



**MYSQL END OF MODULE CHALLENGE ANSWER SHEET**

| **Assessor** | Luvuyo |
| --- | --- |
| **Moderator** | Joel / Ryan / Oslin |
| **Due Date** | 3 Feb 2023 |
| **PASS MARK** | 50% |
| **Student Name** |  |

**INSTRUCTIONS**

1. Please adhere to the due dates.

2. Coping/Plagiarism is not accepted.

3. Make sure you write the code in the space provided for each question

4. Upload the Answer Sheet together with the database dump (backup file)

**1. CREATING DATABASE AND MANIPULATING DATA**

A local shop owner has asked you to design a database which they will use to store information about their suppliers and the products they supply. They sell fresh produce to the public. They provided you with a list of their suppliers and the products they supply to help you design the database. The database consists of TWO tables (*Suppliers* and *Products*).

1.1 Database Creation. Paste your code in the space provided. **[3]**

| CREATE DATABASE FruitMarket; |
| --- |

1.2 Table Suppliers. Paste your code in the space provided. **[8]**

| CREATE TABLE suppliers(  supplierID VARCHAR(10) PRIMARY KEY NOT NULL,  companyName VARCHAR(30) NOT NULL,  contactPerson VARCHAR(30) NOT NULL,  contactNO VARCHAR(13) NOT NULL,  productCategory VARCHAR(55) NOT NULL  ); |
| --- |

3 marks for correct SQL statements

5 Marks for correct field-names and keys as stated above

1.3 Products table using details provided below.

**Solution**

Write your code on the space provided in the space provided below. [4]

| CREATE TABLE products(  productID INT PRIMARY KEY NOT NULL,  productName VARCHAR(30) NOT NULL,  price DECIMAL(10,2) NOT NULL,  weight VARCHAR(10) NOT NULL,  stock INT NOT NULL,  supplierID VARCHAR(10) NOT NULL,  CONSTRAINT product\_supplier FOREIGN KEY(supplierID)  REFERENCES suppliers(supplierID)  ON DELETE CASCADE  ON UPDATE CASCADE  ); |
| --- |

1 Mark for correct SQL and table

2 Marks for correct field-names and sizes, deduct 1 mark for wrong fields sizes and   
 1 mark for data types

1 mark for correct linking of the foreign key

1.4 Relationship [1]

| 1:1 One is to One relationship |
| --- |

1.5 Foreign key [1]

| Supplier ID. |
| --- |

1.6 Adding records to the database. [1]

Write your answer in the space provided.

| INSERT INTO suppliers(supplierID,companyName,contactPerson,contactNo,productCategory)  Values('SUPP0001','Fruit City','Themba','0115062089','Fruit'),  ('SUPP0002','Vegan Veggie Xpress','Chiyere','0137228936','Vegetables'),  ('SUPP0003','The Nut House','Sam','0216965133','Nuts'),  ('SUPP0004','The Lazy Cow','Angelo','0216964157','Dairy'); |
| --- |

1 Marks for each correct record

Deduct 1 Mark if a single insert is not used

1.7 Add records into the Products table. Write your answer in the space provided. [4]

| -- Fruit City  INSERT INTO products(productID,productName,price,weight,stock,supplierID)  Values(1001,'Lady Finger Bananas',17.95,'750g',45,'SUPP0001'),  (1002,'Pink Lady Apples',18.95,'1.5kg',15,'SUPP0001'),  (1003,'Red Anjou Pears',22.99,'1kg',24,'SUPP0001'),  (1004,'The Lazy Cow',12.65,'900g',18,'SUPP0001');  -- Vegan Veggie Xpress  INSERT INTO products(productID,productName,price,weight,stock,supplierID)  Values(2001,'Tenderstem Broccoli',35.90,'400g',8,'SUPP0002'),  (2002,'Portabellini Mushrooms',18.99,'250g',16,'SUPP0002');  -- The Nut House  INSERT INTO products(productID,productName,price,weight,stock,supplierID)  Values(3001,'Raw Almonds',99.00,'1kg',6,'SUPP0003'),  (3002,'Macroon Butter',32.95,'400g',9,'SUPP0003'),  (3003,'Organic Coconut Oil',89.95,'500ml',15,'SUPP0003');  -- Vegan Veggie Xpress  INSERT INTO products(productID,productName,price,weight,stock,supplierID)  Values(4001,'Ayrshire Milk',33.95,'3L',23,'SUPP0004'),  (4002,'Simonzola Blue Cheese',27.65,'270g',4,'SUPP0004'); |
| --- |

4 Marks for adding all the records correctly

Deduct 1 mark if a record is missing. Maximum of 2 marks

1.8 SQL query to extract the following ProductId, ProductName, Price, Weight, Stock, ProductCategory [6]

| -- Extract Records  SELECT p.productID,p.productName,p.price,p.stock,s.productCategory FROM products p  INNER JOIN suppliers s  USING (supplierID)  WHERE stock < 20  ORDER BY price DESC; |
| --- |

2 marks for correct SQL statement

Maximum of 2 marks for correct fields.

1 mark for the order

1 mark for permanently saving into the database

1.9 View Q1.9 [6]

Your Output in the space below:

| CREATE VIEW Q9 AS  SELECT  p.productID,  p.productName,  p.price,  p.weight,  p.stock,  s.productCategory,  FORMAT((p.price \* p.stock \* 1.15), 4) AS TotalPrice  FROM suppliers s  INNER JOIN products p  ON s.supplierID = p.supplierID; |
| --- |

1 Mark for the view

1 Mark for the correct fields

3 marks for calculation of total price and ceiling

1 mark for correct records

Deduct a mark for incorrect tax calculation, total calculation

1.10 User called ‘yourname\_initialofyoursurname’ with INSERT privileges on to the current database and Supplier Table. Write your answer in the space provided [3]

| CREATE USER 'matthew\_t'@'localhost' IDENTIFIED BY 'password';  GRANT INSERT ON FruitMarket.suppliers TO 'matthew\_t'@'localhost';  FLUSH PRIVILEGES; |
| --- |

1 mark for the correct user created

1 mark for the privileges

1 mark for correct database and table

1.11 Number of tables accessible. [1]

| ONLY 1. |
| --- |

1.12 Add a record with the following details to the Suppliers table

('SUPP006', 'Fruit&Veg', 'Abdu', '0216965111', 'Nuts'); [1]

| INSERT INTO suppliers(supplierID,companyName,contactPerson,contactNo,productCategory)  VALUES ('SUPP0006','Fruit&Veg','Abdu','0216965111','Nuts'); |
| --- |

1 mark for correct SQL and record details

No mark if the output is wrong

1.13 Display all records in the Suppliers table. [2]

| NONE at all to display, User=matthew\_t has INSERT privilege only and an error=1142 of denied privileges appears. |
| --- |

1 Mark for correct answer

1 Mark for reason

1.14 View called Q1.14 which will display ONLY the new user you created and the root. [2]

| CREATE VIEW DisplayUsers\_View AS  SELECT HOST,USER FROM mysql.user WHERE USER='root'  OR USER='matthew\_t'; |
| --- |

1 Mark for correct view

1 Mark for correct condition

**1.15** SQL statement which produces the given records and permanently saves them. Paste the code in the space provided below [3]

| CREATE VIEW ProductRecordDisplay\_View AS  SELECT \* FROM products WHERE productID = 4002 OR productID = 3001  OR productID = 2001 OR productID = 3002 OR productID = 4001  OR productID = 1003 OR productID = 1001 ORDER BY products.stock ASC;  COMMIT; |
| --- |

1 view created

1 mark for correct Select statements

2 mark for correct condition

**1.16** SQL statement which produces the following output.

Paste the code in the space provided. [3]

| Create View Q1\_16VIEW AS  SELECT  s.companyName,  s.contactNo,  p.productName,  p.price  FROM suppliers s  Left JOIN products p  USING (SupplierID); |
| --- |

1 Mark for creating view

1 Mark for correct fields [no mark for incorrect fields]

1 mark for the correct statement and correct records.

1.17 SQL code with output correctly formatted. [6]

| SELECT  SUM(price) AS 'Total\_unit\_price',  ROUND(AVG(price), 2) AS 'Average\_price',  COUNT(price) AS 'Number\_OF\_Products'  FROM  products; |
| --- |

1 Mark for permanent SQL statement

3 marks for the correct functions.

1 marks for the headings

1 mark for formatting output

1.18 SQL statement to display the number of distinct supplierId from the products table.   
 Add the SQL statement in the space provided. [3]

| CREATE VIEW Q1\_18 AS  SELECT p.productID,p.productName,p.price,p.stock,p.supplierID  FROM products p  INNER JOIN suppliers s  USING (supplierID); |
| --- |

1 Mark for creating a permanent sql statement

2 Mark for correct SQL statement producing correct output

[lecturer to award a mark if the statement is partially correct]

1.19 SQL statement to produce the following output as shown below.   
 Make sure you save the output permanently as Q1.19.  
 Add the SQL statement in the space provided. [4]

| SELECT supplierID, COUNT(productID)  AS 'Count(productID)'  FROM products  GROUP BY supplierID;  COMMIT; |
| --- |

1 Mark for permanently saving the solution on to the databases

3 Marks for correct SQL producing correct output

1.20SQL Statement to produce a given output. Add the code in the space provided.

[5]

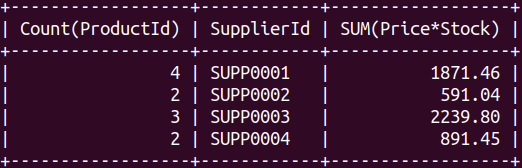
| SELECT  COUNT(productID)  AS 'Count(productID)',  supplierID,  SUM(price \* stock) AS 'SUM(Price\*Stock)'  FROM products  GROUP BY supplierID;  COMMIT; |
| --- |

1 Mark for saving output permanently

1 Mark for correct headings

2 Marks for correct functions

1 Mark for correct output



Make sure the result is visible in your database.   
 Add the SQL statement in the space provided in the LMS.

1.21 SQL Statement to produce a given output.   
 Add the code in the space provided. [2]

| UPDATE products SET productName = 'Cavendish Bananas',  price = 15.95, weight = '1kg'  WHERE productID =1004; |
| --- |

1 mark for correct SQL command

1 Mark for correct output

1.22Create a database dump called FruitMarketYourname.sql and upload it into the   
 LMS. [1]

| mysqldump -u root -p FruitMarket > FruitMarket.MatthewJesseThomas.sql |
| --- |

1 Mark for database dump backup

1.23 What is the purpose of applying ON Delete cascade in a foreign key constraint?   
 [2]

| If there in one’s parent table data can be deleted when all/selected rows in the child table are deleted then data that is referenced by other tables will not be deleted if cascading deletes were not put in. |
| --- |

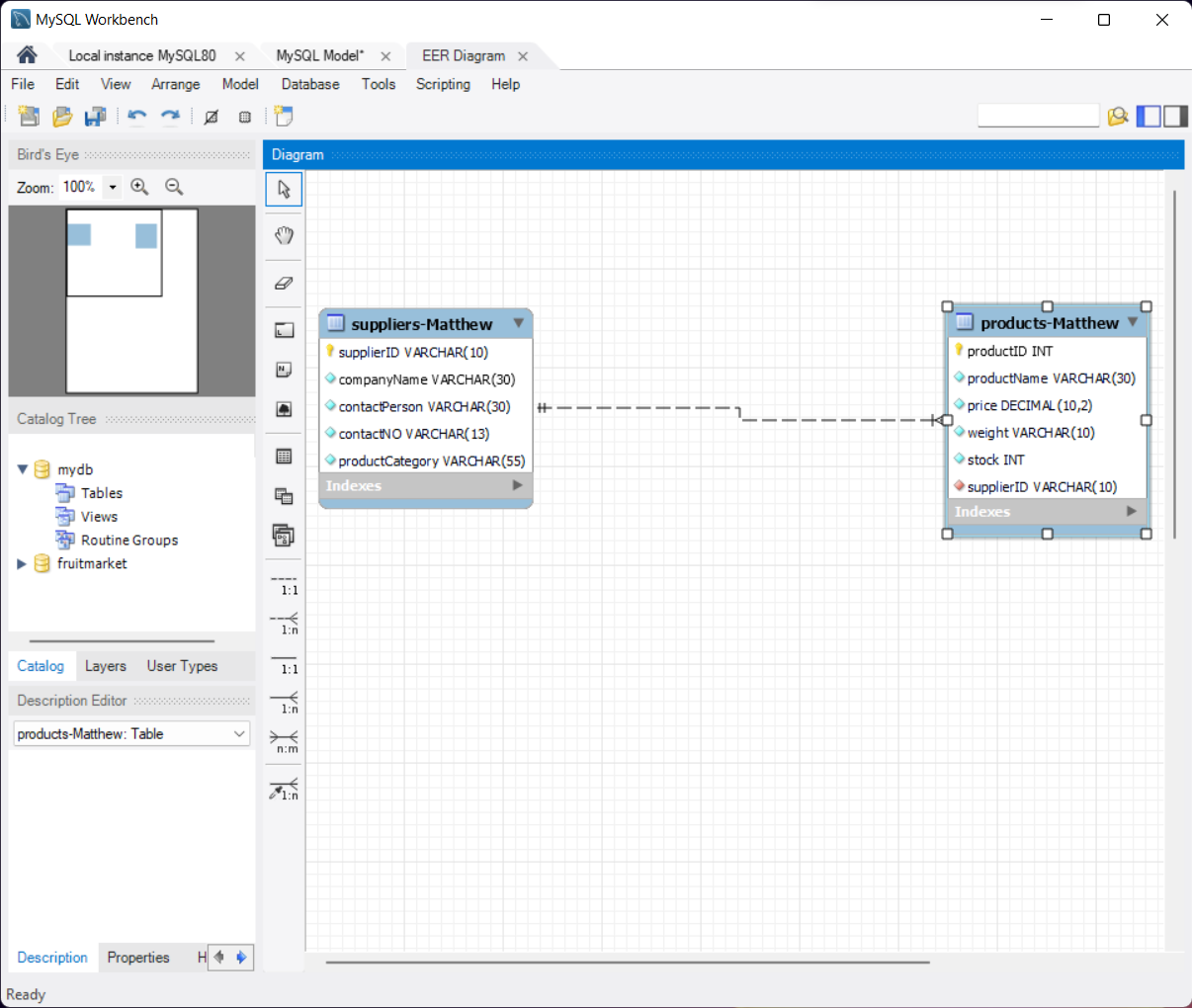
1.24 What is the difference between composite key and candidate key? [2]

| A composite Key is where there is a Primary Key an Foreign key which is unique in identifying a row and or column in a table but while a candidate key is a Super Key |
| --- |

1.25 What is the difference between SQL and DBMS [2]

| Database Management Systems and SQL are two of the most important and widely used tools. You use a Database Management System (DBMS) to store the data you collect from various sources, and SQL to manipulate and access the particular data you want in an efficient way. |
| --- |

1.26 EER Diagram [3]



1.27 Write a MySQL statement to extract the following records in that order. [3]

| SELECT \* FROM products WHERE productID = 4001 OR productID = 1004  OR productID = 2002 OR productID = 1002  ORDER BY products.stock DESC; |
| --- |

1.28 Write an SQL statement to for products table to produce the following records: [5]

| SELECT supplierID, SUM(price) AS 'Prices',  GROUP\_CONCAT(DISTINCT productName)  FROM products  GROUP BY supplierID  ORDER BY prices; |
| --- |

**QUESTION 2 NORMALISATION (30 Marks)**

Consider the following unnormalised table.

| **User\_ID** | **User\_Name** | **MSE\_ID** | **Rec\_Date** | **Subject** | **Text** | **Srvr\_ID** | **Server\_Name** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2301 | Smith | 54101 | 05/07 | Meeting Today | There is... | 3786 | IMAP05 |
| 2301 | Smith | 54098 | 07/12 | Promotions | I like to... | 3786 | IMAP05 |
| 2301 | Smith | 54445 | 10/06 | Next Assignment | Your next... | 3786 | IMAP05 |
| 5607 | Jones | 54101 | 05/07 | Meeting Today | There is... | 6001 | IMAP08 |
| 5607 | Jones | 54512 | 06/07 | Lunch? | Can you... | 6001 | IMAP08 |
| 5607 | Jones | 54660 | 12/01 | Jogging Today? | Can you... | 6001 | IMAP08 |
| 7773 | Walsh | 54101 | 05/07 | Meeting Today | There is... | 9988 | EMEA01 |
| 7773 | Walsh | 54554 | 03/17 | Stock Quote | The latest... | 9988 | EMEA01 |
| 0022 | Patel | 54101 | 05/07 | Meeting Today | There is... | 2201 | EMEA09 |
| 0022 | Patel | 54512 | 06/07 | Lunch? | Can we... | 2201 | EMEA09 |

Normalize the above table to Third Normal Form (3NF). Show all the tables

First Normal Form 2 tables (9 marks) 1 mark for each correct field.

Second Normal Form 3 Tables (10 Marks) 1 mark for each correct field.

Third Normal Form 4 Tables (11 Marks) 1 mark for each correct field.