COMP 4980-03 Applied Cloud Computing

Final Exam, Fall 2022

Name: Student ID:

1. (2 marks) In DynamoDB, explain what tables, items, and attributes are with examples.

* Table is a collection of data organized as key-value pairs. Serves as the top-level container for data.
  + Table named Users to store user data (username, email, password, etc.)
* Item is a single record in a table. Each item is uniquely identifiable by a primary key.
  + { “Username”: martin, “Password”: password }
* Attributes are individual pieces of data that make up an item. They can store different data types.
  + In the table above, “Username” and “Password” are attributes of the item.

1. (2 marks) List four HTTP methods with corresponding CRUD operations, which are used in RESTful APIs.

|  |  |
| --- | --- |
| HTTP method | CRUD operation |
| POST | CREATE |
| GET | READ |
| PUT | UPDATE |
| DELETE | DELETE |

1. (2 marks) Explain what CORS is, and list the full name.

Cross-Origin Resource Sharing.

Security standard that allows client web applications to interact with resources from other domains/web servers. It uses HTTP headers to indicate which origins a browser can load resources from.

1. (2 marks) Explain what AWS Lambda is. List two advantages of Lambda.

Lambda is a serverless compute service provided by AWS that enables you to run code without providing or managing servers. Uploaded code can execute in response to specific events, such as HTTP requests, database updates, or file uploads to Amazon S3.

* Cost Efficiency – Pay-as-you-go pricing model, meaning you only pay for the compute time your code consumes.
* Simplified Management – With Lambda you don’t need to manage servers, OS, or infrastructures. AWS handles updates, patches, and maintenance.

(4 marks) Complete the next code that reads “memo.html” and sends the HTML content back to the client side.

const fs = require('fs');

exports.handler = (event) => {

const html = fs.readFileSync('memo.html', { encoding: 'utf-8' });

const response = {

statusCode: 200,

headers: { ‘Content-type’: ‘text/html’},

body: html

};

return new Promise((resolve, reject) => { resolve(response) });

}

1. (6 marks) Complete the next code that reads the ‘NextId’ value of the item of Id = -1 from the ‘Memos’ table and updates the item with the new ‘NextId’ value which is increased by 1.

var AWS = require('aws-sdk');

var dbclient = new AWS.DynamoDB.DocumentClient();

let params = {

TableName: 'Memos',

Key: { Id: -1 }

};

Let tableObject = await dbclient.get(params, (err, data) => {

if (err) console.log(err);

else console.log(data.Item);

}).promise();

params = {

TableName: 'Memos',

Item: {

Id: -1,

NextId: tableObject.Item.Id + 1

}

await dbclient.update(params, (err, data) => {

if (err) console.log(err);

}).promise();

1. (4 marks) Complete the code that processes the ‘ADD’ request sent from the memo client through the API Gateway, ‘Memo-API-DynamoDB’.

const serverless = require('serverless-http');

const express = require('express');

const app = express();

app.use(express.urlencode({extended:true}));

app.use(express.json());

module.exports.handler = serverless(app);

app.post('/memos', (req, res) => {

addMemo(req, res);

});

function addMemo(req, res) {

let memo = req.body.memo;

addMemoIntoMemos(memo);

res.send({

result: 'OK'

});

}

1. (2 marks) Complete the code that reads the environment variable, ‘UploadBucket’.

const uploadBucket = process.env.UploadBucket;

1. (2 marks) List the s3 methods in the next table.

const aws = require(‘aws-sdk');

const s3 = new aws.S3();

|  |  |
| --- | --- |
| s3 method | Purpose |
| getSignedUrl | To obtain an URL to upload a file |
| getObject | To read a file from a bucket |
| putObject or upload | To upload a file to a bucket |