UNIVERSITY OF LONDON

CM2040

BSc EXAMINATION

COMPUTER SCIENCE

Databases, Networks and the Web

Release date: Wednesday 6 March 2024 at 12:00 midday Greenwich Mean Time

Close date: Thursday 7 March 2024 by 12:00 midday Greenwich Mean Time

Time allowed: 4 hours to submit

INSTRUCTIONS TO CANDIDATES:

Part A of this assessment consists of a set of **TEN** Multiple Choice Questions (MCQs). You should attempt to answer **ALL** the questions in **Part A.** The maximum mark for Part A is **40**.

Candidates must answer **TWO** out of the **THREE** questions in **Part B**. The maximum mark for Part B is **60**.

Part A and Part B will be completed online together on the Inspera exam platform. You may choose to access either part first upon entering the test area but must complete both parts within **4 hours** of doing so.

A handheld non-programmable calculator may be used when answering questions on this paper but it must not be able to display graphics, text or algebraic equations. Please hold your calculator to the camera at the start of the examination to clearly show the make and type.

You may use **ONE** A4 page of previously prepared notes in this examination. Please hold up your notes to the camera at the start of the examination.

File upload is **NOT** permitted.

Do not write your name anywhere in your answers.

PART A

Candidates should answer the **TEN** Multiple Choice Questions (MCQs) in Part A.

PART B

Candidates should answer any **TWO** questions in Part B.

Question 2

This question is about web application development.

You are a middleware developer building a web application for a new health club which will be opening in 6 months time. The health club is currently called ‘Unicorn Health’, but the name may change. The club has a pool and a gym.

The club requires an online form to allow new visitors to the site to sign up for the monthly newsletter, which will be sent by email.

You have written a node.js application with a route that adds data collected from a 'sign-up' form to a new user record in the database.

The form looks like this:

The database is called `unicorn_health`. A table is created in the database as follows:

```
CREATE TABLE  users (
    user_id INTEGER PRIMARY KEY AUTO_INCREMENT,
    name    VARCHAR(50) NOT NULL,
    email   VARCHAR(10) NULL
);
```

The main webserver code is in `index.js`:

```
// Set up express, bodyparser and EJS
const express = require('express');
const app = express();
const port = 3000;
var bodyParser = require("body-parser");
app.use(bodyParser.urlencoded({ extended: true }));
app.set('view engine', 'ejs');

// Set up MySQL
const mysql = require('mysql');
global.db  = mysql.createConnection ({
    host: 'localhost',
    user: 'appuser',
    password: 'dja873jakjs',
    database: 'unicorn_health'
});
// Connect to the database
db.connect((err) => {
    if (err) {
        throw err;
    }
    console.log('Connected to database');
});

global.club_name = "Unicorn Health"

// Handle requests to the home page
app.get('/', (req, res) => {
    res.send('Welcome to ' + global.club_name)
});

// Users routes
const usersRoutes = require('./routes/users');
app.use('/users', usersRoutes);


// Make the web application listen for HTTP requests
app.listen(port, () => {
    console.log(`App listening on port ${port}`)
})
```

Code to handle the sign-up form is contained in a middleware module called `users.js`:

```
router.get("/signup", (req, res) => {
    res.render("signup.ejs", {club_name: global.club_name});
});


router.post('/signup', (req, res, next) => {
    let query = "INSERT INTO users (user_name) VALUES ( ?, ? )"
    let query_parameters = [req.body.name, req.body.email];
    db.query(query, query_parameters, (err, result) => {
        if (err) {
            next(err);
        } else {
            res.send('User added to the database');
        }
    })
})
```

The form is rendered from the following template, called `signup.ejs`:

```
<!DOCTYPE html>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet"  type="text/css" href="/main.css" />
    <title>Add User</title>
</head>
<body>
    <h1>Welcome to <%= club_name %></h1>
    <p>Sign up for our monthly newsletter here:</p>
    <form action="add-user" method="post">
        <p>Name: <input id="user" type="text" name="user_name" /></p>
        <p>Email: <input id="email" type="text" name="user_email" /></p>
        <button type="submit">Submit</button>
    </form>
</body>
</html>
```

(a) What is the purpose of these lines of code in `index.js`?

```
var bodyParser = require("body-parser");
app.use(bodyParser.urlencoded({ extended: true }));
```

[2 marks]

(b) State **TWO** reasons why the connection to the MySQL database in `index.js` might fail.

[2 marks]

(c) Looking at the definition of the `email` column in the `users` table, describe **TWO** issues that you think this definition might cause when the application is running in a real-life scenario.

```
    email    VARCHAR(10) NULL
```

[4 marks]

(d) Describe the purpose of the question marks in the string `query` in `users.js` and explain what happens to this string when the query is executed in the code below it.

```
    let query = "INSERT INTO users (user_name) VALUES ( ?, ? )"
    let query_parameters = [req.body.name, req.body.email];
    db.query(query, query_parameters, (err, result) => {
      ...
    })
```

[4 marks]

(e) Consider the following line of code in `signup.ejs`:

```
    <h1>Welcome to <%= club_name %></h1>
```

i. Explain the purpose of this line in relation to the requirements defined for the health club.

[2 marks]

ii. Explain the mechanism by which this line results in the text that actually shows in the signup form.

[4 marks]

(f) After running the website for a few months and gathering a few hundred signups, the health club wants to extend the form to additionally collect information about whether the user is interested in the gym, pool or both. Answer the questions below to explain the changes that are needed to the web application.

i. Describe the changes you would make to the `users` table in the database to accommodate these changes. There is no need to write code.

[2 marks]

ii. Describe the changes you would make to the signup form in `signup.ejs`. There is no need to write code.

[2 marks]

iii. Show the changes you would make to the code in the signup route handler in `users.js`.

[4 marks]

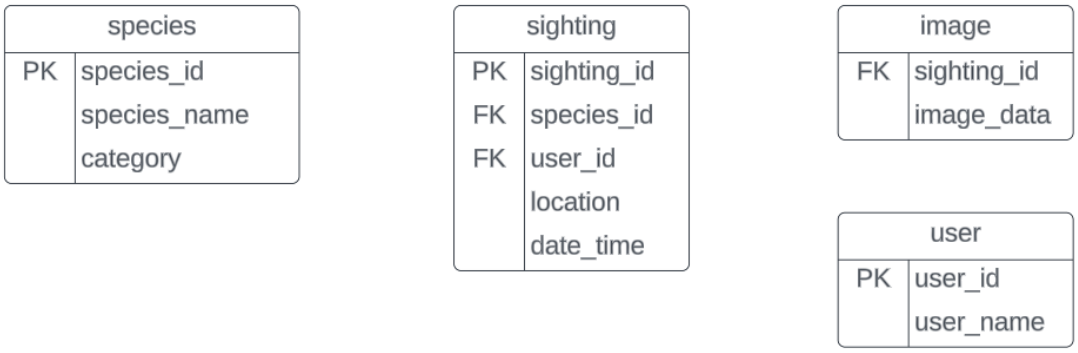
(g) You are ready to apply the changes you have identified in part (f). Your colleague suggests you should delete the database and recreate it because the table structure has changed. Why might this be a bad idea? What would an alternative approach be?

[4 marks]

Question 3

This question is about database modelling.

Consider the database schema below, which allows a group of wildlife enthusiasts to record their sightings of animals of interest:



In the first few days, the following data is captured:

species

species_id	species_name	category
1	Goldfinch	Bird
2	Starling	Bird
3	Smooth Newt	Amphibian
4	Bull Frog	Amphibian
5	Red Squirrel	Mammal

user

user_id	user_name
1	Miranda
2	Carlos

sighting

sighting_id	species_id	user_id	location	date_time
1	2	1	Richmond Park	2023-05-12 17:36
2	5	2	Regents Park	2023-05-14 12:10
3	4	2	Barnes Wetlands	2023-05-16 09:25
4	4	1	Barnes Wetlands	2023-05-17 09:38

The image table is used to upload photos of the animals seen.

(a) Identify the foreign key relationships. For each foreign key relationship, specify the two tables involved in the relationship and the cardinality (i.e. one-to-one, one-to-many or many-to-many), making sure you indicate the direction of the relationship.

[6 marks]

(b) Looking at the species table, you notice that storing the category name in the category column means the same category name is stored multiple times.

i. Describe **TWO** reasons why you think this approach for modelling the category name is not ideal.

[4 marks]

ii. Describe an approach to improve the model that resolves the problems you identified above, listing all the changes you would make to the data model.

[6 marks]

(c) Write queries for the requirements below, using the original data model, not the one you modified in question (b).

i. Write a SQL query to return the names of species that are stored.

[2 marks]

ii. Write a SQL query to return the names of amphibians that are stored.

[2 marks]

iii. Write a SQL query to return the location, date_time and species_name of all sightings.

[5 marks]

iv. Write a query to show the total number of sightings by each user. Show the user_name and number of sightings they have recorded.

[5 marks]

Question 4

This question is about networking and network protocols.

(a) Describe the steps involved in making a web request, showing how the HTTP protocol is used by a web application to facilitate communications between a client browser and a web server.

[5 marks]

(b) What information is passed in the request made from the browser to the web server?

[3 marks]

(c) What information is passed back to the browser in the response made by the web server?

[3 marks]

(d) From a security perspective, what issues are inherent in HTTP and how does HTTPS address these issues?

[4 marks]

(e) Explain the benefits of a three-tier architecture when compared to a two-tier architecture.

[5 marks]

(f) Explain how packet switching works and explain how it brings advantages in network communication. When listing advantages, explain how these advantages are achieved.

[5 marks]

(g) The three-way handshake is a key element of network communications.

i. Explain the purpose of the three-way handshake.

[2 marks]

ii. Explain how the SYN, SYN-ACK and ACK flags are used in the three-way handshake. Refer to the client (initiating device) and server (target device).

[3 marks]

END OF PAPER