Q3,Q4

Create iam user,then group,give user and grp cloud9environment permission

Open the console log-in link in incognito tab.

Go to cloud9 save a random file. Share ->invite user u made ->now open environment link in incognito tab.

Collab in chat box

Q5

YOU ARE FUCKED

Q6

YOU ARE FUCKED

Q7

YOU ARE FUCKED YET AGAIN

Q8,9,10

Set path for sonarqube and sonar path “bin->windows->lib” is the path to be given in env variable

Sonarscanner path is given till bin , path to be given in environment var and “PATH” ->new

Go to c->sonarqube->bin->windows->start sonar

Local9000

Create new project->give name,key is generated and generate token(command is generated with token)

Now go to sonarscanner->conf->sonarscanner-> open and give sonar.projectKey=TypeScript sonar.projectName=TypeScript sonar.projectVersion=1.0 sonar.sources=C:\sonar-scanner-5.0.1.3006-windows\conf

Open sonarscanner->conf(where the code is) open terminal and paste token command

View op on dashboard

Q11,12,13

-> Search Lambda -> Create Function -> Use a blueprint -> (Blueprint name) Hello world Python 3.7

(Execution role) -> Create a new role from AWS policy templates -> role name -> Create function

Test -> Configure test event -> Event name -> Edit event JSON

Q14

import json

import boto3

s3=boto3.client('s3')

def lambda\_handler(event,context):

bucket="q14bucket"

dataToUpload = {}

dataToUpload['PID'] = '211121'

dataToUpload['DEPT'] = 'INFT'

dataToUpload['NAME'] = 'Brijraaj'

dataToUpload['FILE'] = 'brij'

fileName = 'brij' + '.json'

uploadByteStream= bytes(json.dumps(dataToUpload).encode('UTF-8'))

s3.put\_object(Bucket=bucket,Key=fileName,Body=uploadByteStream)

print('an object has been added')

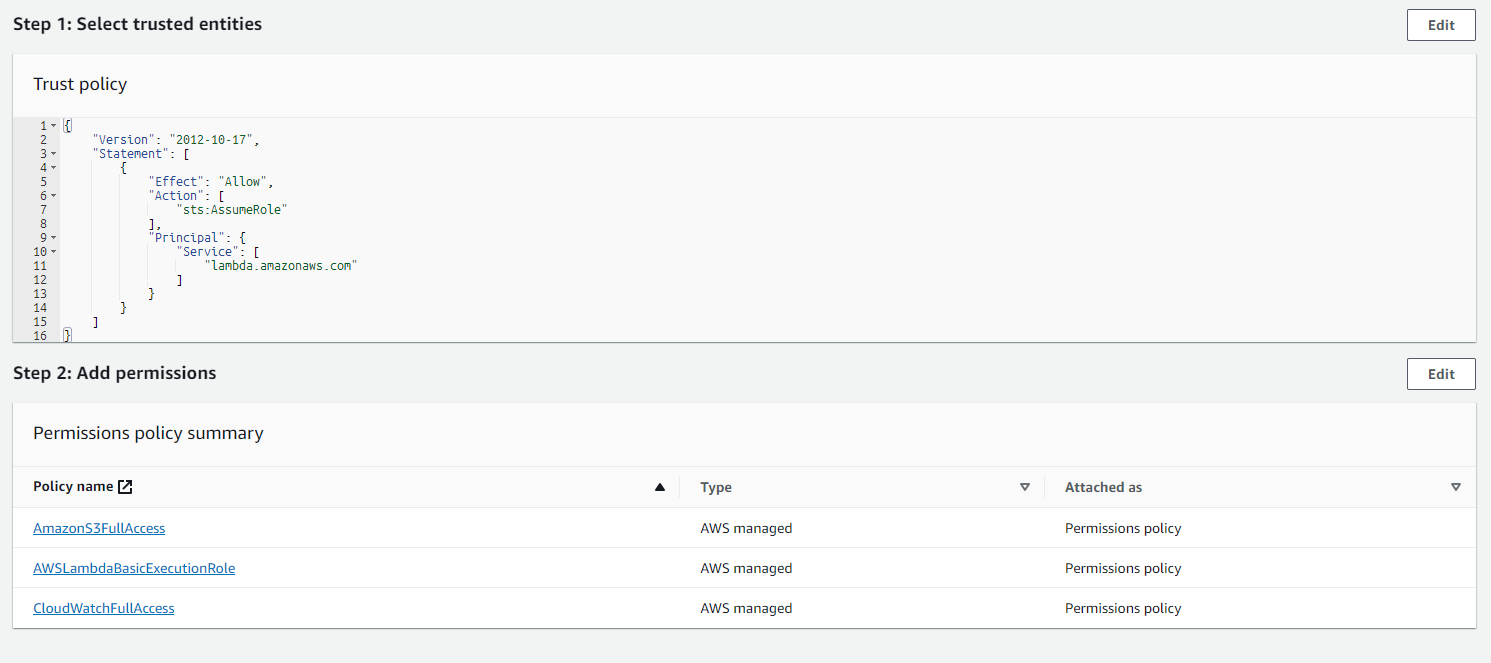
Search IAM -> Roles -> Create role -> (Usecase) Lambda -> Next

**Permissions policies**

[**CloudWatchFullAccess**](https://us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/policies/details/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FCloudWatchFullAccess)

[AWSLambdaBasicExecutionRole](https://us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/policies/details/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2Fservice-role%2FAWSLambdaBasicExecutionRole)

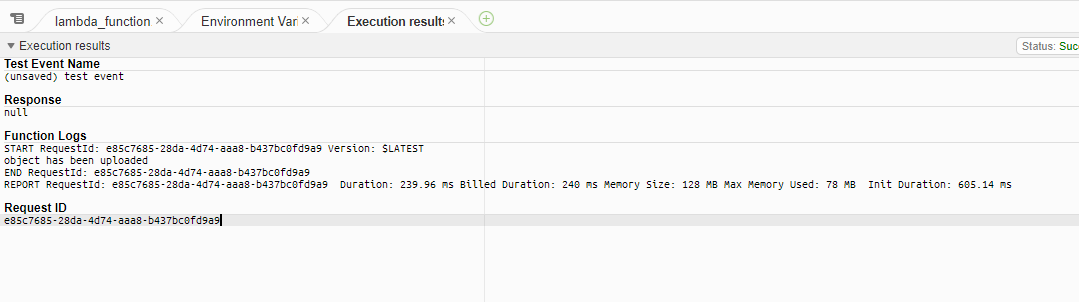
[AmazonS3FullAccess](https://us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/policies/details/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAmazonS3FullAccess)



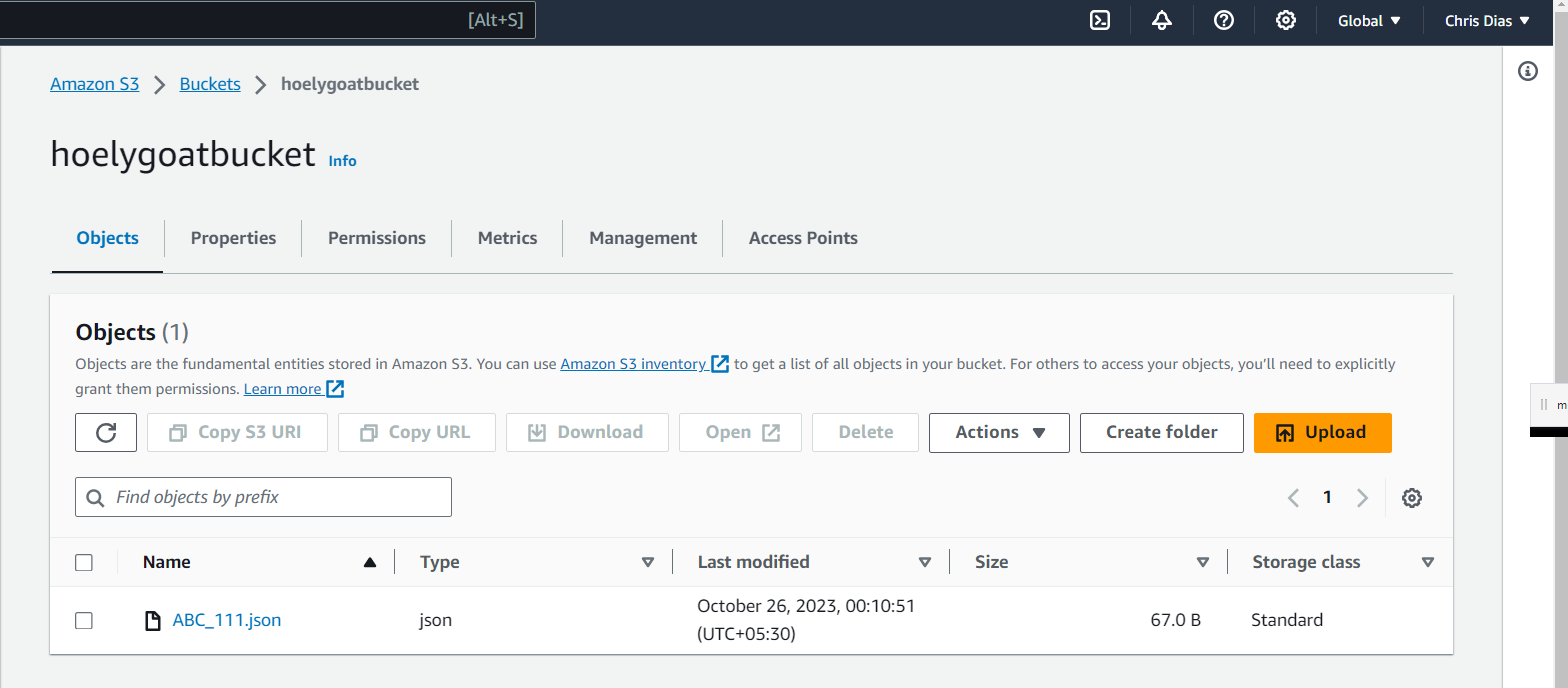
->Enter Role Name -> Create Role

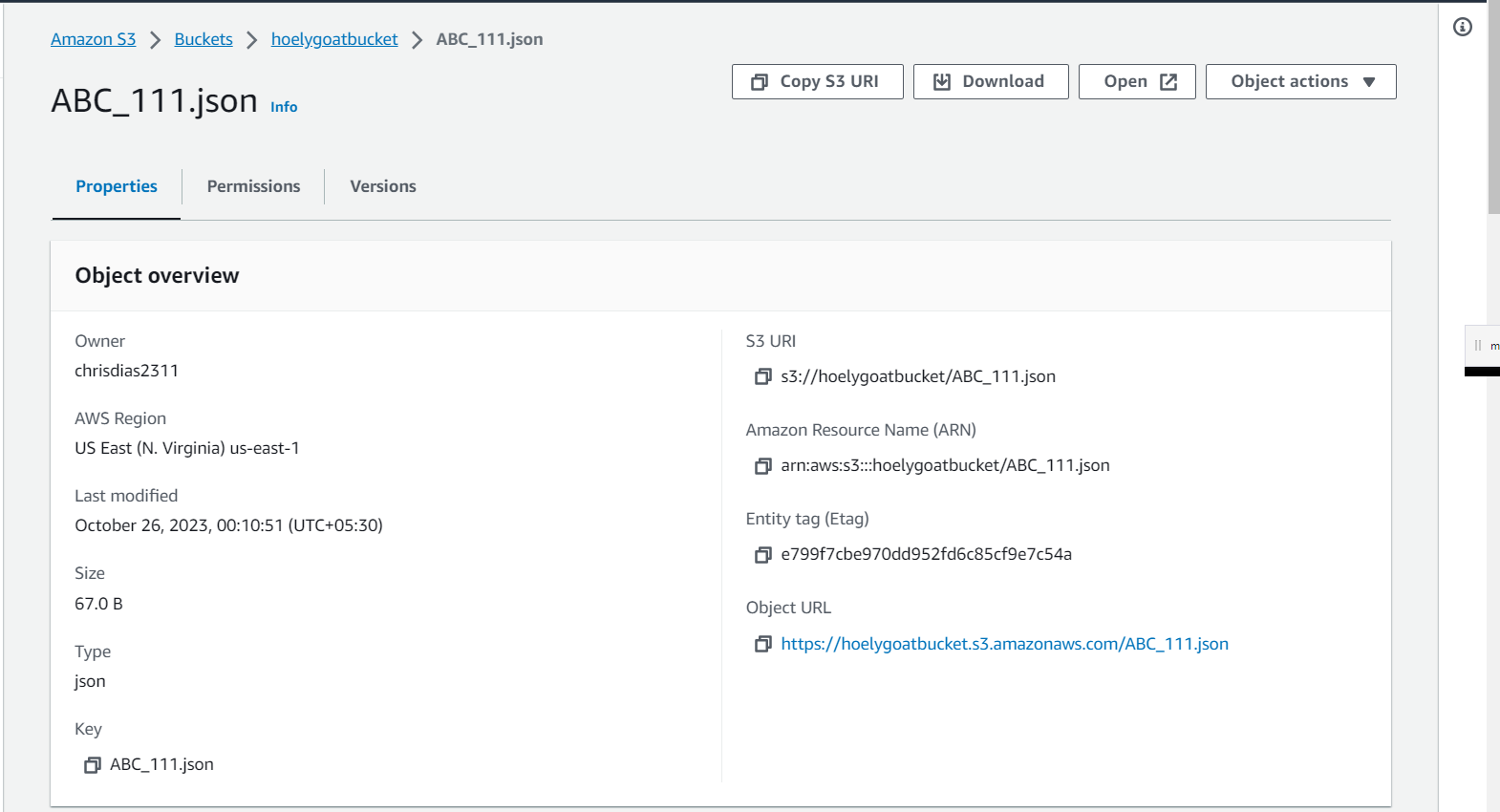
->Search S3 -> Create Bucket -> Enter Bucket name -> Create

->Search Lambda -> Create function -> Enter name -> Change default execution role -> Use an existing role -> role1->Create function

Click on Deploy -> click test -> Invoke  


Search S3 ->





Q15

Search Lambda -> Create Function -> Use a blueprint -> (Blueprint name) Hello world Python 3.7

(Execution role) -> Create a new role from AWS policy templates -> role name -> Create function

Test -> Configure test event -> Event name -> Edit event JSON

Use invoke instead of run/save. Next go to monitor for visualization

Q16

Search Lambda -> Create Function -> Use a blueprint -> (Blueprint name) Hello world Python 3.7

(Execution role) -> Create a new role from AWS policy templates -> role name -> Create function

Test -> Configure test event -> Event name -> Edit event JSON

Index.mjs is the automatic code,change it,invoke it and test.op will be generated in logs

Q17

Display present working directory in cloudshell- pwd

Q18

Q19

Easy

Q20

Same as 14

Q21

Same as q3

Q22