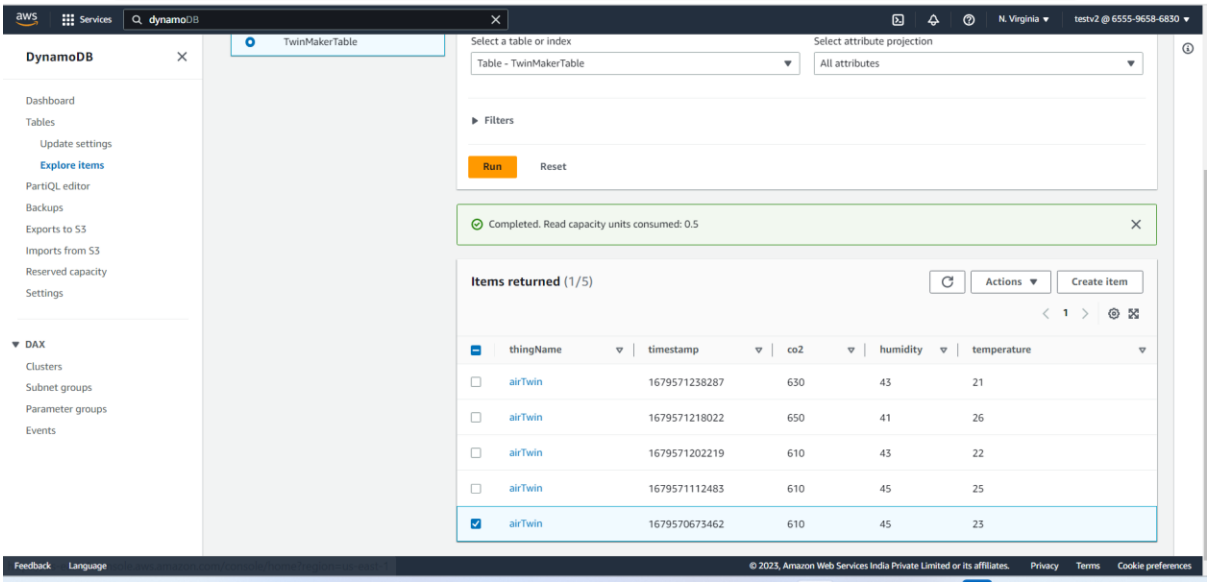
- Click on Create item



- Copy timestamp from the site mentioned below



- Enter 5 random values for the properties: temperature, humidity and co2. You can get timestamp Current Millis - Milliseconds since Unix Epoch



1.6 Creating Connectors

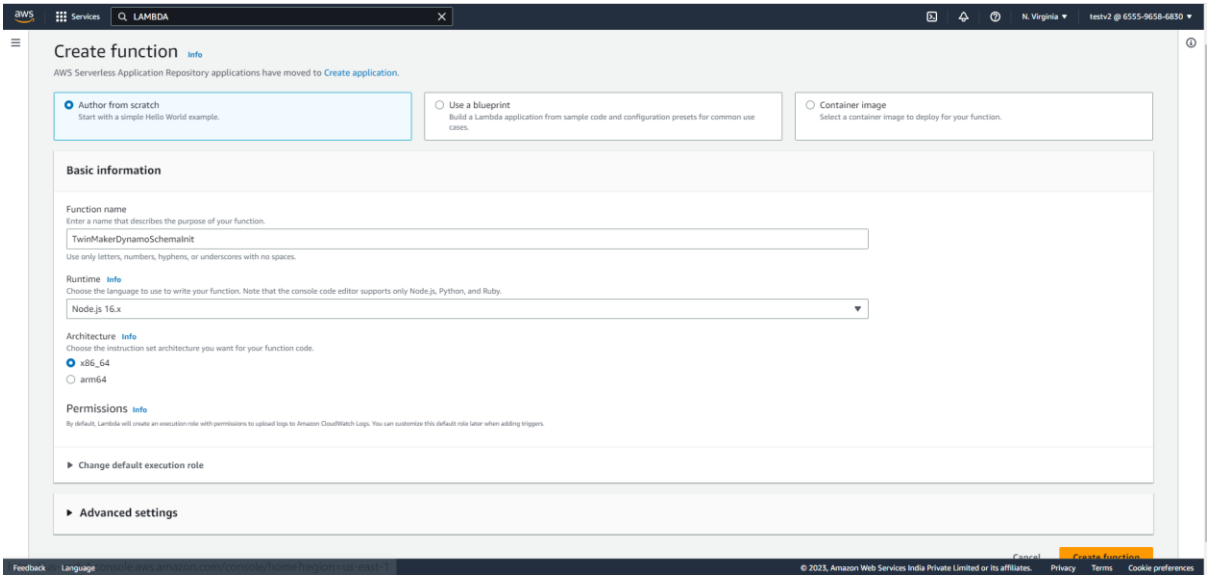
- We will creating two lambda functions which will be acting as connectors for twinmaker to read data from database such as DynamoDB.
- We will be creating two lambda functions one for identifying the schema and one for data reader

1.6.1 Creating Schema Initializer

- Go to lambda console
- Click on Create a lambda functions which will act as a schema initializer connector

- Under Basic Information

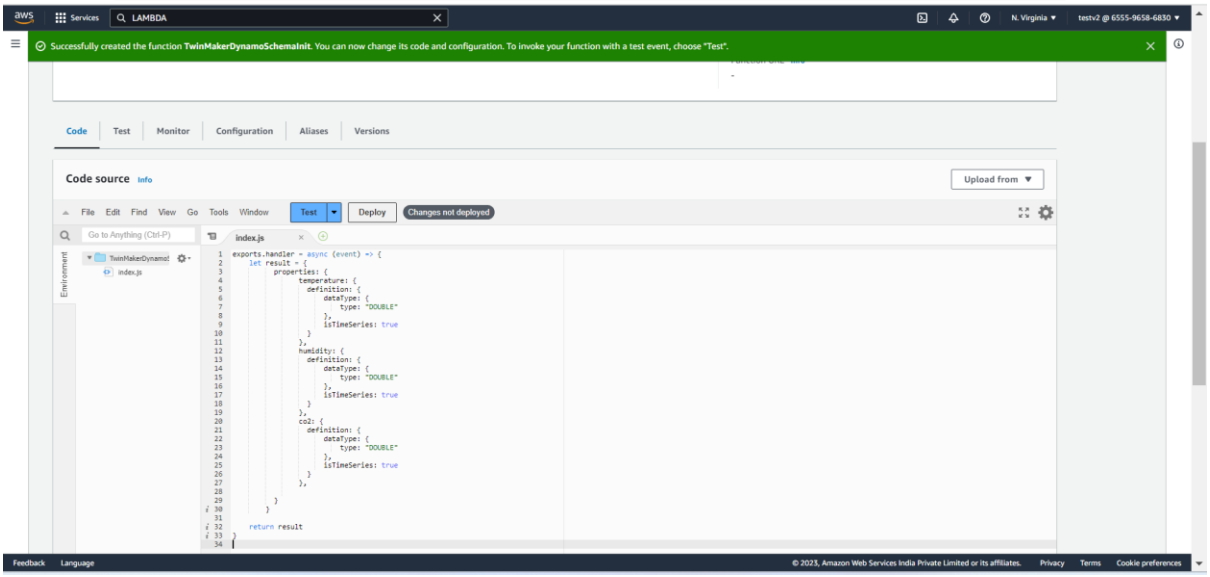
  For Function Name:- TwinMakerDynamoSchemaInit
  For Runtime:- Node.js 16x
- Choose Create Function



- Under Function code, in the inline code editor, copy/paste the following code and click "Deploy"

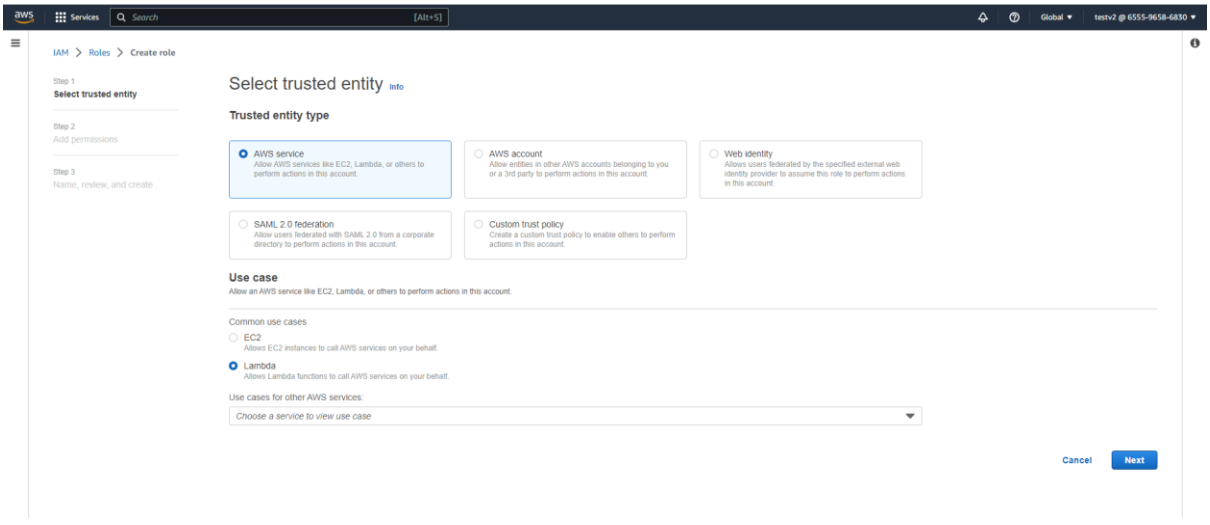
```
exports.handler = async (event) => {
  let result = {
      properties: {
          temperature: {
           definition: {
              dataType: {
                 type: "DOUBLE"
              },
              isTimeSeries: true
           }
          },
          humidity: {
           definition: {
              dataType: {
                 type: "DOUBLE"
              },
              isTimeSeries: true
           }
          },
          co2: {
           definition: {
              dataType: {
                 type: "DOUBLE"
              },
              isTimeSeries: true
           }
          },
```

```
        }
      }

   return result
}
```

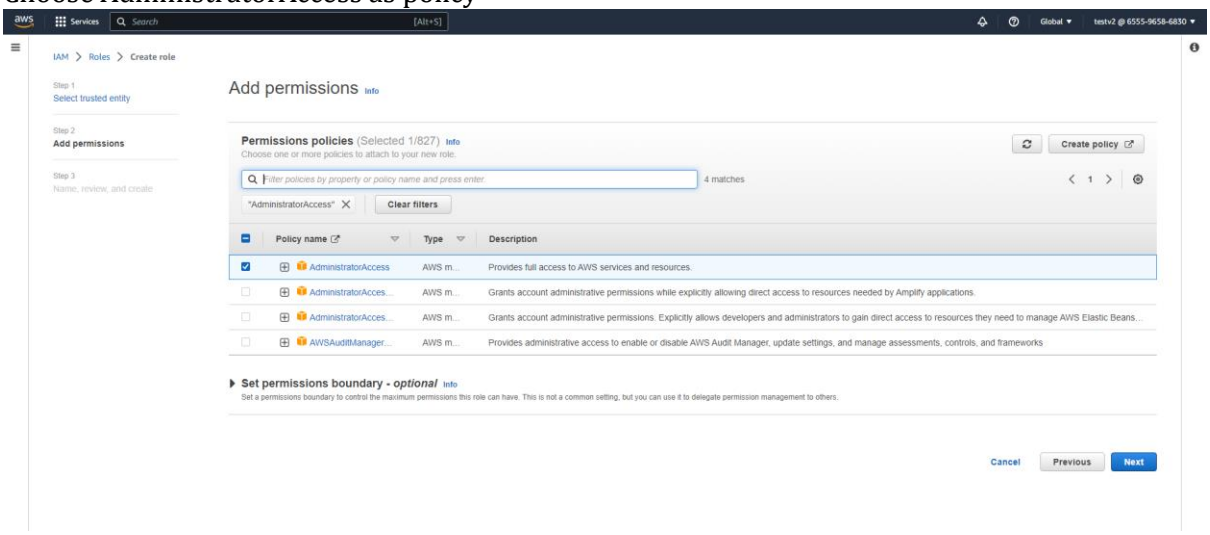


1.6.2 Creating Data Reader

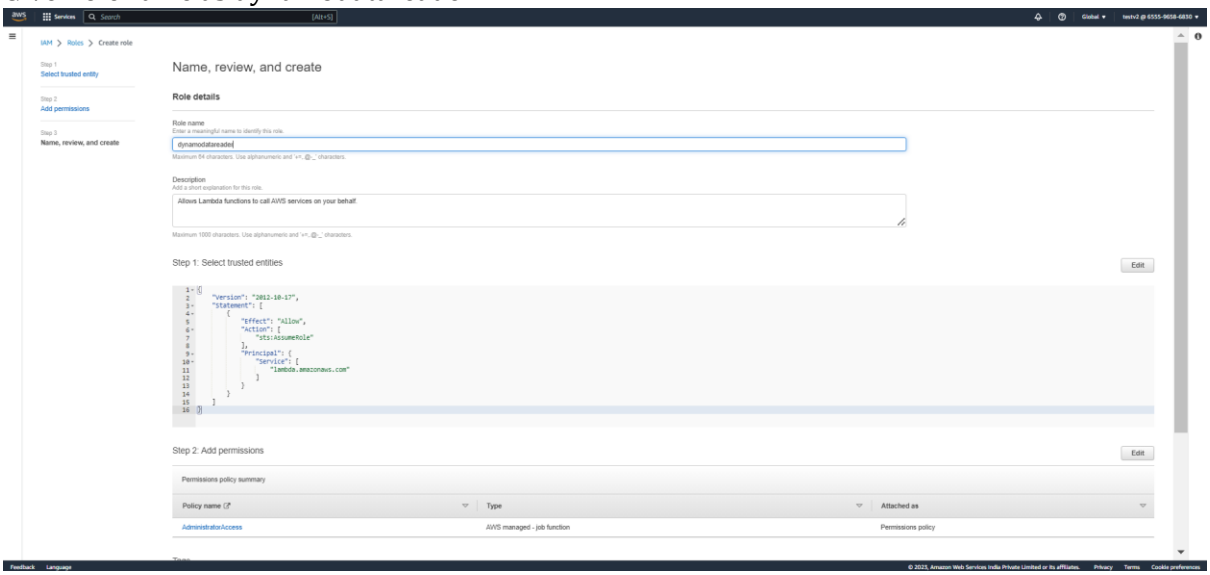
- Create a second lambda function which will act as a data reader
- Before Creating a Data Reader we have to grant the function the access to the table.
- Create a IAM Role
- Choose Lambda as a Use Case

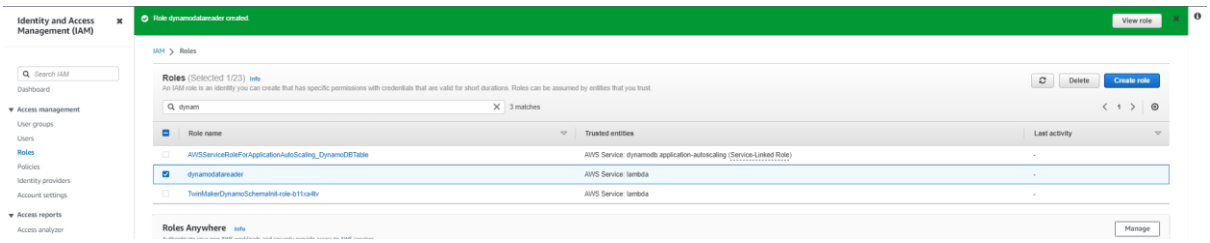


- Choose AdministratorAccess as policy



- Give RoleName as dynamodatareader

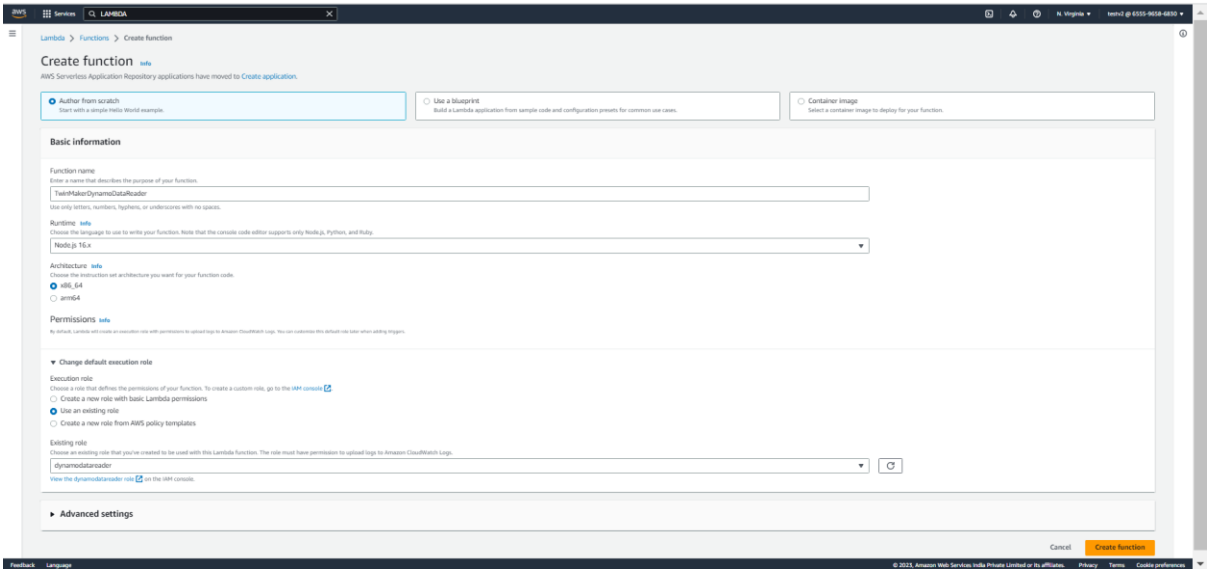




- Under Basic Information

For Function Name: TwinMakerDynamoDataReader
For Runtime:- Node.js 16x
For IAM Role :- dynamodatareader

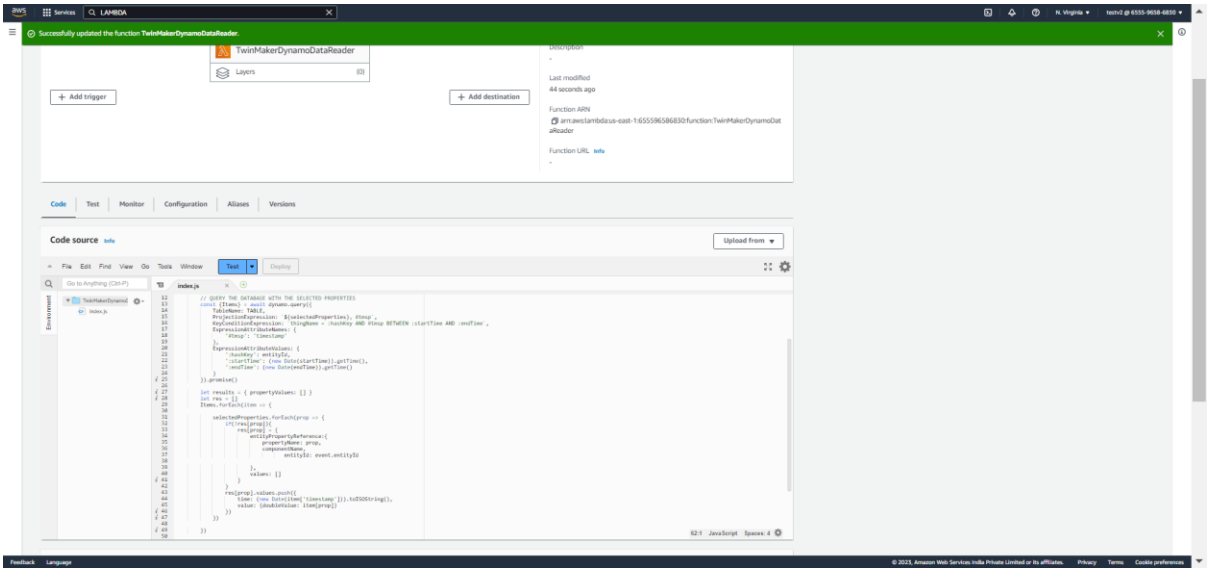
- Choose Create Function



- Under Function code, in the inline code editor, copy/paste the following code and choose Deploy – this code is actually querying the data in DynamoDB

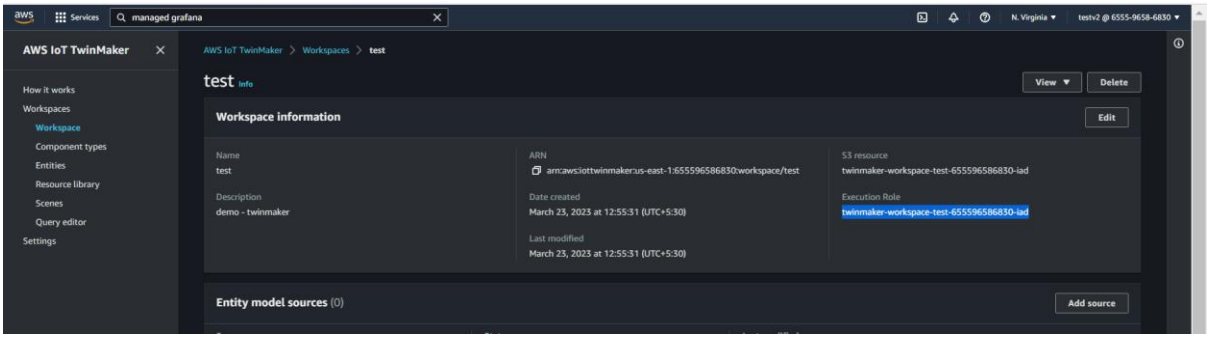
```
const TABLE = 'TwinMakerTable'
const aws = require('aws-sdk')
const dynamo = new aws.DynamoDB.DocumentClient()


exports.handler = async (event) => {
   try {

      let {workspaceId, entityId, componentName, selectedProperties, startTime,
endTime } = event


      // QUERY THE DATABASE WITH THE SELECTED PROPERTIES
      const {Items} = await dynamo.query({
         TableName: TABLE,
         ProjectionExpression: `${selectedProperties}, #tmsp`,
         KeyConditionExpression: `thingName = :hashKey AND #tmsp BETWEEN
:startTime AND :endTime`,
         ExpressionAttributeNames: {
            '#tmsp': 'timestamp'
         },
         ExpressionAttributeValues: {
            ':hashKey': entityId,
            ':startTime': (new Date(startTime)).getTime(),
            ':endTime': (new Date(endTime)).getTime()
         }
      }).promise()

      let results = { propertyValues: [] }
```

```
        let res = []
        Items.forEach(item => {

            selectedProperties.forEach(prop => {
                if(!res[prop]){
                    res[prop] = {
                        entityPropertyReference:{
                            propertyName: prop,
                            componentName,
                                entityId: event.entityId

                        },
                        values: []
                    }
                }
                res[prop].values.push({
                    time: (new Date(item['timestamp'])).toISOString(),
                    value: {doubleValue: item[prop]}
                })
            })

        })

        for (let key in res){
            results.propertyValues.push(res[key])
        }

        console.log(results)
        return results
    } catch (e) {
        console.log(e)
    }

}
```

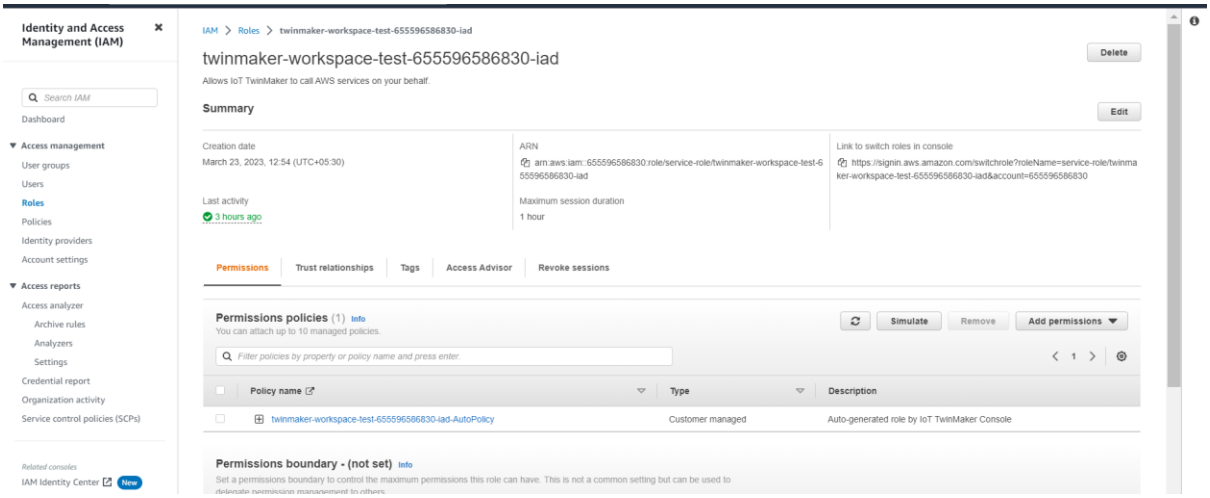


1.6.3 Connecting IOT-Twinmaker & Lambda

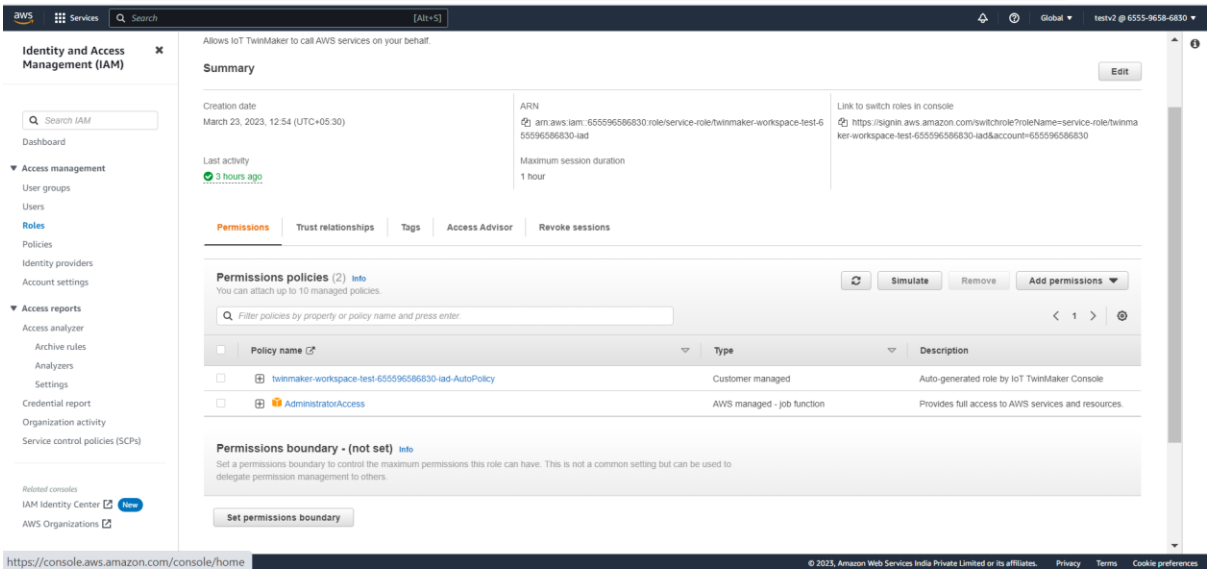
- Find the execution role of iot-workspace



- Go to IAM Console Roles and Search the above Execution Role

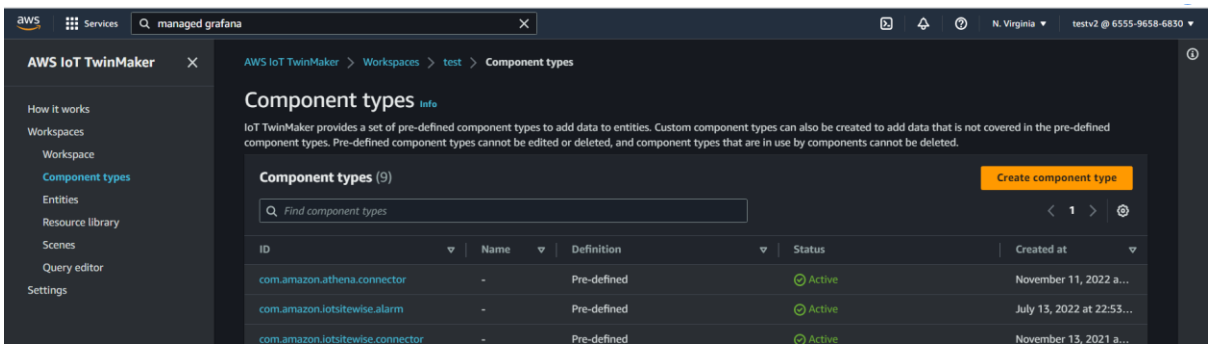


- Click on the Role and Add Permission under that attach AdministratorAccess Policy and update the role.



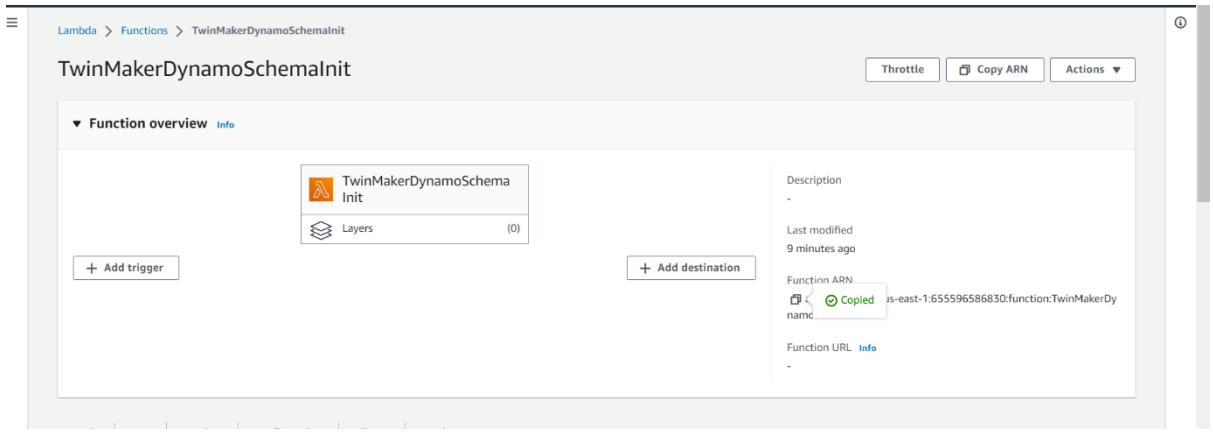
1.7 Creating Components & Entities

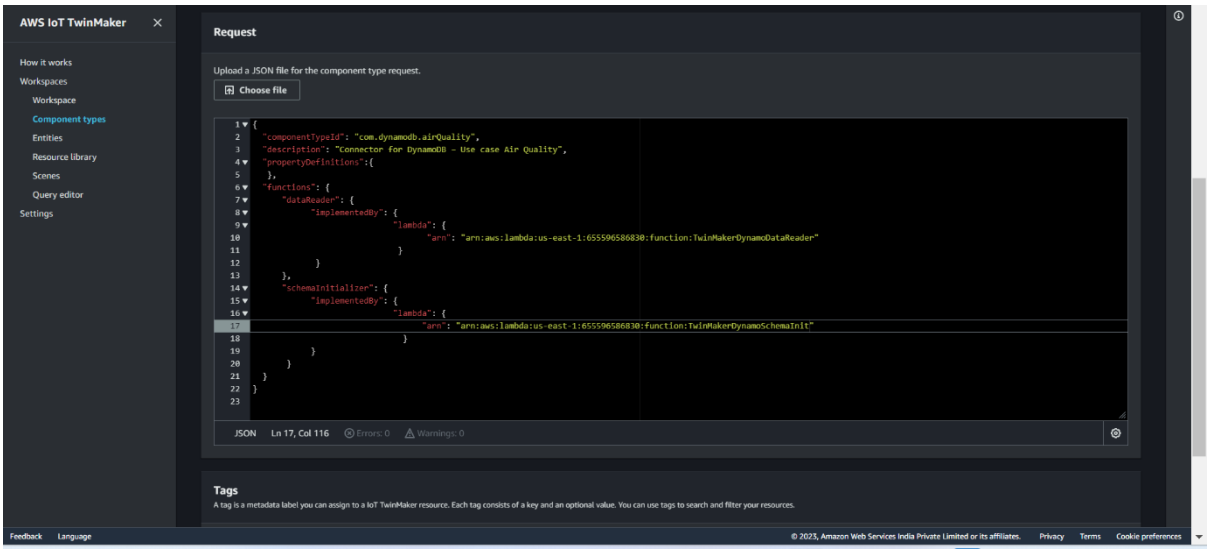
- Go IOT-Twinmaker console
- Create an IOT Twinmaker Component



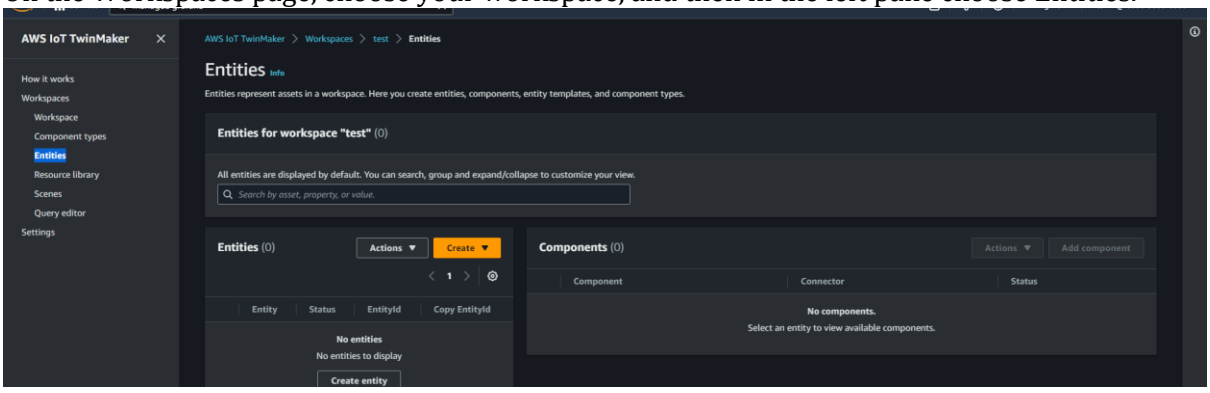
- In the workspace, choose Component types and then choose Create component type
- Copy the following JSON document in the Request section and replace the ARN of the DataReader and Schema initializer functions respectively with the ones you created before:

```
{
"componentTypeId": "com.dynamodb.airQuality",
"description": "Connector for DynamoDB – Use case Air Quality",
"propertyDefinitions":{
},
"functions": {
"dataReader": {
"implementedBy": {
"lambda": {
"arn":
"arn:aws:lambda:{{AWS_REGION}}:{{ACCOUNT_ID}}:function:TwinMakerDynamoDa
taReader"
}
}
},
"schemaInitializer": {
"implementedBy": {
"lambda": {
"arn":
"arn:aws:lambda:{{AWS_REGION}}:{{ACCOUNT_ID}}:function:TwinMakerDynamoSc
hemaInit"
}
}
}
}
}
```

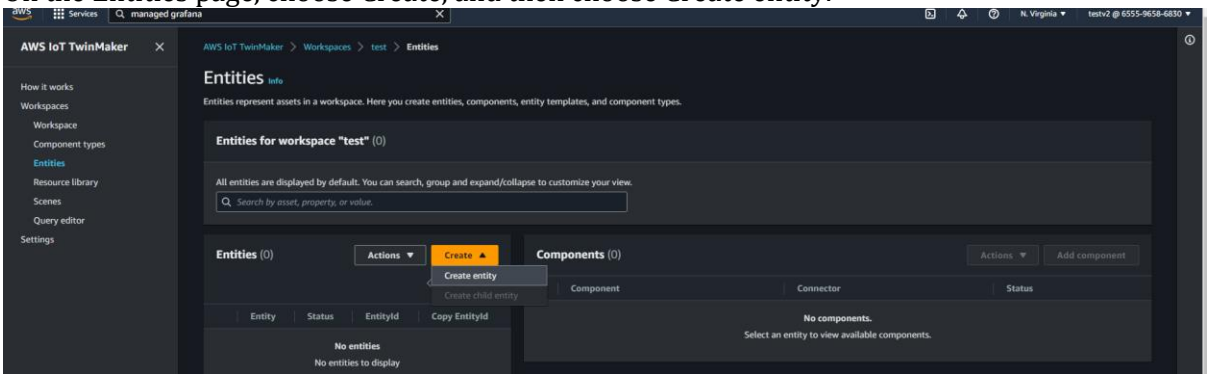




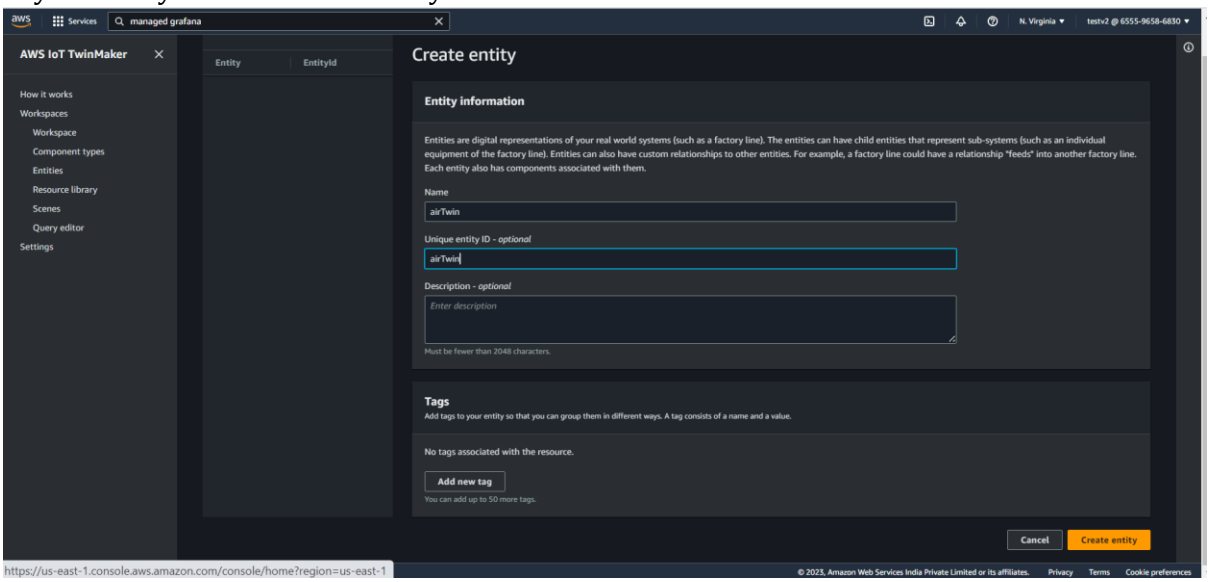
- Click on Create Component type
- On the Workspaces page, choose your workspace, and then in the left pane choose Entities.



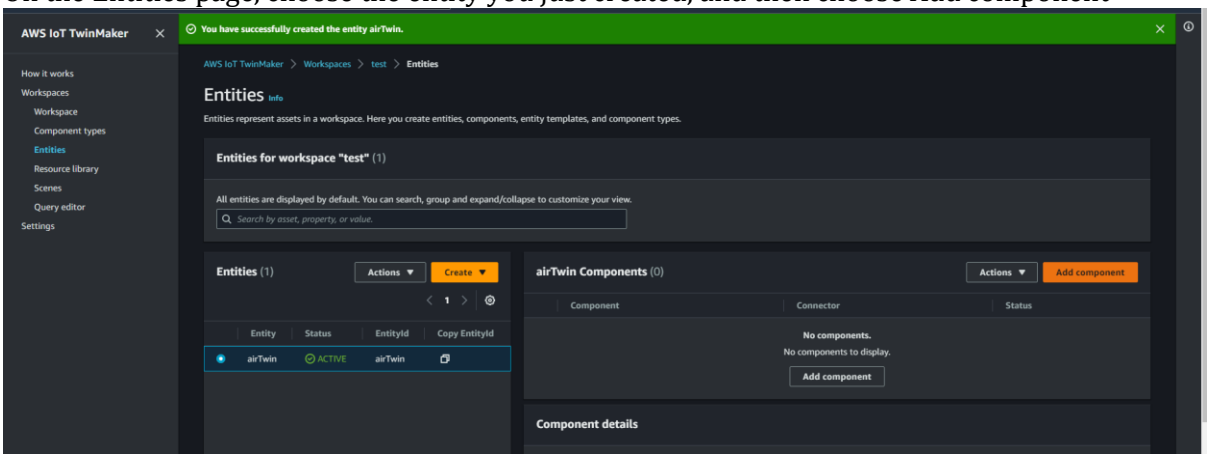
- On the Entities page, choose Create, and then choose Create entity.



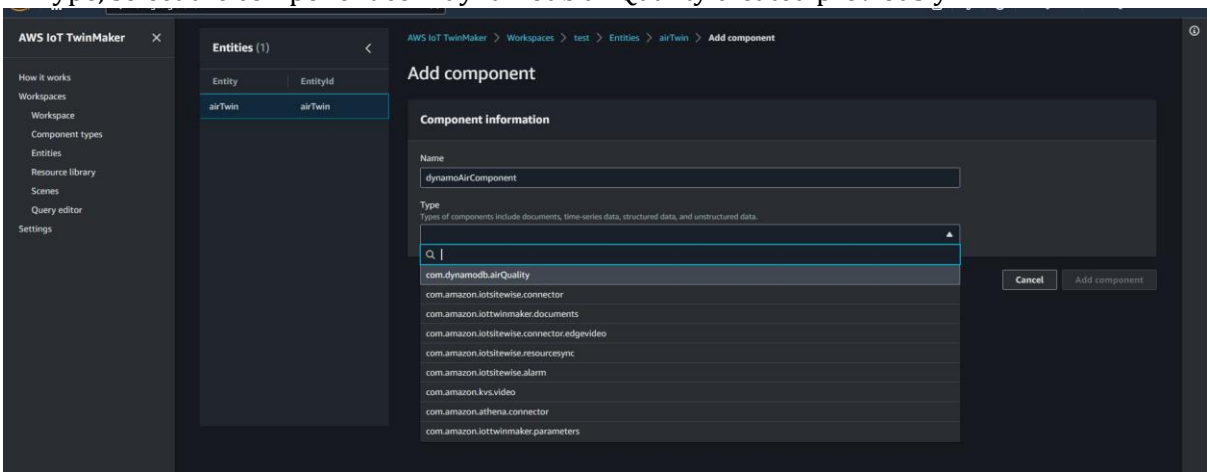
- In the Create an entity window, enter airTwin for the entity name and also for the entity ID of your entity. Choose Create entity



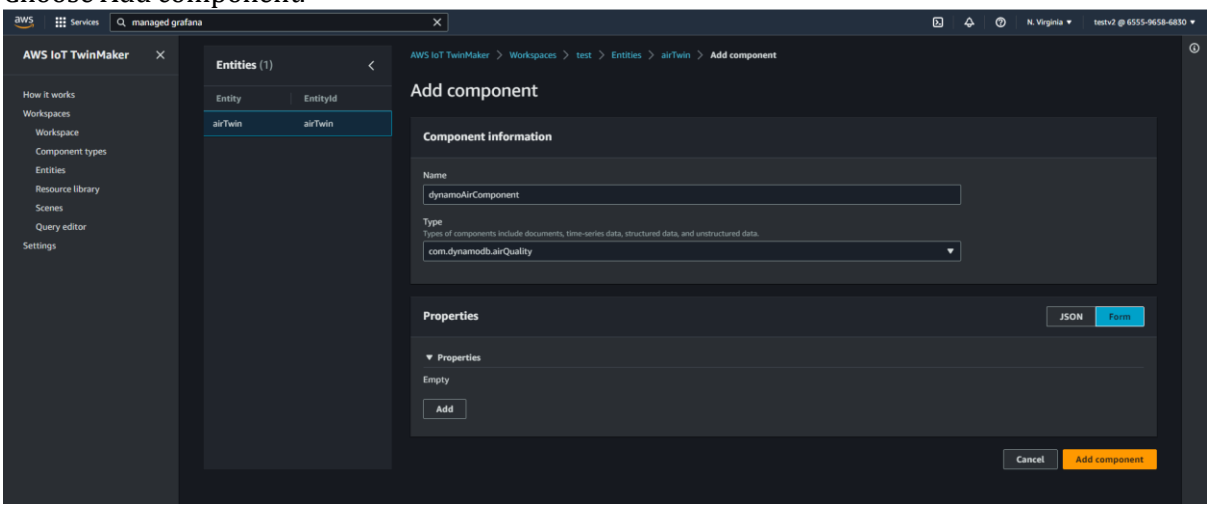
- On the Entities page, choose the entity you just created, and then choose Add component



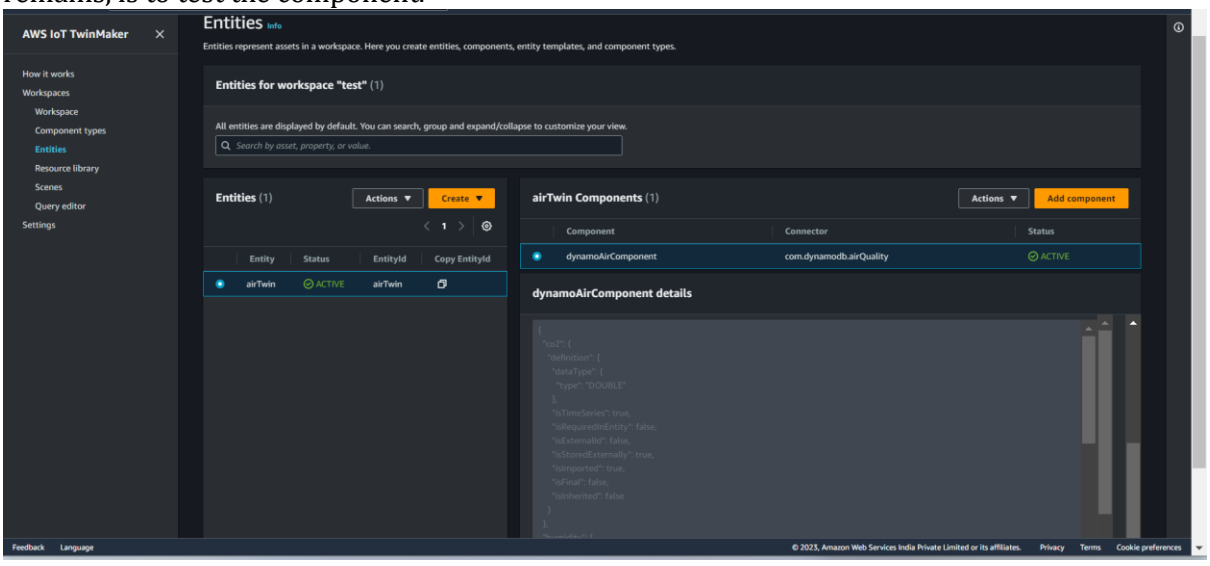
- Enter a name for the component. You can call it dynamoAirComponent
- In Type, select the component com.dynamodb.airQuality created previously



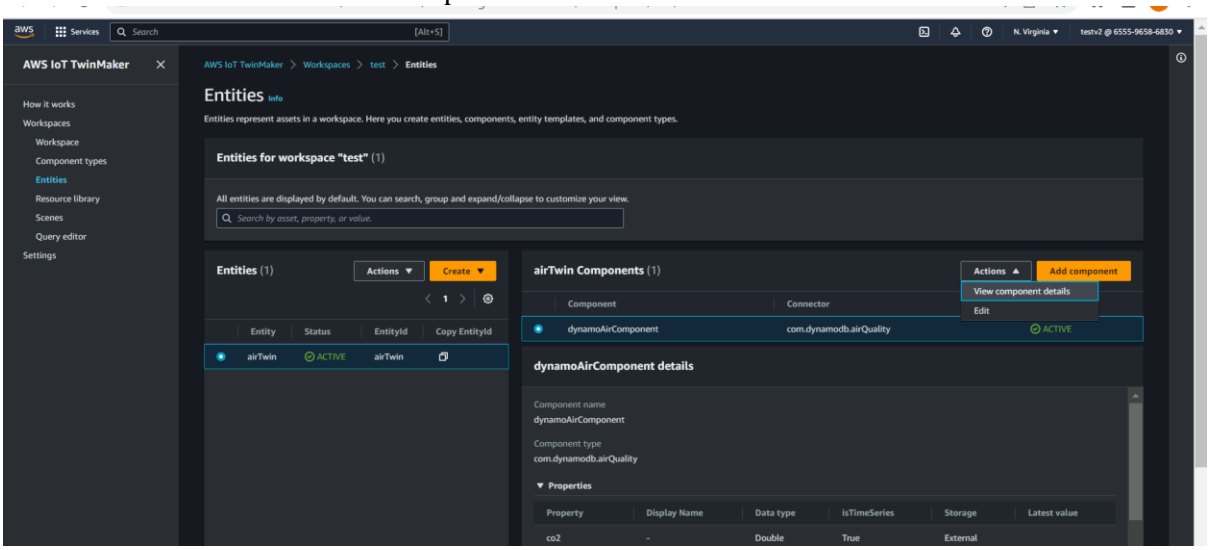
- Choose Add component.



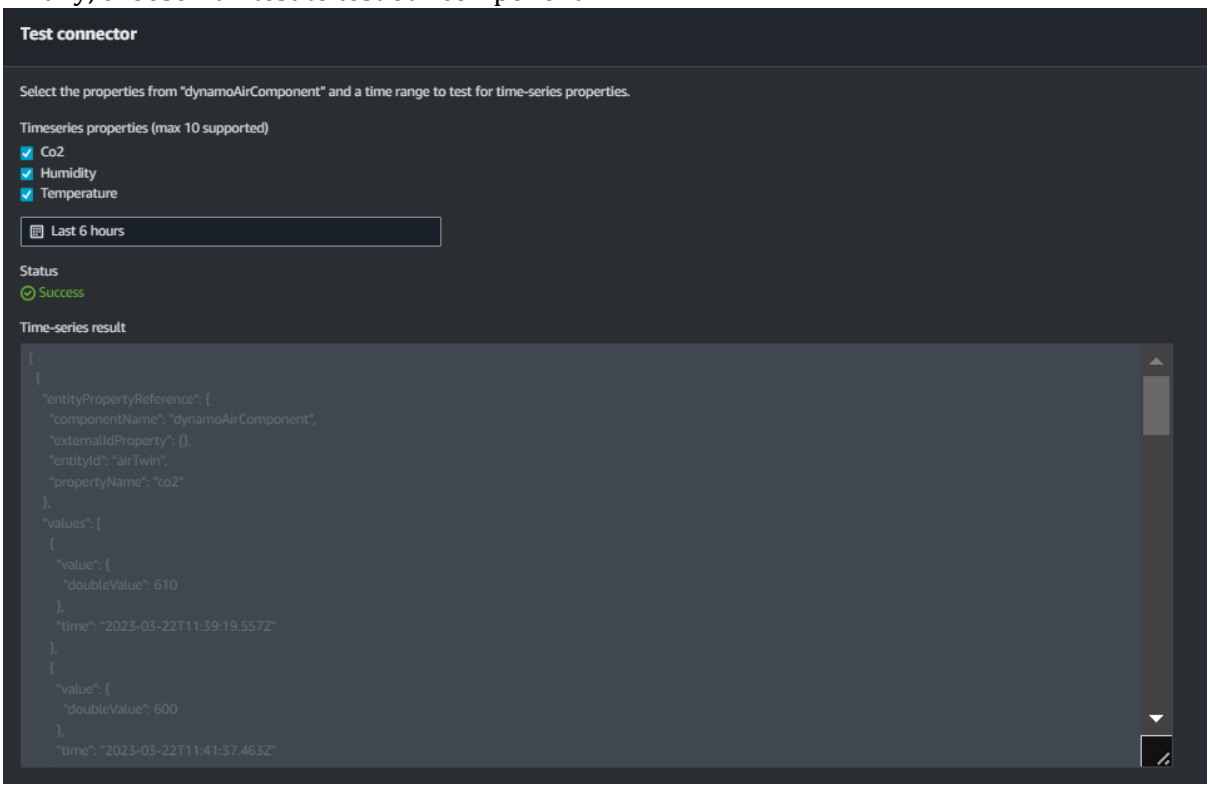
- The component is attached to the entity with the ID airTwin. Now the only step that remains, is to test the component.



- On the Entities page, choose the entity airTwin, and then select the component com.dynamodb.airQuality
- Then choose Actions and View component details.

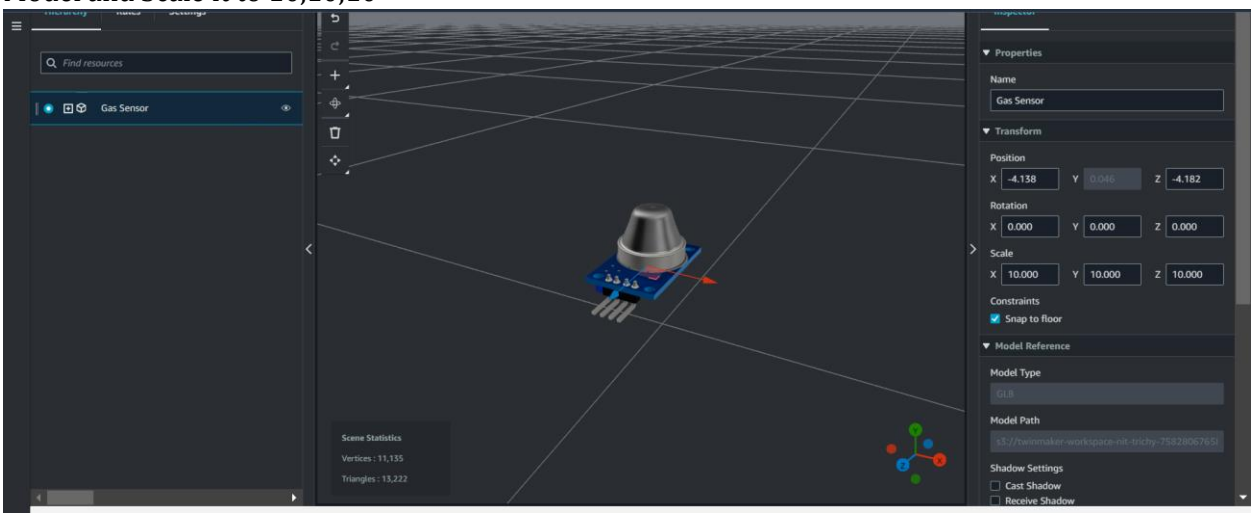


- In the tab Test, select the properties you want to retrieve and a time range. Make sure that the time range selected includes the timestamp of the measurements.
- Finally, choose Run test to test our component.

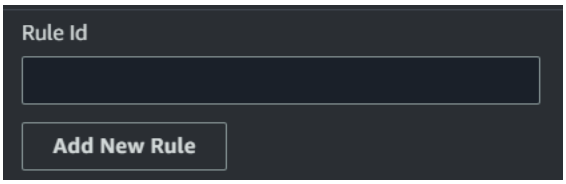


1.8 Creating 3D Scene

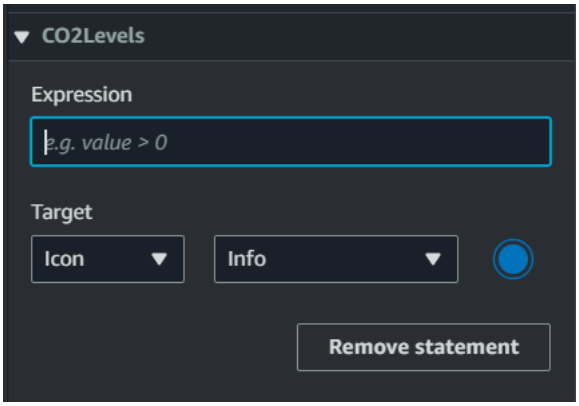
- Go to Resource Library and add a 3d Model. You can download the 3d model from github provided in the pre-requisite.
- Create a 3d Scene with Name AirSensor. Open the Scene and change the environment to Neutral
- Add 3d Model and select the model you just uploaded. Click on the 3d plane to place your 3d Model and Scale it to 10,10,10



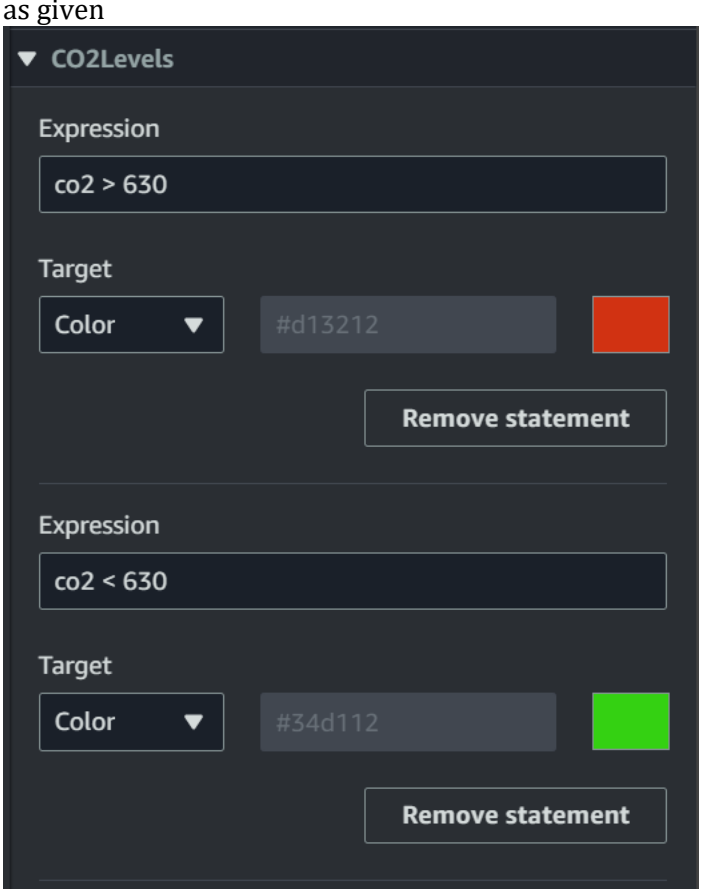
- Go to Rules and give rule id as “CO2Levels” and click on “Add New Rule”



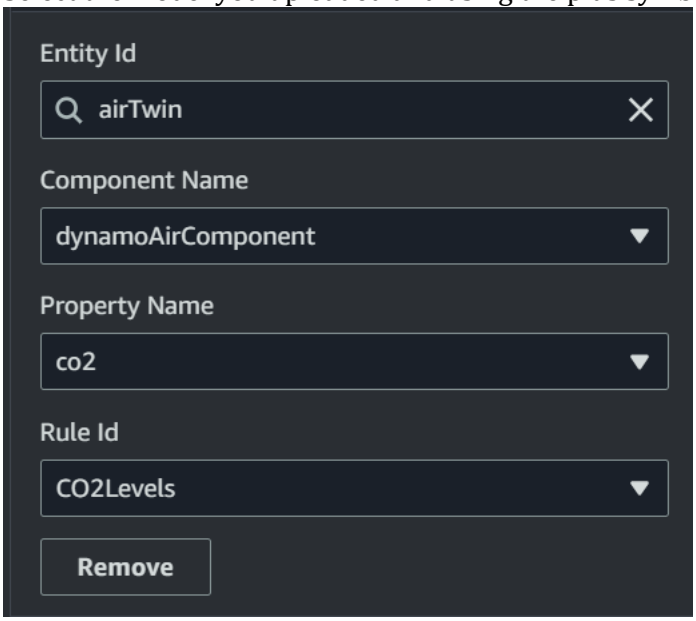
- Using Add New Statement change the “Target” as Colour



- Using Add New Statement apply the rules as following set Target as colour and Expression as given

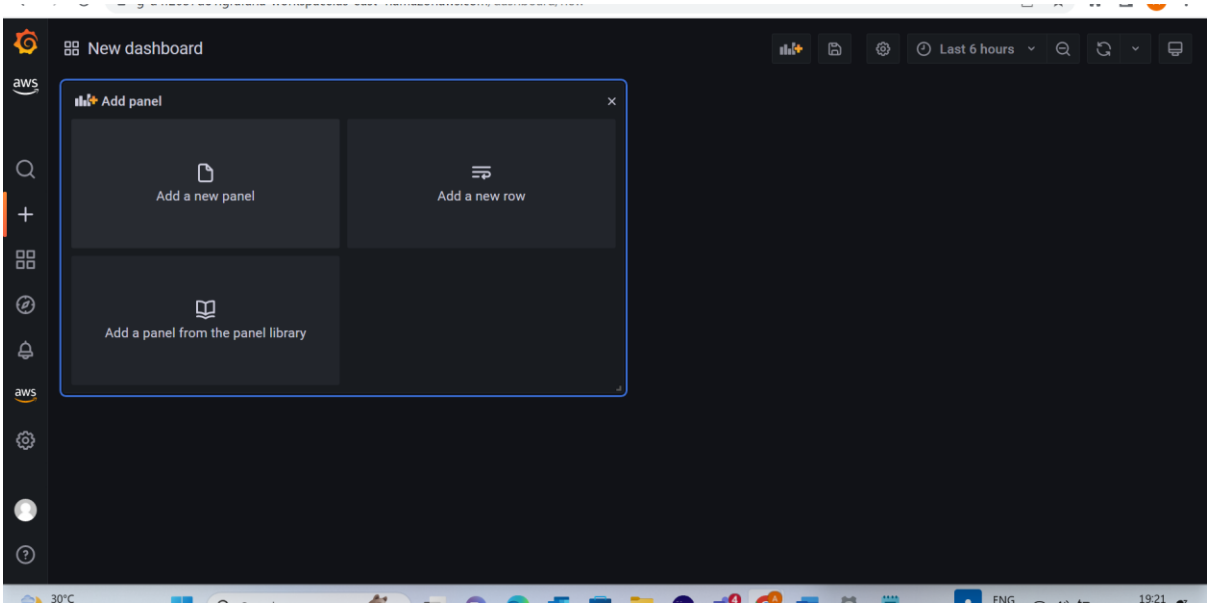


- Select the model you uploaded and using the plus symbol click on Add Model Shader.

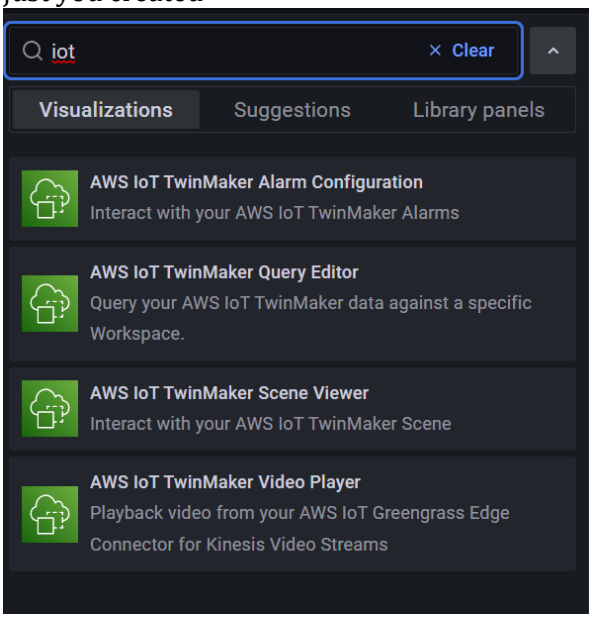


1.9 Working Dashboard

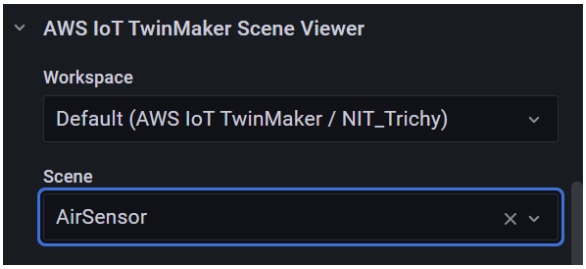
- Go to Grafana Dashboard using the plus symbol on left create a new dashboard



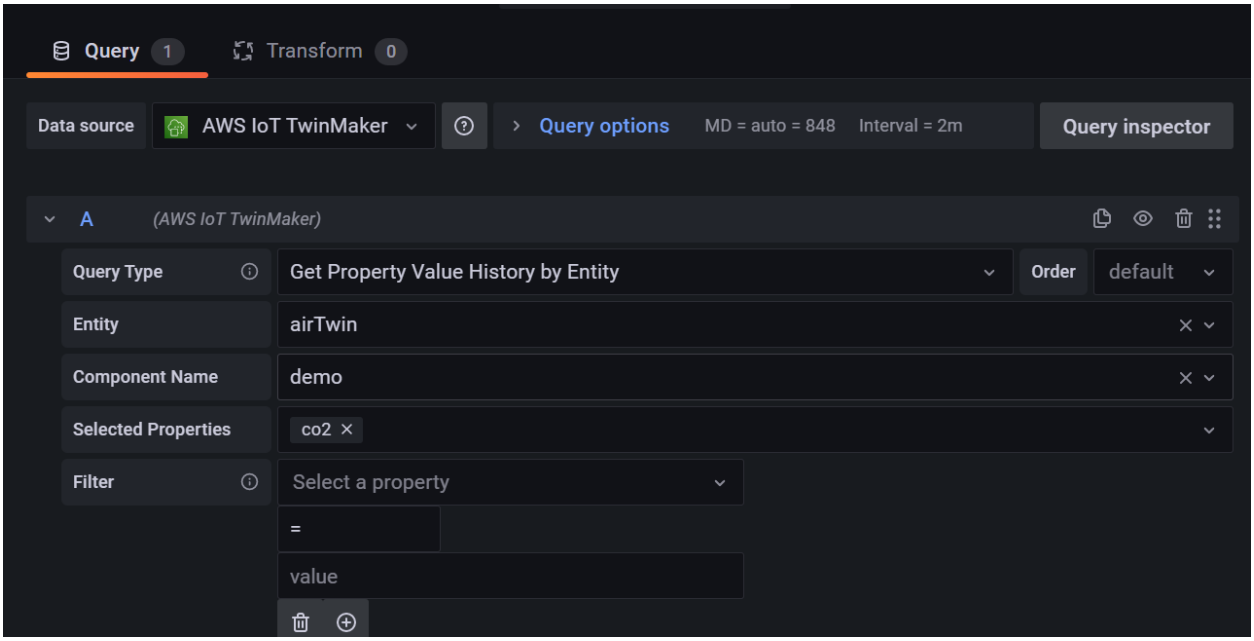
- Add a New Panel use Aws IoT Twin maker Scene Viewer under Scene select the scene which just you created



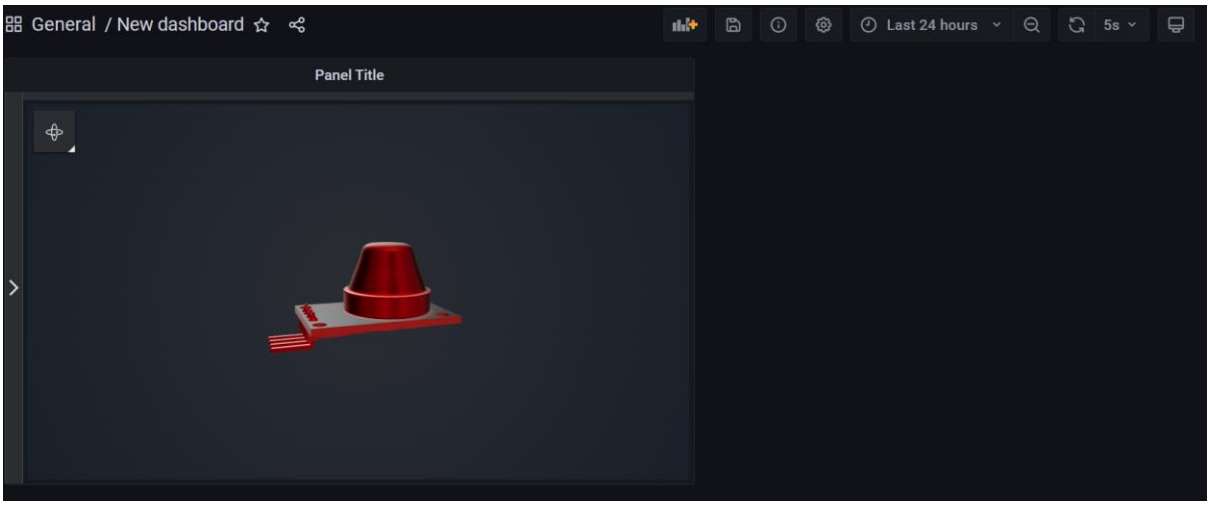
- In the Scene select the scene we created “AirSensor”



- Under the query inspector select “Query Type” as “Get Property Value History by Entity”.
- Select the entity as “AirTwin”, component as “dynamoaircomponent” and selected properties as co2

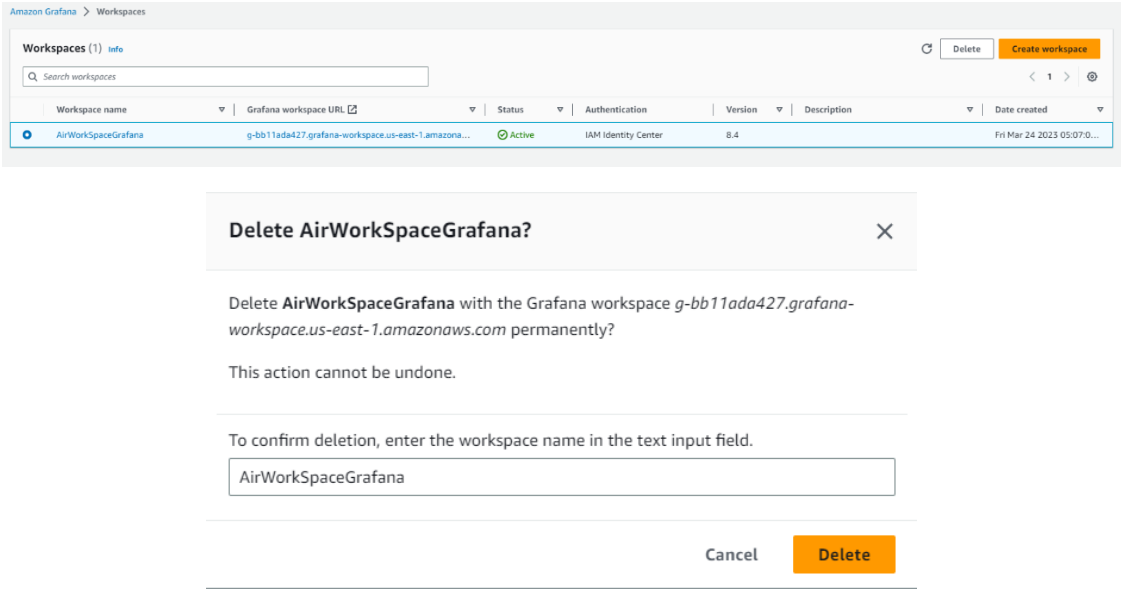


- Click on apply and save the dashboard

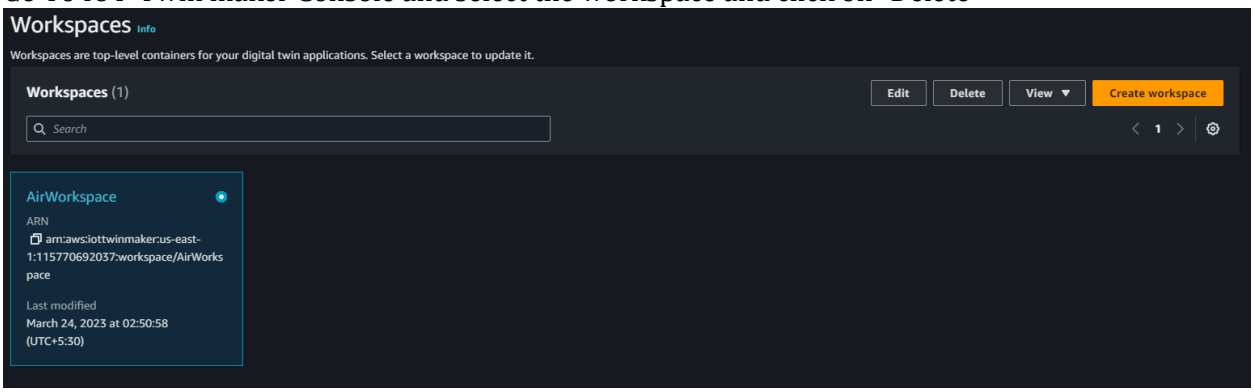


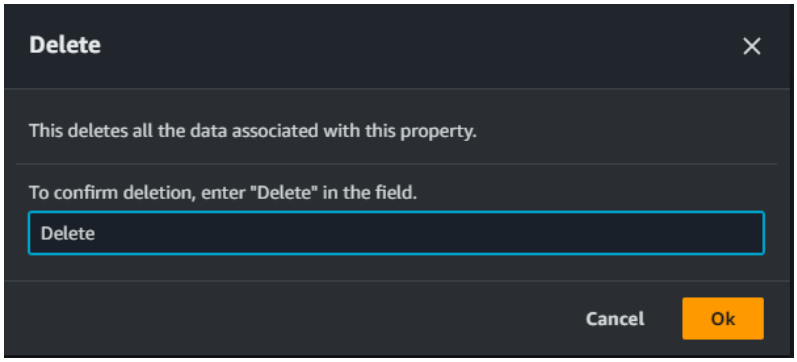
1.10 Deleting Resources

- Go To Amazon Grafana and select the workspace and then click on “Delete”



- Go To IOT-Twin maker Console and select the workspace and click on “Delete”





- Go to DynamoDB Console and select the table and click on “Delete” and type “confirm” in the popped message