Cloud Development, 2021 fall.
Final

Report

made by Zhuman Rakhat

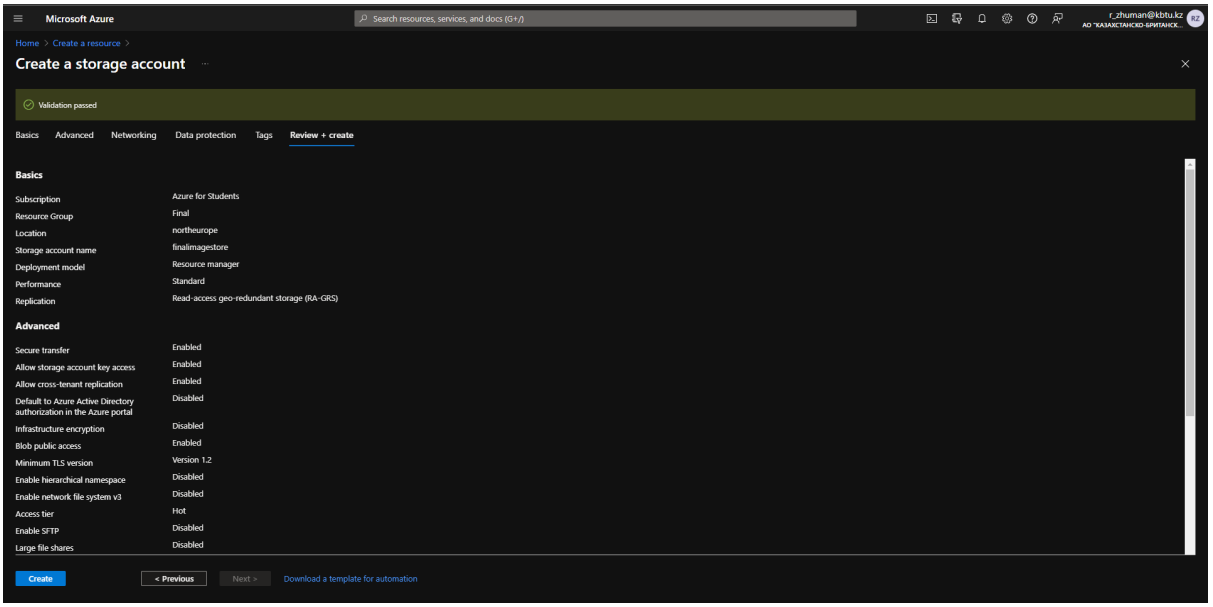
Almaty 2021

Tasks

- [x] ~~A console application, which lets a user to save files (images) to an Azure Blob Storage~~
- [x] ~~Azure Storage Account, having a blob container, where files (images) saved from the console application will be stored.~~
- [x] ~~Function that will track the container in the storage account and save smaller versions of newly appeared files (images) to another container.~~
- [x] ~~Azure Cosmos DB, containing information about newly uploaded images (it has to be a Database named ImagesDB with a container ImagesInfo, each item has to be a JSON-document, containing information about the image id, name and path)~~
- [x] ~~Logic App that tracks a container with thumbnail images and sends an email notification when a new image appears. Additionally, this app has to add documents in the ImagesDB with information about each added image.~~

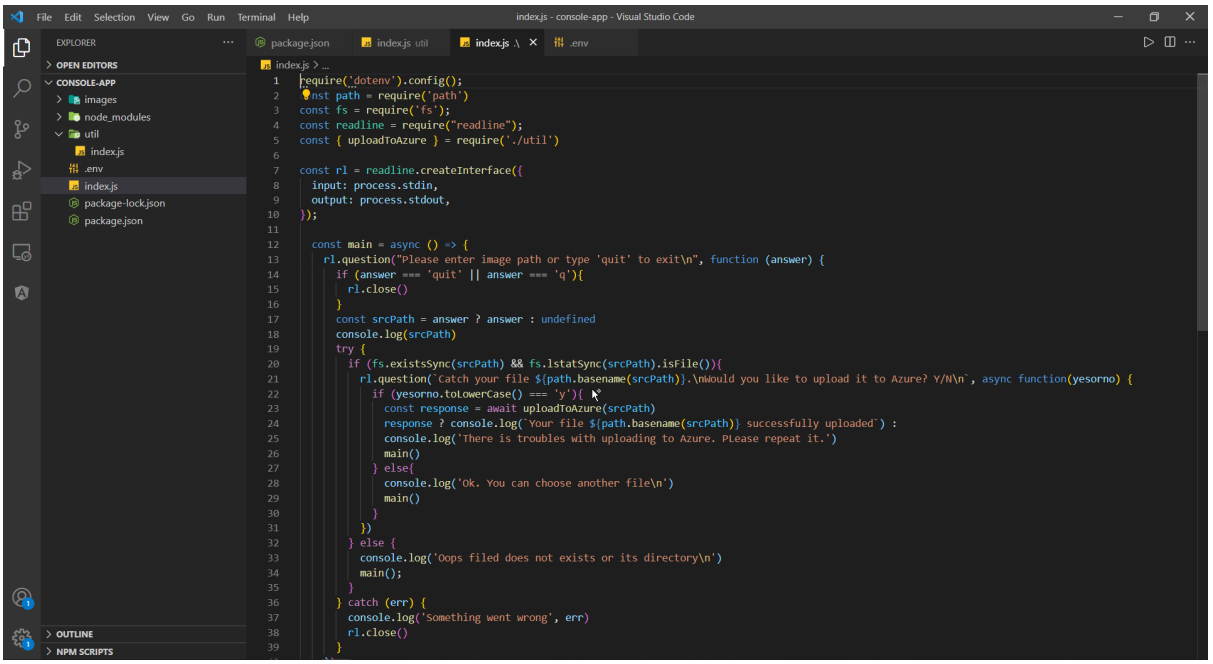
Report

Create Storage Account

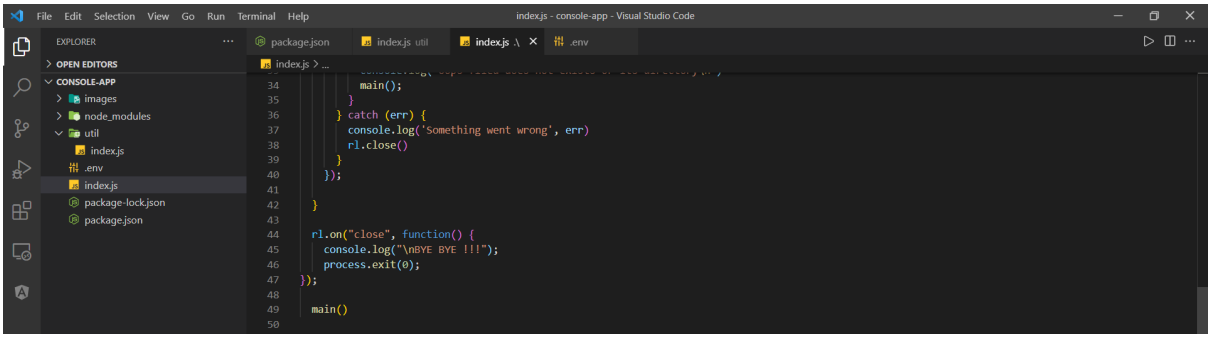


Get credentials to connect

Create console app (there is difficult to create console app in Node JS, unless Python or C#, please be kinder)

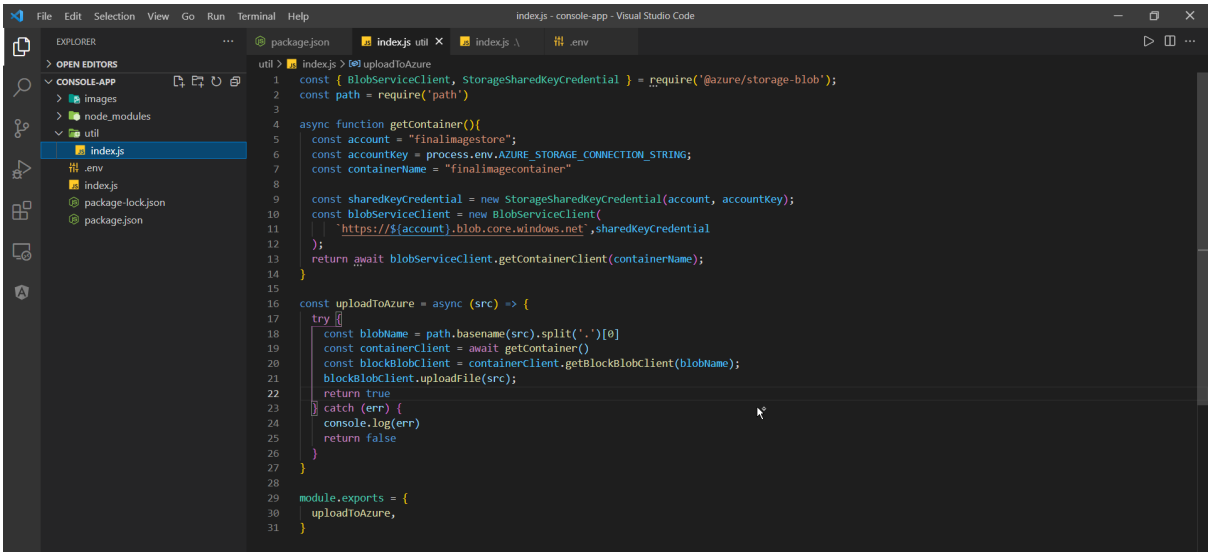


```
require('dotenv').config();
const path = require('path')
const fs = require('fs');
const readline = require("readline");
const { uploadToAzure } = require('./util')

const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout,
});

  const main = async () => {
    rl.question("Please enter image path or type 'quit' to exit\n", function (answer) {
      if (answer === 'quit' || answer === 'q'){
        rl.close()
      }
      const srcPath = answer ? answer : undefined
      console.log(srcPath)
      try {
        if (fs.existsSync(srcPath) && fs.lstatSync(srcPath).isFile()){
          rl.question(`Catch your file ${path.basename(srcPath)}.\nWould you like to upload it to Azure? Y/N\n`, async function(yesorno) {
            if (yesorno.toLowerCase() === 'y'){
              const response = await uploadToAzure(srcPath)
              response ? console.log(`Your file ${path.basename(srcPath)} successfully uploaded`) :
              console.log('There is troubles with uploading to Azure. PLease repeat it.')
              main()
            } else{
              console.log('Ok. You can choose another file\n')
              main()
            }
          })
        } else {
          console.log('Oops filed does not exists or its directory\n')
          main();
        }
      } catch (err) {
        console.log('Something went wrong', err)
        rl.close()
      }
```



```
      main();
    }
  } catch (err) {
    console.log('Something went wrong', err)
    rl.close()
  }
 });

}

rl.on("close", function() {
  console.log("\nBYE BYE !!!");
  process.exit(0);
});

  main()
```

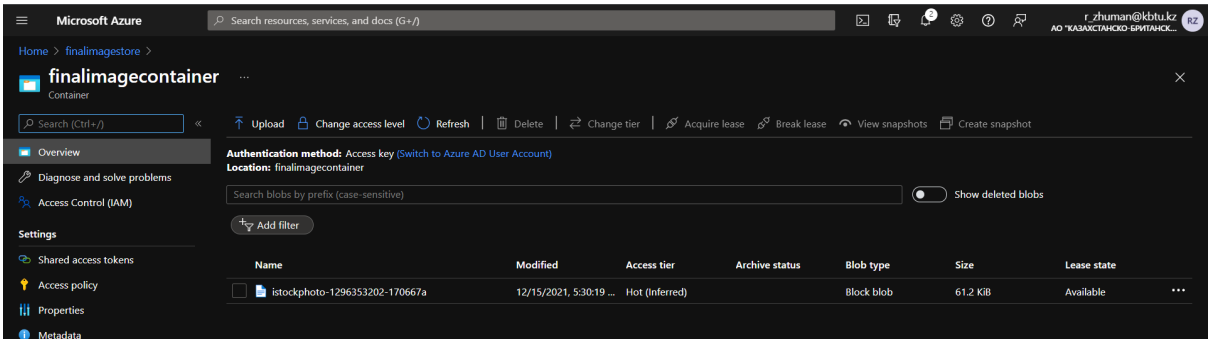
Function to upload Azure



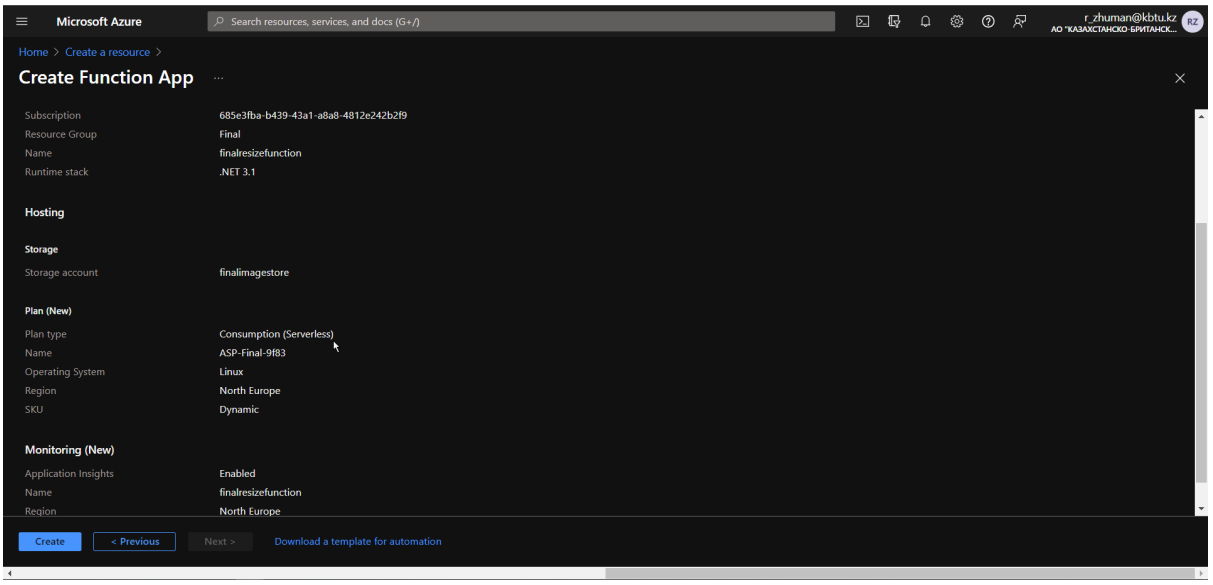
```
const { BlobServiceClient, StorageSharedKeyCredential } = require('@azure/storage-blob');
const path = require('path')

async function getContainer(){
  const account = "finalimagestore";
  const accountKey = process.env.AZURE_STORAGE_CONNECTION_STRING;
  const containerName = "finalimagecontainer"

  const sharedKeyCredential = new StorageSharedKeyCredential(account, accountKey);
  const blobServiceClient = new BlobServiceClient(
    `https://${account}.blob.core.windows.net`,sharedKeyCredential
  );
  return await blobServiceClient.getContainerClient(containerName);
}

const uploadToAzure = async (src) => {
  try {
    const blobName = path.basename(src).split('.')[0]
    const containerClient = await getContainer()
    const blockBlobClient = containerClient.getBlockBlobClient(blobName);
    blockBlobClient.uploadFile(src);
    return true
  } catch (err) {
    console.log(err)
    return false
  }
}

module.exports = {
  uploadToAzure,
}
```

Result

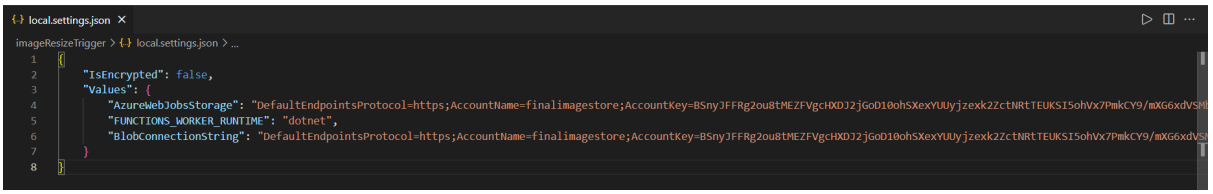
```
F:\Source\Azure\console-app>node index.js
Please enter image path or type 'quit' to exit
F:\Source\Azure\console-app\images\istockphoto-1296353202-170667a.jpg
F:\Source\Azure\console-app\images\istockphoto-1296353202-170667a.jpg
Catch your file istockphoto-1296353202-170667a.jpg.
Would you like to upload it to Azure? Y/N
y
Your file istockphoto-1296353202-170667a.jpg successfully uploaded
Please enter image path or type 'quit' to exit
quit

BYE BYE !!!

F:\Source\Azure\console-app>
```



Create Function App



Update keys in HW1 with new credentials from storage account



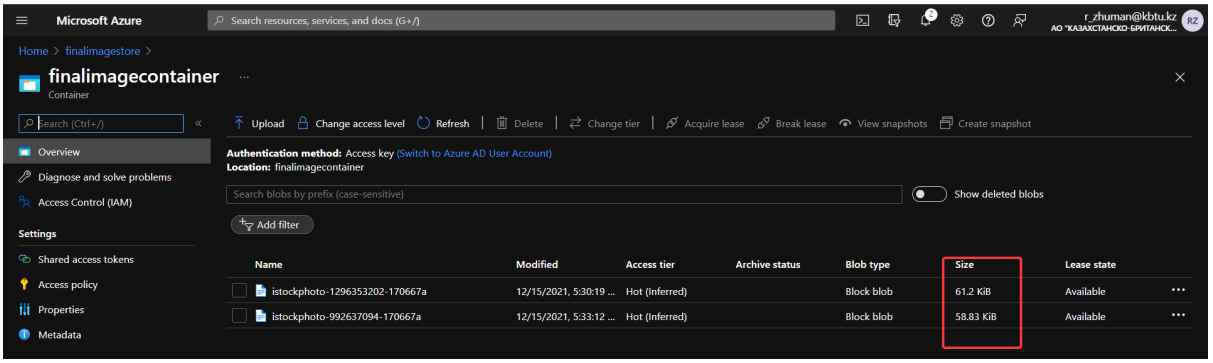
Publish function app

```
F:\Source\Azure\HW\HW1\HW1\imageResizeTrigger>func azure functionapp publish finalresizefunction
Microsoft (R) Build Engine версии 16.7.2+b60ddb6f4 для .NET
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

  Определение проектов для восстановления...
  Все проекты обновлены для восстановления.
  imageResizeTrigger -> F:\Source\Azure\HW\HW1\HW1\imageResizeTrigger\bin\publish\imageResizeTrigger.dll

Сборка успешно завершена.
    Предупреждений: 0
    Ошибок: 0

Прошло времени 00:00:04.95


Getting site publishing info...
Uploading package...
Uploading 5,01 MB [########################################################################################]
Upload completed successfully.
Deployment completed successfully.
Syncing triggers...
Functions in finalresizefunction:
    imageResizeTrigger - [blobTrigger]


F:\Source\Azure\HW\HW1\HW1\imageResizeTrigger>
```

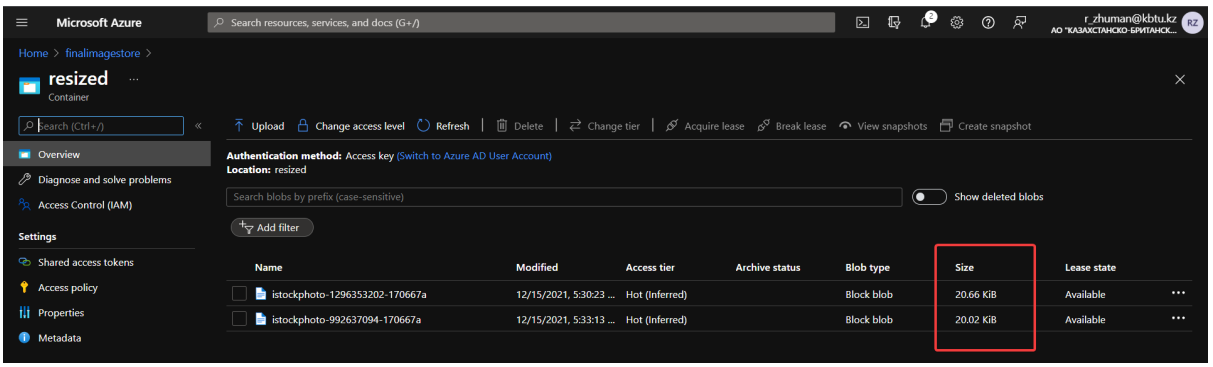
Upload new image

```
F:\Source\Azure\console-app>node index.js
Please enter image path or type 'quit' to exit
F:/Source/Azure/console-app/images/istockphoto-992637094-170667a.jpg
F:/Source/Azure/console-app/images/istockphoto-992637094-170667a.jpg
Catch your file istockphoto-992637094-170667a.jpg.
Would you like to upload it to Azure? Y/N
y
Your file istockphoto-992637094-170667a.jpg successfully uploaded
Please enter image path or type 'quit' to exit
quit

BYE BYE !!!

F:\Source\Azure\console-app>
```

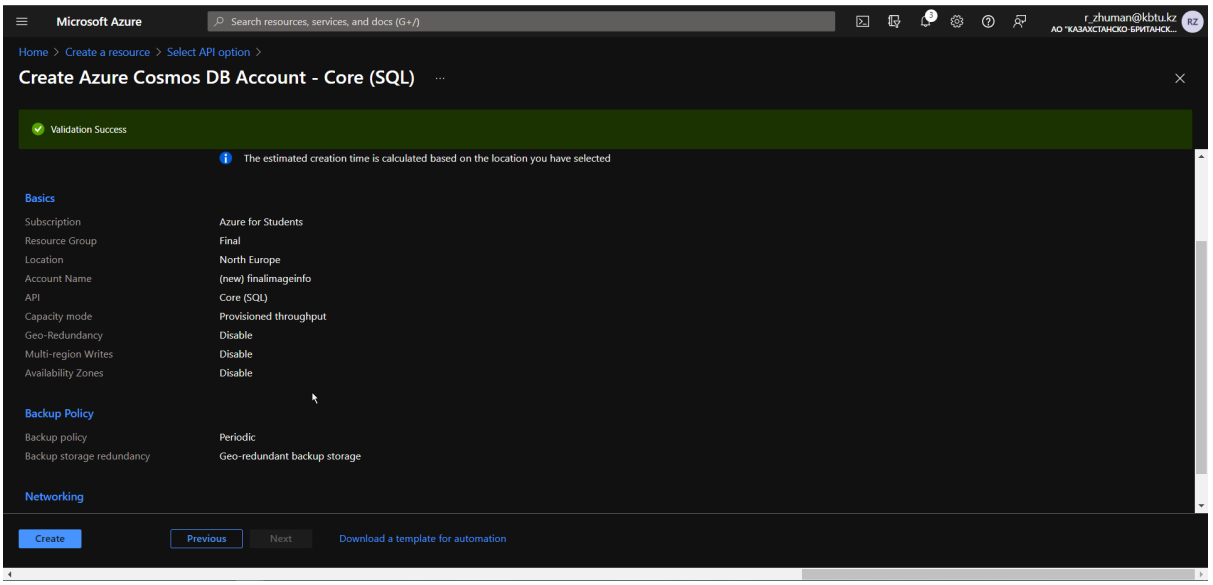
Origin



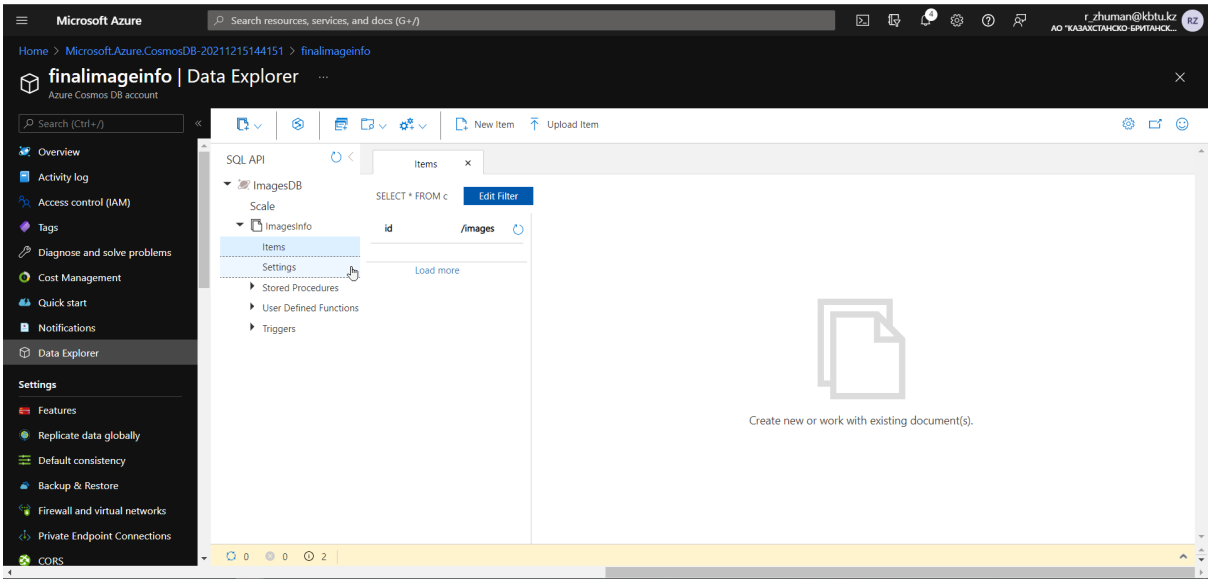
Resized



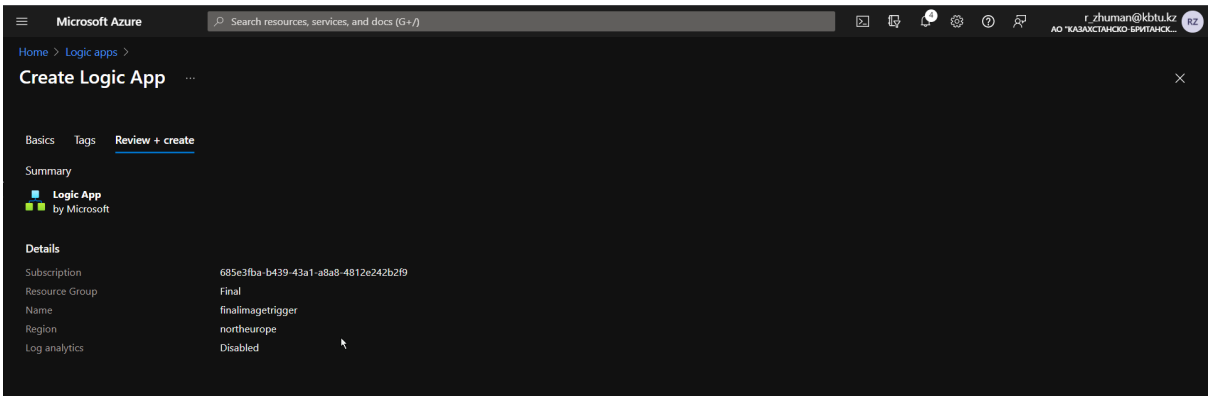
Create Azure Cosmos DB Account



Create Database and Container

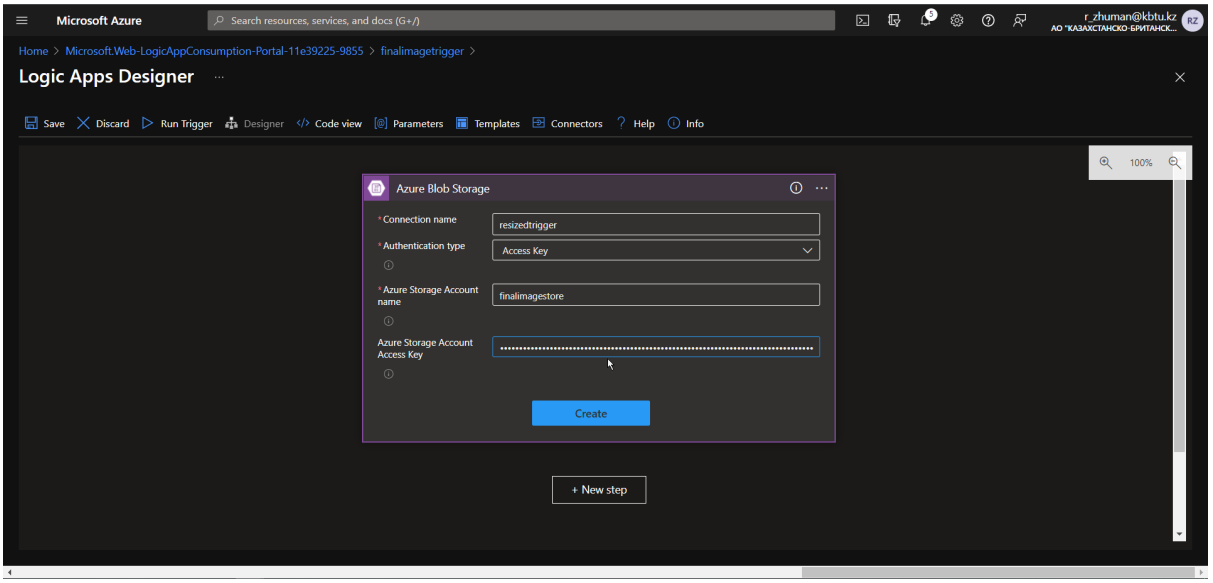


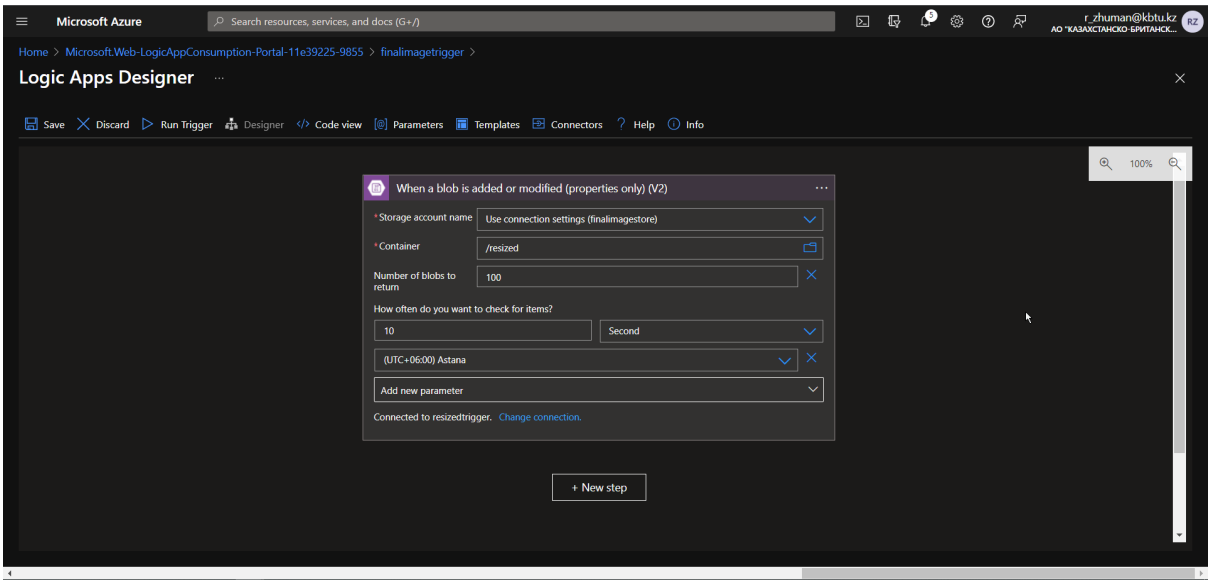
Create Logic App



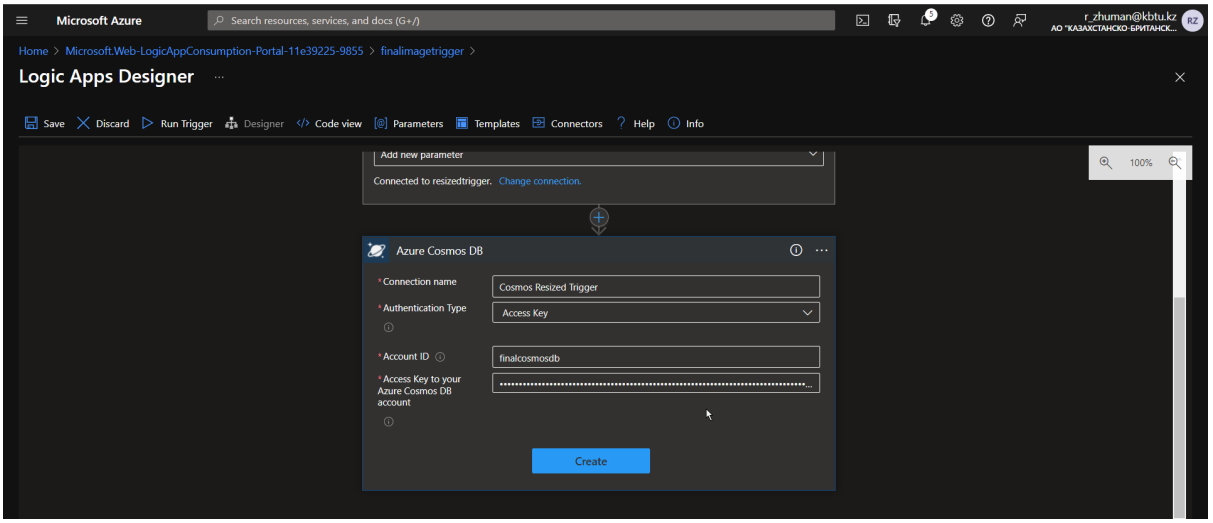
Design Logic App

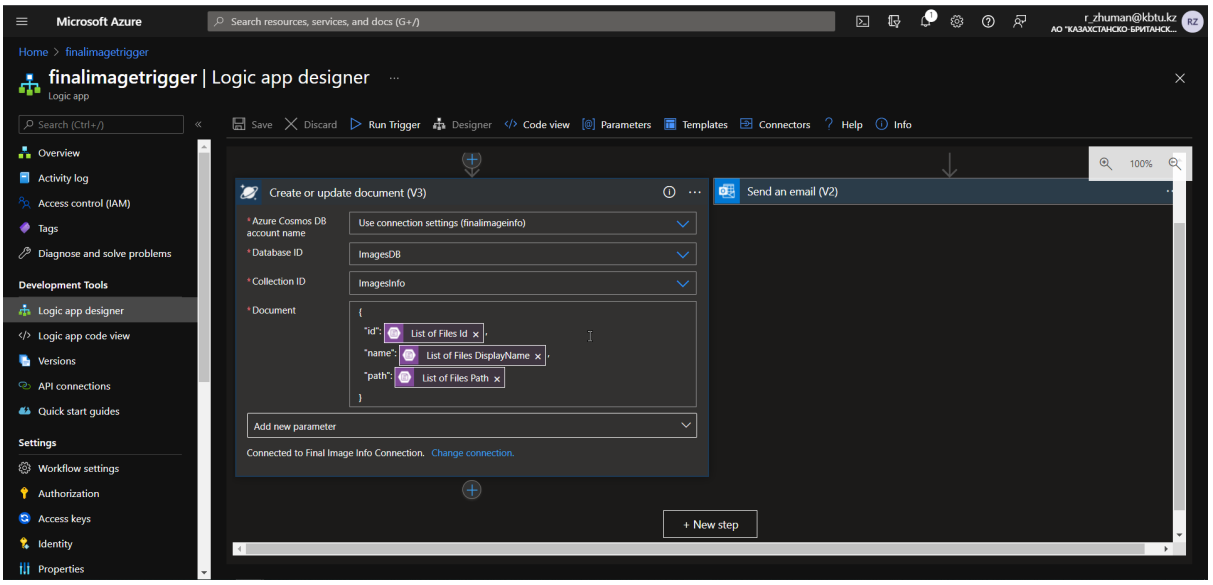
Configure container settings



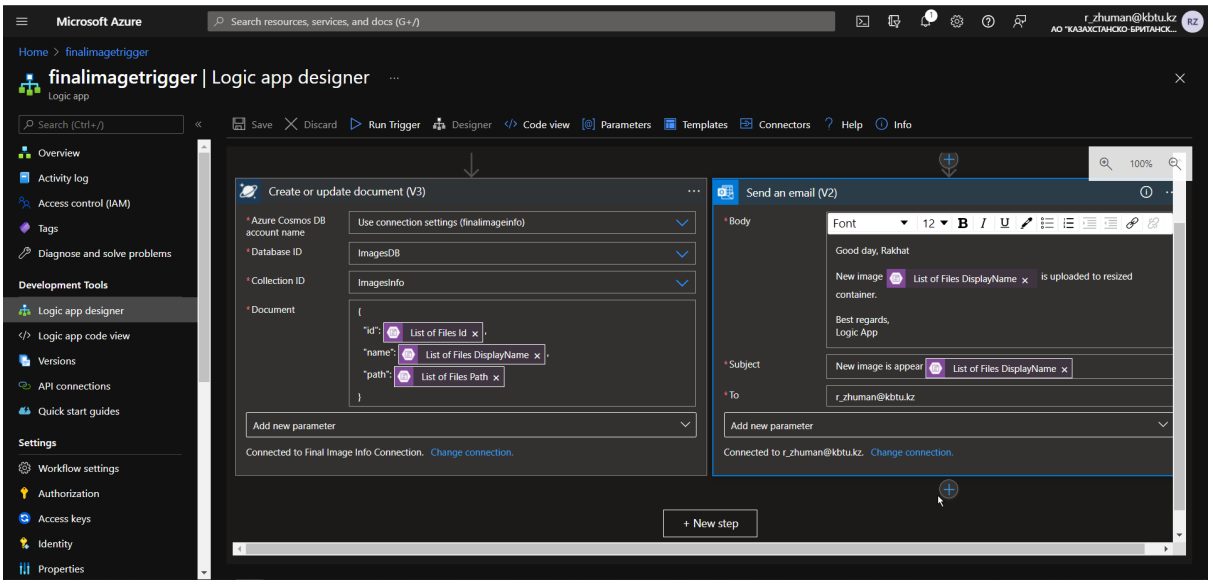


Create and configure Cosmos DB settings



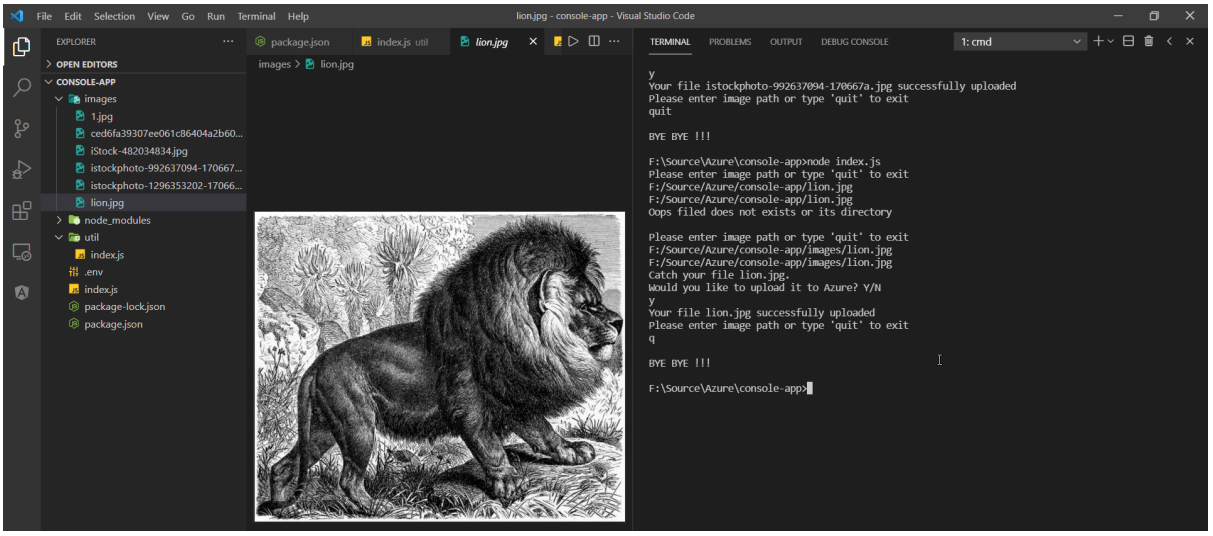


Create and config Outlook settings

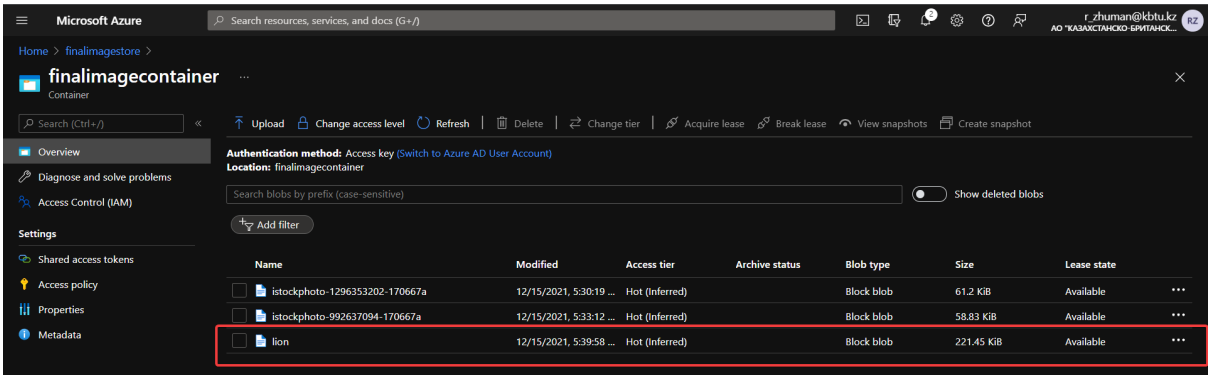


Tests

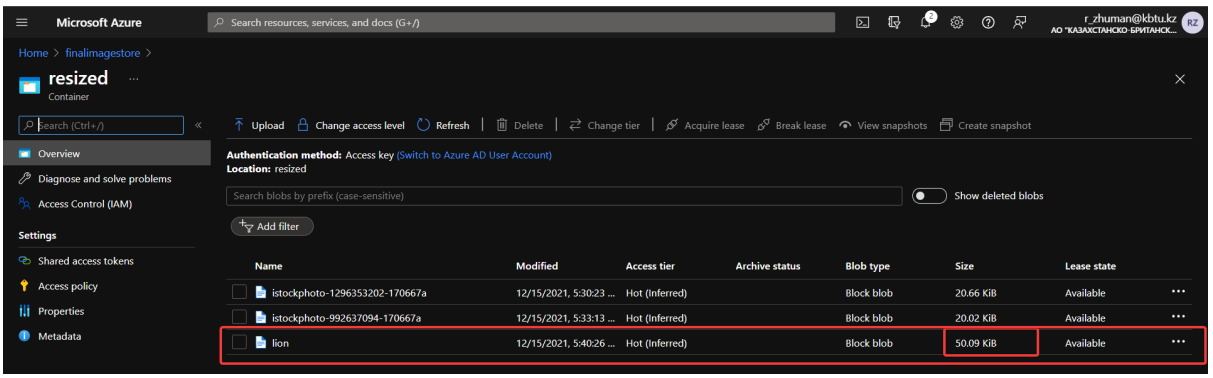
Upload image via console app



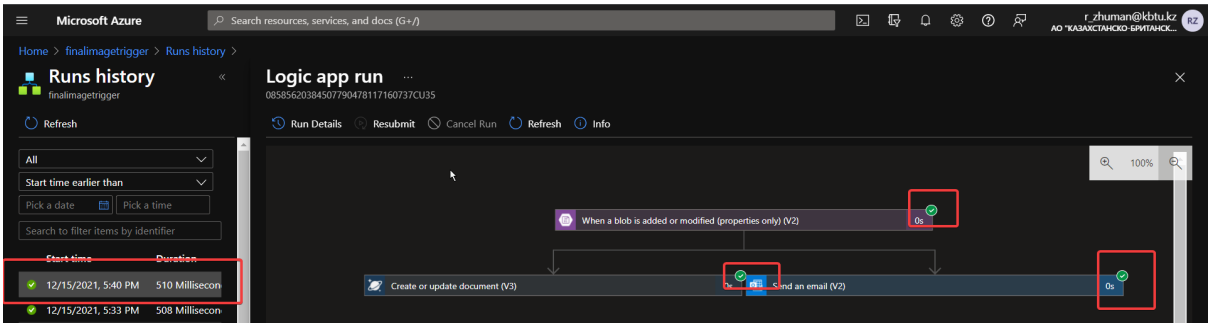
Check in blob container



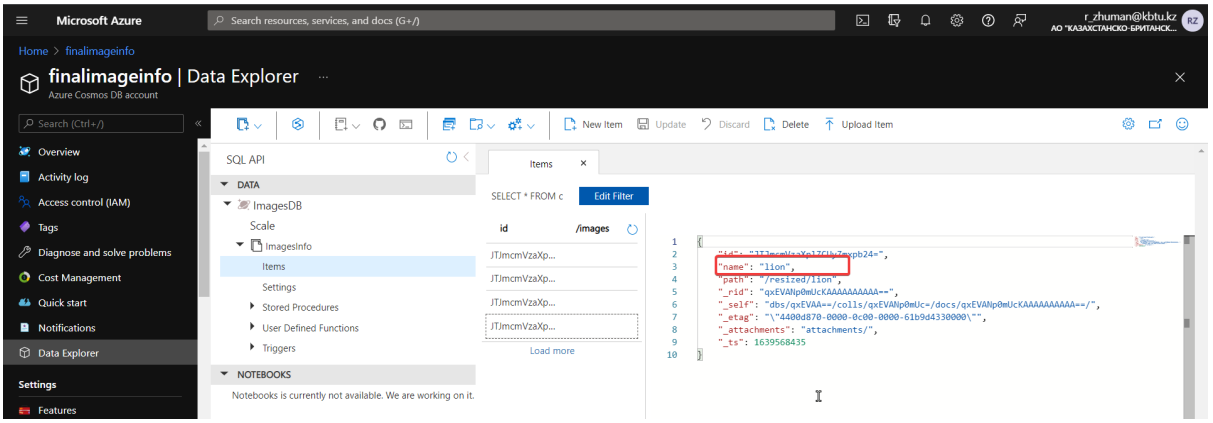
Check is function resized image



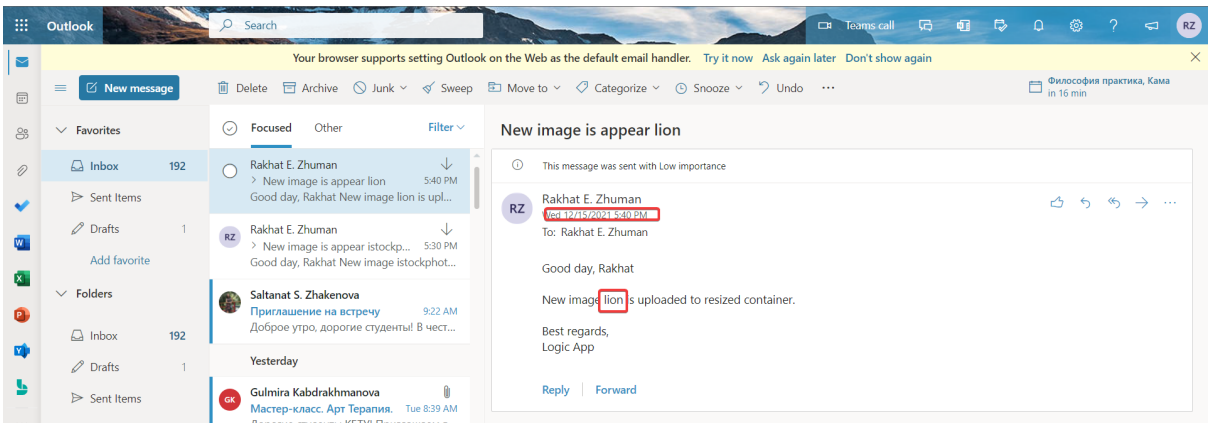
Check is trigger success

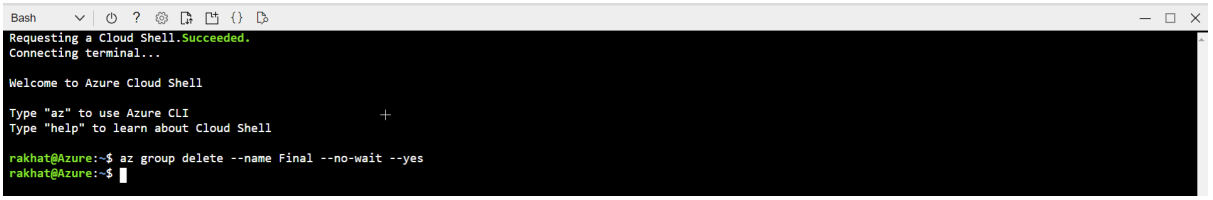


Check record in Cosmos DB



Check Outlook



Tests ended successfully!

Clean up subscription