# cid:ii_hvdlwih90_14613f15db7a4b70

# LABORATORY SESSIONS

**NAME OF SCHOOL:** SCIENCE AND TECHNOLOGY

**SEMESTER:** SPRING 2023

**COURSE:** APT1050: DATABASE SYSTEMS

**LECTURER : Prof. Elisha Toyne O. Omulo**

***TIME/DAYS:***  *ALL DAYS UNTIL 16/3/2023*

***VENUE:***  *SHSS ICTLAB*

***CREDIT:***  *3 UNITS*

***OFFICE HOURS:*** *ICT CENTRE LILINA BEAM BUILDING FACULTY OFFICE 10*

***CONTACT:*** *EMAIL:* [*elopiyo@usiu.ac.ke*](mailto:elopiyo@usiu.ac.ke)

**NUMBER:** 666248 **NAME:**  SHARON JUSTINE NALUBIRI

**1. INSTRUCTIONS**

1. Laboratory sessions are designed to help you strengthen your hands on experienced for the course APT1050A/B database systems.
2. You need to work alone and document your experiences.
3. Submit the final report by week 9, latest week 10. It will count for your participation.
4. Use this document when reporting.
5. Your name and number should be on this document.
6. Send your final report to [toyneo53@gmail.com](mailto:toyneo53@gmail.com) or the mail above.

====================================================================

**LABORATORY SESSION 1**

**Task:**  Conduct WEB search for the following items and report.

1. Available pdf version of the class text book, and download.

2. Any other one different textbook on Database Systems.

3. Use not more than 3 sentences to summarize the findings when you type the keyword phrase: ‘Database systems in 2023’

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATES | SITES LOCATED | PDFS DOWNLOADED | CHALLENGES | COMPUTING PLATFORM USED- ON CAMPUS OR OFF CAMPUS |
| 18/2/2023 | <https://cs.franklin.edu/~crawforl/DB10th.pdf> | DATABASE PRINCIPLES | I managed to download the book without any challenges. | Off-Campus |
| 18/2/2023 | <https://bayanbox.ir/view/8736593520639826197/Ramakrishnan-Database-Management-Systems-3rd-Edition-1-1.pdf> | "Database Management Systems" by Raghu Ramakrishnan and Johannes Gehrke | Finding the right link to download from was a bit challenging. | Off-Campus |

**2.**

* "Database Management Systems" by Raghu Ramakrishnan and Johannes Gehrke <https://bayanbox.ir/view/8736593520639826197/Ramakrishnan-Database-Management-Systems-3rd-Edition-1-1.pdf>
* "Database Systems: The Complete Book" by Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom

<https://people.inf.elte.hu/miiqaai/elektroModulatorDva.pdf>

**3.**

However, based on current trends and advancements, it is expected that database systems in 2023 will continue to evolve towards cloud-based and hybrid systems, with increased use of NoSQL and NewSQL databases for big data processing and real-time analytics, while also incorporating more advanced security and privacy measures to protect sensitive data.

**LABORATORY SESSION 2**

**Task:** Conduct WEB search for the following items and report on:

1. List of database management systems in 2023.

2. Download and install Oracle 21c for Windows or Linux depending on your platform.

3. Download and acknowledge the source of one Conceptual Schema not highlighted anywhere in this course.

|  |  |  |  |
| --- | --- | --- | --- |
| DATES | ITEMS AND SITES USED TO DOWNLOAD THEM | CHALLENGES/INSTALLED | COMPUTING PLATFORM USED- ON CAMPUS OR OFF CAMPUS |
| 20/02/2023 | 1) ManageEngine Applications Manager – Best overall database tracking software.  2) DbVisualizer – Best for SQL database analysis and development.  3) Plan Explorer – SQL Query Analysis.  4) MySQL – Best For Managing business-critical SQL applications.  5) Oracle Database – Best for its security and reliability.  6) Valentina Studio – Best For Small organizations with limited budgets  7) Microsoft SQL Server – Best For comprehensive database management solution  8) IBM DB2 – Best for running mission-critical workloads for businesses  9) Hadoop HDFS – Best for Parallel processing  10) PHPMyAdmin – Best for Administration of MySQL over the internet  11) MongoDb – Best for Development and scaling  **LINK:**  <https://www.guru99.com/best-database-management-software.html> | I did not find any challenges accessing them. | OFF CAMPUS |
| 9/02/2023 | ORACLE 21C  <https://www.oracle.com/database/technologies/xe-downloads.html> | It was a bit had having to resave it in a different folder | OFF CAMPUS |

3.

One conceptual schema that is not highlighted in this course is the Object-Role Modeling (ORM) schema.

ORM is a conceptual modeling language used to represent the semantic structure of information systems. It was developed by Terry Halpin in the 1980s and is now widely used in software engineering, database design, and business analysis. Certainly! Here's an example of an ORM schema for a simple online shopping system:

An example of an ORM schema for an online shopping system

In this schema, we have four entities: Customer, Product, Order, and Payment. A customer can place one or more orders, and an order can contain one or more products. Each product can have one or more orders associated with it. An order can have one payment associated with it, and a payment can only be associated with one order. Additionally, each order must be associated with exactly one customer.

**LABORATORY SESSION 3**

**Task:**  Using an implementation of Oracle Express 18c or 21c:

1. Enter- user-name and password and leave. Create at least one table called mytable.

|  |  |  |  |
| --- | --- | --- | --- |
| DATES | INSTALLATION OF ORACLE DBMS | CHALLENGES/TABLE CREATED | COMPUTING PLATFORM USED- ON CAMPUS OR OFF CAMPUS |
| 24/2/23 | Successful after getting assistance | Username and password created successfully | Off Campus |
| 24/2/23 |  | Table successfully created. | Off-Campus |

CREATE TABLE mytable (

id INT PRIMARY KEY,

name VARCHAR (50),

age INT,

email VARCHAR (100)

);

**LABORATORY SESSION 4**

**Task:**  Create a database and three tables (mytable.) in Microsof Access DBMS and enter at least 10 records in each of the tables.

|  |  |  |  |
| --- | --- | --- | --- |
| DATES | CREATION OF DATABASE AND TABLE IN MS ACCESS | CHALLENGES/ACCESS DATABASE CREATED AND TABLES POPULATED WITH DATA | COMPUTING PLATFORM USED- ON CAMPUS OR OFF CAMPUS |
|  | Open Microsoft Access.  Click "Blank database" in the "Available Templates" section of the startup screen.  Choose a location and name for your new database, then click "Create".  In the Navigation Pane on the left-hand side, click "Table Design" to create a new table.  In the "Table Design" view, add the columns you want for your first table. For example, you might create a table called mytable1 with columns for id, name, age, and email. Be sure to set the appropriate data types and field sizes for each column.  Save the table by clicking "Save" in the Quick Access Toolbar, or by pressing Ctrl+S. When prompted, enter a name for the table (e.g. mytable1) and click "OK".  Repeat steps 4-6 to create two more tables called mytable2 and mytable3, each with their own set of columns.  That's it! You now have a new Microsoft Access database with three tables: mytable1, mytable2, and mytable3. You can add data to each table by switching to "Datasheet" view, or you can create relationships between the tables using the "Relationships" tool in the "Database Tools" tab. | It was a bit tiresome | OFF CAMPUS |

**LABORATORY SESSION 5**

Task: Create an Oracle database schema tables for Product Orders see slides from week 5, slide number 16 [CUSTOMER, INVOICE, LINE, PRODUCT, and VENDOR];

1. Create the Schema.

2. Enter at least 15 records in each of the tables in the database.

3. Perform at least 5 queries using the SELECT command.

4. Modify at least one entry in each table.

5. Formulate at least 5 queries each with a different aggregated value.

6. Enforce referential integrity in at least two tables.

|  |  |  |  |
| --- | --- | --- | --- |
| DATES | CREATION OF DATABASE TABLES; SELECT command. | CHALLENGES/WORKS | COMPUTING PLATFORM USED- ON CAMPUS OR OFF CAMPUS |
| 09/03/2023 | Created tables; CUSTOMER, INVOICE, LINE, PRODUCT, and VENDOR | Initially it was challenging then after several attempts I was successful | OFF CAMPUS |
| 12/3/23 | I managed to use the update command to change the entries.  I also managed to use ‘sum’, ‘count’, ‘min’, ‘max’, and ‘average’ queries  Managed to create referential integrity between product and vendor by using primary key (V\_CODE) in Vendor table as foreign key in Product table | I took several takes, and with research on the internet I managed to | OFF-CAMPUS |

Open your RDBMS and create a new database.

CREATE DATABASE mydatabase;

Create tables for your database, defining the columns and data types for each table.

CREATE TABLE customers (

cust\_id INT PRIMARY KEY,

cust\_lname VARCHAR(50),

cust\_fname VARCHAR(50),

cust\_email VARCHAR(50),

cust\_initial VARCHAR(5),

cust\_areacode VARCHAR(10),

cust\_phone VARCHAR(14)

);

CREATE TABLE vendor (

v\_code INT PRIMARY KEY,

v\_name VARCHAR(50),

v\_contact VARCHAR(100),

v\_areacode VARCHAR(50),

v\_phone VARCHAR(50),

v\_state VARCHAR(20),

v\_order VARCHAR(50),

);

CREATE TABLE product (

pdt\_code INT PRIMARY KEY,

pdt\_name VARCHAR(50),

pdt\_descript VARCHAR(100),

pdt\_indate VARCHAR(50),

pdt\_QOH VARCHAR(10),

pdt\_min VARCHAR(12)

pdt\_price DECIMAL(8,2),

pdt\_discount DECIMAL(8,2),

v\_code INT,

FOREIGN KEY (v\_code) REFERENCES VENDOR(v\_code) ON DELETE CASCADE

);

CREATE TABLE invoice (

invoice\_num INT PRIMARY KEY,

customer\_id INT,

invoice\_date VARCHAR(10),

total\_amount DECIMAL(8,2),

FOREIGN KEY (cust\_id) REFERENCES CUSTOMER(cust\_id) ON DELETE CASCADE

);

CREATE TABLE line (

line\_num INT PRIMARY KEY,

invoice\_num INT,

pdt\_code INT,

line\_units INT,

line\_price DECIMAL(8,2),

FOREIGN KEY (invoice\_num) REFERENCES INVOICE(invoice\_num) ON DELETE CASCADE,

FOREIGN KEY (pdt\_code) REFERENCES PRODUCT(pdt\_code) ON DELETE CASCADE

);

4. Modify at least one entry in each table

UPDATE customers SET cust\_email = 'newemail@email.com' WHERE cust\_id = 1;

UPDATE pdt\_QOH SET quantity = 7 WHERE v\_code = 2;

5. Formulate at least 5 queries each with a different aggregated value.

-- Count the number of customers

SELECT COUNT(\*) FROM customers;

-- Get the average price of a product

SELECT AVG(pdt\_price) FROM product;

-- Get the total quantity of a product sold

SELECT SUM(pdt\_QOH) FROM product WHERE pdt\_name = 'Widget';

-- Get the minimum price of a product

SELECT MIN(pdt\_price) FROM product;

-- Get the maximum quantity of a product sold

SELECT MAX(pdt\_QOH) FROM product WHERE pdt\_name = 'Gadget';

**LABORATORY SESSION 6**

Task: SubQueries in SQL.

1. Using the schema from Laboratory Session 5, formulate at least 10 SQL Subqueries.

|  |  |  |  |
| --- | --- | --- | --- |
| DATES | SUBQUERIES IN SQL | CHALLENGES/WORKS | COMPUTING PLATFORM USED- ON CAMPUS OR OFF CAMPUS |
| 13/3/23 | Managed to use the ‘Where’, ‘Having’, ‘Select’ and ‘From’ subqueries | Had several tries but managed after researching on the internet and following YouTube tutorials | On-Campus |
| 14/3/23 | Kept practicing on writing the subqueries. | Successful | Off-Campus |