

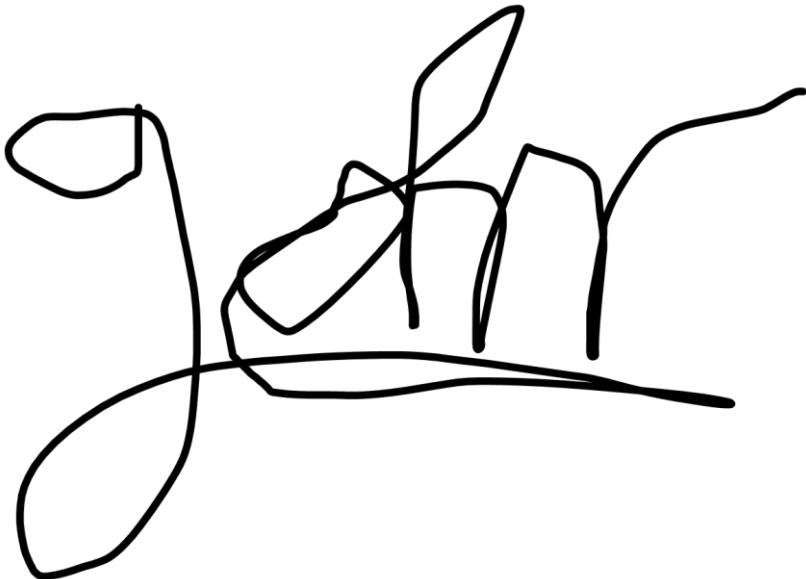
OCTO: Software Engineer Design and Manipulate Databases (DEMD)	
SAQA Qualification ID	119458
Intake / year	2-2025
Assessment Name	Summative Assessment
Module NQF Level	5
Credits	5
Open date and time	03 October 2025 • Section B: 17:15
Close date and time	03 October 2025 • Section B: 20:15
Assessment Total Marks	100 (50% of total mark percentage)
Pass percentage	65%

Lecturer Information	
Lecturer	Thaveshan Govender
Lecturer E-mail	thaveshang@aie.ac

IMPORTANT NOTICE

Before submitting your assessment, please ensure that it is signed and that all your personal details are accurately filled in the designated space below (Section in red). Work will not be accepted unless the required sections below are fully completed prior to submission, resulting in a 0% grade for this assessment.

The signature must be your valid handwritten signature and NOT your name and surname typed in the space provided.

Student Number:	461476		
Student name and surname:	John Herbst	Date:	2026/08/05
Student cell phone number:	0604944665		
Student Signature:			
Lecturer name and surname:	Thaveshan Govender	Date:	
Lecturer Signature:			
Assessor name and surname:		Date:	
Assessor Signature:			
Moderator name and surname:		Date:	
Moderator Signature:			

Instructions to Candidates

1. Read each question carefully.
2. You must answer Section B below the space provided named “STUDENTANSWER”. Once completed, the answer sheet, with your answers, must be converted to PDF and uploaded to the student portal. Make sure that all additional information is uploaded in a single Zipped folder.
3. QCTO students who achieve between 55-64% may have a further submission opportunity, subject to the departmental rules.
4. **NO LATE submissions will be accepted. (System will automatically close when due date and time is reached). If you encounter any difficulty with the student portal, immediately take a screenshot, log a ticket (attention support), and attach your assessment to the ticket.**
5. You must follow the departmental formatting guidelines.
6. You must write your name, surname, and student number on your script. A failure to do so will result in your script not being marked.
7. By submitting your answer sheet for marking, you declare the following:
 - a. You are familiar with the AIE’s policies on academic dishonesty and plagiarism;
 - b. Your submitted is your own original work. Where you have made use of someone else’s work, you have duly referenced such work making using of the APA style of referencing;
 - c. You have not allowed, and will not allow any other student to copy your work; and
 - d. You did not copy the work of another student.
8. Students that are dishonest will be subjected to disciplinary procedures.

Assessment Objective/s:

The Formative assessment will cover the following assessment topics:

No	PM number	PS Description
1.	PM-02-PS01	Design a database
2.	PM-02-PS02	Manipulate data in the database

Scope:

This module is crafted to offer you a unique opportunity to immerse yourself in the activities related to system design, moving beyond theoretical knowledge to the practical application of designing effective and efficient systems. It aims to equip you with the skills necessary to navigate the complexities of system design, from the initial problem definition to the final design solution.

Technical Aspects:

Please verify that all pages are included when downloading this assessment by checking the page numbers at the bottom of the document.

Ensure the following font and size is used in this assessment

- a. Font: Arial
- b. Size: 12
- c. Font colour: Black
- d. **Ensure clear and adequate screenshots are taken where requested**

Save and upload the report as a .PDF (**No backgrounds**) with the following naming convention:

- a. StudentName_ StudentSurname _Student no _ModuleCode _SA1

Mark allocation for report

See Mark allocation sheet below

Scenario:

You have been appointed as the Junior Database Developer for QuickCart, a growing online retail company. The company has requested a new backend database system to manage its inventory, customer records, and sales transactions for its various products. Once the database is designed, the next objective is to manipulate the data using SQL for reporting and operational purposes.

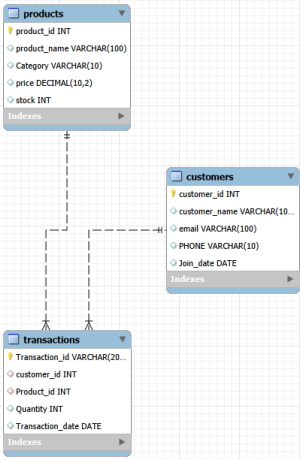
Use the “PM02_OnlineRetail_Database.xlsx” excel as a reference to complete questns below.

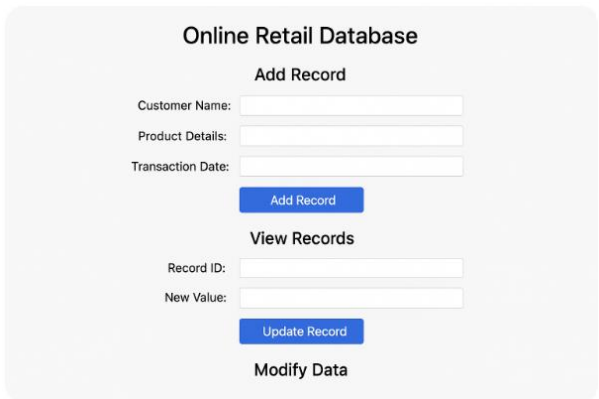
Question 1

(20)

<i>PM No</i>	<i>PS No</i>	<i>PS Description</i>
02	01	Design a database

a)	Create a Data Model - Design an ER diagram using information from the “PM02_OnlineRetail_Database.xlsx” excel spreadsheet. - Identify key attributes and data types. - Indicate primary and foreign keys.	[10]
	ANSWER: (Upload screenshot if necessary)	

		
b)	Construct Tables from the ER diagram - Translate the diagram into SQL CREATE TABLE statements. - Include appropriate data types and constraints.	[15]
	ANSWER: (Upload screenshot if necessary) <pre> create database Online_Retail_db; Use Online_Retail_db; create table Customers(customer_id int auto_increment primary key, customer_name varchar(100), email varchar(100) unique, PHONE VARCHAR(10), Join_date date); create table Products(product_id int auto_increment primary key, product_name varchar(100), Category VARCHAR(50), price decimal(10,2), stock int); </pre>	

	<pre> create table Transactions(Transaction_id varchar(200) Primary key, customer_id int , Product_id Int, Quantity int, Transaction_date date, foreign key (customer_id) references Customers(customer_id), foreign key (Product_id) references Products(Product_id)); </pre>	
c)	Define Relationships Between Tables - Implement 1:1, 1:M or M:M relationships between your tables. - Show the following in your design: - Relationship design - Constraints - Cardinality explanation	[15]
	ANSWER: (Upload screenshot if necessary) Products to transactions = 1:M Customers to Transactions = 1:M	
d)	Link Database to a Basic Application - Create a simple front-end (mock-up or actual UI) that interacts with the database. - Demonstrate adding, viewing, or modifying records. Use the following UI as guidance: 	[10]
	ANSWER: (Upload screenshot if necessary)	

Add Record

All Records

Transaction ID	Customer	Product	Qty	Date
ac079786-b2ef-4b89-9f59-ccc9e8f86df8	John	Beans	1	2026-05-08

Modify Data

Customer ID

New Customer Name

Update Record

Add Record

Customer Name

John

Product Name

Beans

Quantity

1

Transaction Date

mm/dd/yyyy

Add Record

All Records

Transaction ID	Customer	Product	Qty	Date
ac079786-b2ef-4b89-9f59-ccc9e8f86df8	John	Beans	1	2026-05-08

```

import mysql.connector
import uuid

app = Flask(__name__)

# =====
# DATABASE CONNECTION
# =====
def get_db():
    return mysql.connector.connect(
        host='localhost',
        user='root',
        password='J0hn@1508#',
        database='Online_Retail_db'
    )

# =====
# HOME (LOAD + DISPLAY DATA)
# =====
@app.route('/')
def home():
    conn = get_db()
    cursor = conn.cursor()

```

All Records

Transaction ID	Customer	Product	Qty	Date
ac079786-b2ef-4b89-9f59-ccc9e8f86df8	albert	Beans	1	2026-05-08

Modify Data

Customer ID

1

New Customer Name

albert

Update Record

Question 2

(50)

<i>PM No</i>	<i>PS No</i>	<i>PS Description</i>
02	02	<i>Manipulate data in the database</i>

Appropriate SQL code must be used here in the SQL environment.

a)	Select Data from Tables	[10]
----	--------------------------------	------

- Query all customers who made purchases in the last month.
- List available products in stock with quantity > 10.

ANSWER: (Upload screenshot if necessary)

61

62 • `Select * from products where stock > 10;`

63

Result Grid		Filter Rows:		Edit:	
	product_id	product_name	Category	price	stock
▶	11	youM	apples	416.41	98
	12	Thatshop	tomatoes	313.07	15
	13	SuuplierFoods	pepsi	432.24	17
	14	BestItems	Books	292.40	18
	15	Buynow	spinache	355.25	19

```

36 SELECT DISTINCT
37     c.customer_id,
38     c.customer_name,
39     c.email,
40     t.Transaction_date
41 FROM customers c
42 JOIN transactions t
43     ON c.customer_id = t.customer_id
44 WHERE t.Transaction_date >= DATE_SUB(CURDATE(), INTERVAL 1 MONTH);
45
46
47
48

```

Result Grid

Export:

Wrap Cell Content:

	customer_id	customer_name	email	Transaction_date
▶	2	Emily Norton	Emily Norton@mail.com	2026-04-28
	5	Allbert	Allbert@mail.com	2026-05-02
	3	Jared Quinn	Jared Quinn@mail.com	2026-04-30
	4	jesse campbel	jesse campbel@mail.com	2026-05-06
	1	albert	John@mail.com	2026-05-08

Result 5

×

b) Insert Data into Tables

- Add a new customer and product.

[10]

- Insert a new transaction record involving both.

ANSWER: (Upload screenshot if necessary)

```

22 • select * from Products;
23 • insert into products (product_Name,Category,price,stock)
24 values
25 ("youM","apples",41.41,98),
26 ("Thatshop","tomatoes", 313.07,15),
27 ("SuuplierFoods","pepsi",432.24,17),
28 ("BestItems","Books", 292.4,18),
29 ("Buynow","spinache", 355.245,19);
30
31 • INSERT INTO products (product_Name,Category,price,stock)
32 VALUES
33 ("youM","apples",416.4112,98),
34 ("Thatshop","tomatoes",313.07,15),

```

Result Grid | Filter Rows: | Edit: | Export/Import:

	product_id	product_name	Category	price	stock
	20	Buynow	spinache	355.25	19
	21	youM	apples	41.41	98
	22	Thatshop	tomatoes	313.07	15
	23	SuuplierFoods	pepsi	432.24	17
	24	BestItems	Books	292.40	18

Products 16 x

```

64 • Insert into customers (Customer_name,Email,Phone,Join_date) values
65 ("John","HJohn5@gmail.com","0666554776",curdate());
66 • select * from customers where Join_date = curdate();
67

```

Result Grid | Filter Rows: | Edit: | Export/Import:

	customer_id	customer_name	email	PHONE	Join_date
	6	John	HJohn5@gmail.com	0666554776	2026-05-08
	NULL	NULL	NULL	NULL	NULL

```

53 • Insert into transactions ( Customer_ID,Product_ID, Quintity,Transaction_date) values
54 (6,20,4,curdate());
55
56 • select * from Transactions;

```

Output

#	Time	Action	Message
✓ 67	19:34:42	select * from Products LIMIT 0, 1000	25 row(s) returned

c)	Delete Data from Tables - Remove discontinued product. - Demonstrate cascade/delete protection logic.	[10]
	ANSWER: (Upload screenshot if necessary) 	
d)	Update Data in Tables - Update stock levels post-transaction. - Modify a customer's contact details.	[10]
	ANSWER: (Upload screenshot if necessary)	

	89 90 • <code>Update products set stock= 30 where Product_id = 2;</code> 91 92 93 94 Output Action Output <table><tr><th>#</th><th>Time</th><th>Action</th></tr><tr><td>✓ 72</td><td>19:45:28</td><td>Update products set stock= 30 where Product_id = 2</td></tr></table> 89 90 • <code>Update Customers set email= "bjames@gmail.com",phone = 0178765430 where Customer_id = 5;</code> 91 92 93 94 95 Output Action Output <table><tr><th>#</th><th>Time</th><th>Action</th></tr><tr><td>✓ 76</td><td>19:48:01</td><td>Update Customers set email= "bjames@gmail.com",phone = 0178765430 where Customer_id = 5</td></tr></table>	#	Time	Action	✓ 72	19:45:28	Update products set stock= 30 where Product_id = 2	#	Time	Action	✓ 76	19:48:01	Update Customers set email= "bjames@gmail.com",phone = 0178765430 where Customer_id = 5	
#	Time	Action												
✓ 72	19:45:28	Update products set stock= 30 where Product_id = 2												
#	Time	Action												
✓ 76	19:48:01	Update Customers set email= "bjames@gmail.com",phone = 0178765430 where Customer_id = 5												
e)	Join Tables Together using “INNERJOIN” Display all transactions with customer names and product names.	[10]												
	ANSWER: (Upload screenshot if necessary)													

```

93
94 • SELECT
95     t.Transaction_id,
96     c.customer_name,
97     p.product_name,
98     t.Quantity,
99     t.Transaction_date
100 FROM Transactions t
101 INNER JOIN Customers c ON t.customer_id = c.customer_id
102 INNER JOIN Products p ON t.Product_id = p.Product_id;
103
104

```

Result Grid  Filter Rows: Export:  Wrap Cell Content: 

	Transaction_id	customer_name	product_name	Quantity	Transaction_date
▶	4406f784-cd8f-4e04-9dbe-af1b39b8a311	Emily Norton	Potatoes	5	2026-04-28
	71f2e951-a383-4ee3-bea5-f6a068c8f552	Allbert	Coke	3	2026-05-02
	82ebf96d-1a95-45a7-9660-a960ea960b64	Jared Quinn	Red Bull	6	2026-04-30
	a9487e11-9413-43da-b86c-0d2e5daf6aff	jesse campbel	Glass	6	2026-05-06

Mark allocation for student

Question	Maximum Mark	Learner mark
Question 1.a	10	
Question 1.b	15	
Question 1.c	15	
Question 1.d	10	

Question 2.a	10	
Question 2.b	10	
Question 2.c	10	
Question 2.d	10	
Question 2.e	10	
Total:	75	

STUDENT ASSESSMENT FEEDBACK

Qualification Name:	OCTO: Software Engineer
Qualification SAQA Number:	119458
Subject Name:	Design and Manipulate Databases
Subject Code:	DEMD
Assessment Name:	Summative Assessment

Assessment Code:	DEMD_SA
-------------------------	---------

Assessment Decision	Competent		Not yet competent	
----------------------------	------------------	--	--------------------------	--

Feedback report	1st Attempt		2nd Attempt	
	C	NYC	C	NYC
Practical module/Practical skill				
PM02				
PS01				
PS02				

General feedback to learner
Supply comprehensive feedback why learner is found NYC