

# **AQUA WATER REFILLING MANAGEMENT SYSTEM**

**A Database Design**

**In Database Management System 1**

**Presented to the Faculty of CHMSC ITE Department**

**In Partial Fulfillment**

**Of the Requirements for the**

**Bachelor of Science in information Technology 1**

**GLADYS T. ELIVER**

**March 2019**

## TABLE OF CONTENTS

Title Page .....	i
Table of contents .....	ii
System Description .....	1
Entity Relationship Diagram .....	2
Database Dictionaries .....	3-5
Select Statements using Different Functions .....	6-11
Select Statements using Sub Query .....	12
Select Statements using Count and Group Functions .....	13
Select Statements using Different Joins .....	14-16
Insert Statement .....	17-21
Update Statement .....	22-26
Delete Statement .....	27-28
Sample Records .....	29-31

## SYSTEM DESCRIPTION

One of the fastest and expanding businesses today is water refilling station. Water refilling station is small water system that has its own water purification facility producing a portable drinking water. The aqua water refilling system has their own water tank and equipment that intend on their business.

So, we design a system in this kind of business in order to be on top and align on the fast-growing business that is demand now a day.

The purpose of water refilling management system is to overcome difficulties in manual operation in refilling station. The difficulty in manual system are one of the reasons why the efficiency in availing services of the clients is not satisfying and keeping of records is often misplaced and not secure.

This system will be programmed to java that can enable the user to record things that are being purchased by the clients and it be created using MySQL database.

This system manages to display the data to be filled by the user according to the information of the customer in organize manner, such that their personal details, and the services they want to avail as well as the payment on the transaction they purchased. The system keeps the information of the customer and the details of what they purchased.

The system coordinates the arrangement on delivery of products. It consists all the records for the location of the clients, date of transaction, schedule of delivery, contact number and the person assign to deliver and the payment of customer to the quantity of product that about to deliver.

The system also views the information about the availability of the products as well as the containers. The system views the available containers to provide stocks again.

This system also manages the information of the employees that a refilling station must have just like front liner, cashier, technical assistant, and delivery an. It stores the information in organize so that it easy to the owner to access on the detail of his/her employee.

Upon having this system, it will provide the capacity to the owner and clients to transact without spending time and effort.

Basic features:

- Manage user (add, list updates)
- Delivery (confirm, cancel)
- Products (add, delete, update).

## ENTITY RELATIONSHIP DIAGRAM

Figure 1 Proposed water refilling management system Entity Relationship Diagram shows the system entity relationship in each entity and their supposed function in each relationship.

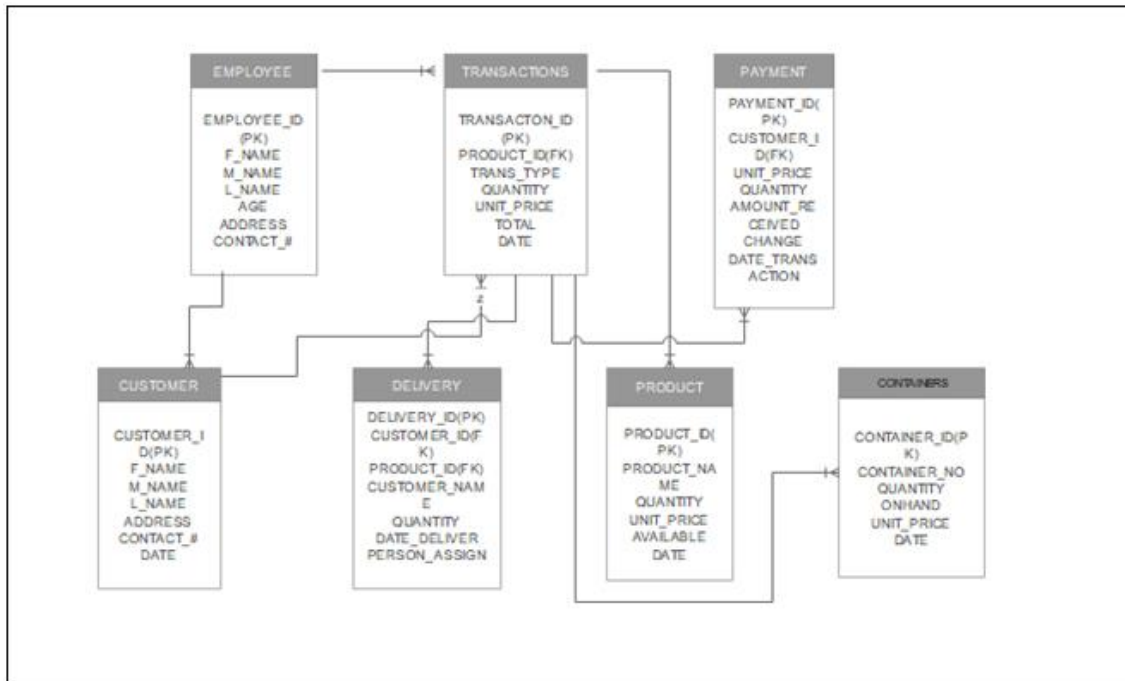


Figure 1. Aqua Water Refilling Management System Entity Relationship Diagram

Based on figure 1, the proposed aqua water refilling management system entity relationship diagram are the entity of the proposed water refilling management system database, which are represented by tables, the tables are made to meet the required specification of the system and provide a much specific details of the each entities within the system.

## DATA DICTIONARIES

The table below provide the entire database tables details such as field names, descriptions, data type and character length.

TABLE 1. EMPLOYEE

FIELD NAMES	DESCRIPTION	TYPE	LENGTH
EMPLOYEE_ID(pk)	EMPLOYEE ID NUMBER	INT	11
F_NAME	EMPLOYEE FIRST NAME	VARCHAR	50
M_NAME	EMPLOYEE MIDDLE NAME	VARCHAR	50
L_NAME	EMPLOYEE LAST NAME	VARCHAR	50
AGE	AGE OF EMPLOYEE	VARCHAR	11
ADDRESS	LOCATION OF EMPLOYEE	VARCHAR	50
CONTACT_#	NUMBER OF EMPLOYEE	VARCHAR	11

Table 2. Customer

FIELD NAMES	DESCRIPTION	TYPE	LENGTH
CUSTOMER_ID(PK)	CUSTOMER ID NUMBER	INT	11
F_NAME	CUSTOMER FIRST NAME	VARCHAR	50
M_NAME	CUSTOMER MIDDLE NAME	VARCHAR	50
L_NAME	CUSTOMER LAST NAME	VARCHAR	50
ADDRESS	LOCATION OF CUSTOMER	VARCHAR	50
CONTACT_#	NUMBER OF CUSTOMER	VARCHAR	11
DATE	DATE OF TRANSACTION	DATE TIME	11

TABLE 3. PRODUCT

FIELD NAMES	DESCRIPTION	TYPE	LENGTH
PRODUCT_ID(PK)	PRODUCT ID NUMBER	INT	11
PRODUCT_NAME	PRODUCT NAME	VARCHAR	50
QUANTITY	QUANTITY OF PRODUCTS	INT	11
UNIT_PRICE	PRICE OF PRODUCTS	VARCHAR	50
DATE	DATE	DATE	11

TABLE 4. DELIVERY

FIELD NAMES	DESCRIPTION	TYPE	LENGTH
DELIVERY_ID(PK)	DEELIVERY ID NUMBER	INT	11
CUSTOMER_ID(FK)	CUSTOMER ID NUMBER	INT	11
PRODUCT_ID(FK)	PRODUCT ID NUMBER	INT	11
TRANSACTION_ID	TRANSACTION ID NUMBER	INT	11
QUANTITY	QUANTITY OF PRODUCTS	INT	11
DELIVERY_MAN	DELIVERY MAN	VARCHAR	50
DATE_DELIVER	DATE TO DELIVER	DATE TIME	11

TABLE5.CONTAINERS

FIELD NAMES	DESCRIPTION	TYPE	LENGTH
CONTAINER_ID	CONTAINER ID NUMBER	INT	11
CONTAINER_NO	CONTROL NUMBER	INT	50
QUANTITY	QUANTITY ORDER OF CUSTOMER	INT	20
ON_HAND	AVAILABLE CONTAINERS	VARCHAR	50
UNIT_PRICE	PRICE OF CONTAINERS	VARCHAR	11

TABLE 6.PAYMENT

FIELD NAMES	DESCRIPTION	TYPE	LENGTH
PAYMENT_ID	PRODUCT ID NUMBER	INT	11
CUSTOMER_ID	CUSTOMER ID NUMBER	INT	50
TRANSACTION_ID	DATE OF TRANSACTION	DATE TIME	11
TOTAL	TOTAL OFPURCHASED PRODUCT	INT	11
AMOUNT_RECEIVED	AMOUNT RECEIVED	INT	11
CHANGED	CHANGED	INT	11

TABLE 7.TRANSACTION

FIELD NAMES	DESCRIPTION	TYPE	LENGTH
TRANSACTION_ID	PRODUCT ID NUMBER	INT	11
PRODUCT_ID	PRODUCT ID NUMBER	INT	50
TRANS_TYPE	TYPE OF TRANSACTION	VARCHAR	11
UNIT_PRICE	PRICE OF PRODUCT	VARCHAR	50
QUANTITY	QUANTITY OF PRODUCT	INT	11
TOTAL	TOTAL COST OF TRASACTION	INT	11
DATE	DATE	DATE	11

## SELECT STATEMENT USING DIFFERENT FUNCTIONS

1. Create a query that display the complete names of all customer.

a. Solution

```
SELECT CONCAT (L_NAME,' ', F_NAME ) " CUSTOMER FULL NAMES"
FROM `customer`;
```

b. Result

CUSTOMER FULL NAMES
magdalena maria
antonio rex
ten ben
grey chris
salde nomer
romano japit
notar hazel

15 Result(s) found.

2. Create a query that display all the name of products.

a. Solution

```
SELECT DISTINCT(product_name) as "Products" FROM `product`;
```

b. Result

Products
alkaline
mineral
PURIFIED

3 result(s) found.

3. Show all the total number of each container.

a. Solution

```
SELECT sum(quantity) as "Total number of all
container",CONTAINER_NAME FROM `containers` GROUP by
CONTAINER_NAME;
```

b. Result

Total number of all container	CONTAINER_NAME
13	gallon with faucet
10	round container
4	round gallon

3 result(s) found.

4. Create a query that display the sum of the ALKALINE products.

a. Solution

```
SELECT SUM(QUANTITY) "total product",Product_name"PRODUCT NAME"
FROM product WHERE Product_name='ALKALINE' ;
```

b. Result

total product	PRODUCT NAME
21	alkaline

1 result(s) found

5. Show all the list of customer who transact in the month February 28.

a. Solution



```
SELECT L_NAME "customer Last names", length (l_name)" last names
length" FROM customer;
```

b. Result

CUSTOMER LASTNAME	DATE
magdalena	2019-02-28
antonio	2019-02-28
ten	2019-02-28
grey	2019-02-28

6 result(s) received

6. Write a query that display the maximum amount received.

a. Solution

```
SELECT MAX(AMOUNT_RECEIVED) " maximum amount received" FROM
`payment`;
```

b. Result

maximum amount received
1000

1 result(s) received

7. Write a query that display the list of customer transaction details that have balance more than 100.

a. Solution

```
SELECT transaction_id "transaction id",PRODUCT_ID"product
id",QUANTITY,TRANS_TYPE"transaction type",UNIT_PRICE"unit
price",TOTAL FROM `transaction` WHERE TOTAL > 100;
```

b. Result

transaction id	product id	QUANTITY	transaction type	unit price	TOTAL
10101	1001	12	REFILL	25	300
10103	1003	2	WATER WITH GALLON	150	300
10108	1008	3	WATER WITH GALLON	150	450
10109	1009	2	WATER WITH GALLON	150	300
10111	1011	1	WATER WITH GALLON	150	150
10112	1012	5	refill	25	125
10113	1012	5	refill	25	125

7 result(s) received

8. For each transaction display the transaction type and calculate the total number of customer choose the transaction.

a. Solution

```
SELECT TRANS_TYPE "transaction type", count(TRANS_TYPE)" total of
customer " FROM `transaction` GROUP by TRANS_TYPE;
```

b. Result

transaction type	total of customer
REFILL	8
WATER WITH GALLON	4

2 result(s) found

9. Show all the number of customer who transact in each address.

a. Solution

```
SELECT COUNT(ADDRESS) "number of customer", ADDRESS "customer address" FROM customer GROUP BY ADDRESS;
```

b. Result

number of customer	customer address
1	ad
3	aguisan
1	Brgy.4
1	Brgy.Carabalan
1	cabanbanan

12 result(s) found

10. Display each transaction type, date of transaction, and date of the week on which the transaction done. Label the column DAY. Order the results by the day of the week.

a. Solution

```
SELECT TRANSACTION_ID"transaction number",TRANS_TYPE"transaction type",`DATE`"date of transaction",dayname(date)"Day" FROM transaction ORDER by weekday(date);
```

b. Result

transaction number	transaction type	date of transaction	Day
10101	REFILL	2019-02-28 18:39:15	Thursday
10111	WATER WITH GALLON	2019-02-28 19:23:28	Thursday
10110	REFILL	2019-02-28 19:01:58	Thursday
10109	WATER WITH GALLON	2019-02-28 19:00:38	Thursday
10108	WATER WITH GALLON	2019-02-28 19:00:19	Thursday
10107	REFILL	2019-02-28 18:59:38	Thursday
10106	REFILL	2019-02-28 18:59:05	Thursday

8 result(s) found

11. Write a query that display product name in lowercase and total number of each product.

a. Solution

```
SELECT lower(Product_name)"product name",COUNT(Product_name)"number of products" FROM `product`GROUP by Product_name;
```

b. Result

product name	number of products
alkaline	7
mineral	8
purified	5

1 result(s) received

12. Write a query that display the total number of transaction.

a. Solution

```
SELECT count(*) "total number of transaction" FROM `transaction`;
```

b. Result

total number of transaction
12

1 result(s) found.

13. Create a query that display the date and amount received.

a. Solution

```
SELECT AMOUNT_RECEIVED "amount received", DATE FROM `payment`;
```

b. Result

amount received	DATE
200	2019-02-28 19:12:36
500	2019-02-28 19:06:57
1000	2019-02-28 19:13:37
25	2019-02-28 19:13:42
50	2019-02-28 19:13:47
100	2019-02-28 19:13:53

15 result(s) found.

14. Show all employee full names and contact numbers.

a. Solution

```
SELECT concat(F_NAME,' ',LAST_NAME) "employee full name", CONTACT_NUM FROM employee;
```

b. Result

employee full name	CONTACT_NUM
BRUNO MARS	09369296777
aurelio laja	09126534451
wilmae camanso	09467873287
carl marcos	09369296755
MARIO tanio	09461263436

3 result(s) found.

15. Show the maximum products.

a. Solution

```
SELECT avg(quantity) "average of container availability" FROM product;
```

b. Result

average of container availability
1.6190

1 result(s) found.

16. For each transaction, display the transaction number, product id, unit price, and total amount to be paid discounted by 1% and expressed as a whole number.

a. Solution

```
SELECT transaction_ID as "transaction Number", PRODUCT_ID "product number", UNIT_PRICE "unit price", total, total- ROUND(total*.1, 0) AS "total discounted by 1%" FROM `transaction`;
```

b. Result

transaction Number	product Id	unit price	total	total discounted by 1%
10101	1001	25	300	270
10103	1003	150	300	270
10104	1004	25	25	22
10105	1005	25	25	22
10106	1006	25	25	22
10107	1007	25	25	22
10108	1008	150	450	405

12 result(s) found.

17. Create a query that display the average of amount tender by the customer.

a. Solution

```
SELECT avg(amount_received) " average of amount received " FROM
payment;
```

b. Result

average of amount received
266.3636

1 result(s) found.

18. Write a query that display the customer last names starting with M, R, T, and it's length.

a. Solution

```
SELECT upper(L_NAME) AS "customer last name",LENGTH (L_NAME)
"LENGTH OF NAME" FROM customer WHERE (L_NAME) LIKE 'r%' OR
L_NAME LIKE 't%'OR L_NAME LIKE 'M%' ORDER BY L_NAME;
```

b. Result

customer last name	LENGTH OF NAME
MAGDALENA	9
ROMANO	6
TALAMAN	7
TEN	3
TENIDO	6
TRINIO	6

6 result(s) found.

19. Display customer last name, first name and date of transaction who transact January 01, 2019 up to March 28, 2019.

a. Solution:

```
SELECT L_NAME as "Last Name", F_NAME as " First Name", DATE as " transact
Date" FROM `customer` WHERE DATE BETWEEN '2019-01-12'AND'2019-03-
28';
```

b. Result

Last Name	First Name	transact Date
magdalena	maria	2019-02-28
antonio	rex	2019-02-28
ten	ben	2019-02-28
grey	chris	2019-02-28
salde	nomer	2019-02-28
romano	japit	2019-02-28
notar	hazel	2019-02-21
verde	green	2019-02-20
trinio	blue	2019-02-27
blauro	john	2019-02-15
tenido	jude	2019-02-18
talaman	mark	2019-02-27
caro	john	2019-02-26
belbar	ralph	2019-02-26
Tenido	Jason	2019-03-07
Hilado	Carlos	2019-03-07

21 result(s) found

20. Display the total of amount received.

a. Solution

```
SELECT sum(AMOUNT_RECEIVED) "total amount received" FROM
`payment`;
```

b. Result

total amount received
2930

1 result(s) found

## Select Statements using Sub Query

1. Write a query that display the product name, price and the product that has many number of availability .

a. Solution

```
SELECT Product_name "Product name",Unit_price"unit price" FROM product
WHERE Product_name in (SELECT max(Product_name) FROM
product)GROUP by Product_name;
```

b. Result

Product name	unit price
PURIFIED	45

1 result(s) found.

2. Write a query that display the customer last name, first name who lived in Talaban.

a. Solution

```
SELECT L_NAME "LAST NAME",F_NAME"FIRST NAME" FROM `customer`
WHERE ADDRESS IN (SELECT ADDRESS FROM customer WHERE
ADDRESS='TALABAN');
```

b. Result

LAST NAME	FIRST NAME
magdalena	maria
antonio	rex
ten	ben
talaman	mark

4 Result(s) found

3. Write a query that display the last name contact number of employees who lived in Binalbagan.

a. Solution

```
SELECT LAST_NAME "LAST NAME", CONTACT_NUM " employee contact
number" FROM `employee` WHERE ADDRESS IN (SELECT ADDRESS FROM
employee WHERE ADDRESS='Binalbagan');
```

b. Result

LAST NAME	employee contact number
laja	09126534451
tanio	09461263436

2 Result(s) found.

## Select Statements using Count and Group Functions

1. Display the name and the total number of containers.

a. Solution

```
SELECT COUNT(CONTAINER_NAME) "total number of containers",
CONTAINER_NAME "container name" FROM containers GROUP by
CONTAINER_NAME;
```

b. Result

total number of containers	container name
2	bottle
3	gallon with faucet
4	round gallon

3 result(s) found.

2. Create a query that display the number of customer that transact on the address in the database.

a. Solution

```
SELECT COUNT(L_NAME)"total number of customer",ADDRESS FROM
`customer` GROUP by ADDRESS;
```

b. Result

total number of customer	ADDRESS
3	aguisan
1	cabanbanan
1	carabalan
2	mambagaton
1	menez
2	saraet
4	talaban
1	tuoy

7 result(s) found.

## SELECT STATEMENT USING DIFFERENT JOINS

1. Create a query that display the customer full name, total cost of purchased product, the amount tender and the changed of the customer.

a. Solution

```
SELECT concat(customer.L_NAME," ",customer.F_NAME) "customer full name",payment.TOTAL_COST"totalcost",payment.AMOUNT_RECEIVED "amount render",payment.CHANGED FROM customer INNER JOIN payment on customer.CUSTOMER_ID=payment.CUSTOMER_ID;
```

b. Result

customer full name	totalcost	amount render	CHANGED
antonio rex	150	200	50
magdalena maria	300	500	200
ten ben	300	1000	800
grey chris	25	25	0
salde nomer	25	50	25

9 result(s) found.

2. Show the list of product that is purchased through refill.

a. Solution

```
SELECT product.product_name"product name",transaction.TRANS_TYPE"transaction type" FROM product INNER JOIN transaction on product.product_id=transaction.PRODUCT_ID WHERE trans_type = "refill";
```

b. Result

product name	transaction type
alkaline	REFILL
MINERAL	REFILL
PURIFIED	REFILL
ALKALINE	REFILL
ALKALINE	REFILL
PURIFIED	REFILL

6 result(s) found.

3. Show the list of customer number ,last name, and the employee first name.

a. Solution

```
SELECT customer.CUSTOMER_ID AS "customer number", customer.L_NAME AS " Customer Last Name", employee.F_NAME AS " Employee First Name" FROM customer LEFT JOIN employee ON
```



customer.CUSTOMER\_ID = employee.EMPLOYEE\_ID ORDER BY  
customer.CUSTOMER\_ID;

b. Result

customer number	Customer Last Name	Employee First Name
101	magdalena	NULL
102	antonio	NULL
103	ten	NULL
104	grey	NULL
105	salde	NULL
106	romano	NULL
107	notar	NULL
108	verde	NULL
109	trinio	NULL
110	blauro	NULL
111	tenido	may
112	talaman	NULL
113	paterno	NULL
114	caro	NULL
115	belbar	NULL
116	Tenido	NULL
117	Hilado	NULL

17 result(s) found.

4. Write a query that will display the delivery id, customer complete name in upper case format. Show also the list of delivery man and the date when the product will be deliver.

a. Solution

```
SELECT Upper(concat( customer.L_NAME,"
",customer.F_NAME))"Customer
Name",delivery.DELIVERY_MAN"delivery
man",delivery.DATE,delivery.DELIVERY_ID FROM customer RIGHT
JOIN delivery ON customer.CUSTOMER_ID=delivery.CUSTOMER_ID
ORDER BY customer.L_NAME;
```

b. Result

+ Options			
Customer Name	delivery man	DATE	DELIVERY_ID
ANTONIO REX	MARIO TANIO	2019-02-28 19:29:36	202
BLAURO JOHN	BRUNO MARS	2019-02-28 19:31:06	209
MAGDALENA MARIA	MARIO TANIO	2019-02-28 19:24:54	200
NOTAR HAZEL	CARL MARCOS	2019-02-28 19:30:08	206
ROMANO JAPIT	CARL MARCOS	2019-02-28 19:29:59	205
SALDE NOMER	BRUNO MARS	2019-02-28 19:29:51	204
TEN BEN	MARIO TANIO	2019-02-28 19:29:28	201
TEN BEN	MARIO TANIO	2019-02-28 19:29:43	203
TENIDO JUDE	BRUNO MARS	2019-02-28 19:31:23	210
TRINIO BLUE	MARIO TANIO	2019-02-28 19:30:36	208
VERDE GREEN	BRUNO MARS	2019-02-28 19:30:17	207

11 result(s) found.

5. Display the product name, transaction type and the total cost.

a. Solution

```
SELECT product.`product_NAME` AS "product Name",
transaction.`trans_type` AS "transactio type",TOTAL"TOTAL COST" FROM
`product` JOIN transaction
on(transaction.`product_ID`=product.`product_ID`);
```

b. Result

product Name	transactio type	TOTAL COST
alkaline	REFILL	300
MINERAL	WATER WITH GALLON	300
MINERAL	REFILL	25
PURIFIED	REFILL	25
ALKALINE	REFILL	25
ALKALINE	WATER WITH GALLON	450
PURIFIED	WATER WITH GALLON	300
PURIFIED	REFILL	25
PURIFIED	WATER WITH GALLON	150
PURIFIED	refill	125
PURIFIED	refill	125

11 result(s) found.

### Insert Statement

1. Create a query to insert 10 new alkaline product and its price is 25 pesos in March 23 2019.

product_id	Product_name	Quantity	Unit_price	Date
1017	MINERAL	1	33	2019-03-14
1018	ALKALINE	1	25	2019-03-15
1019	ALKALINE	1	25	2019-03-14
1020	ALKALINE	1	25	2019-03-14
1021	alkaline	5	25	2019-03-14
1023	alkaline	10	25	2019-03-09

- a. Solution

```
INSERT INTO `product` (`product_id`, `Product_name`, `Quantity`,  
`Unit_price`) VALUES ('1024', 'alkaline', '10', '25');
```

- b. Result

product_id	Product_name	Quantity	Unit_price	Date
1018	ALKALINE	1	25	2019-03-15
1019	ALKALINE	1	25	2019-03-14
1020	ALKALINE	1	25	2019-03-14
1021	alkaline	5	25	2019-03-14
1023	alkaline	10	25	2019-03-09
1024	alkaline	10	25	2019-03-23

1 row(s) inserted.

2. Create a query to insert new customer whose named is Jason Tenido living in Brgy.4 and has contact number of 09122345757 while Carlos Hilado lived in Brgy.Carabalan and has contactnumber of 09367281123.

CUSTOMER_ID	F_NAME	L_NAME	ADDRESS	CONTACT_NUM	DATE
101	maria	magdalena	talaban	09123556467	2019-02-28
102	rex	antonio	talaban	09126746426	2019-02-28
103	ben	ten	talaban	09121321677	2019-02-28
104	chris	grey	aguisan	09461243372	2019-02-28
105	nomer	salde	aguisan	09234556776	2019-02-28
106	japit	romano	mambagaton	09751234432	2019-02-28
107	hazel	notar	mambagaton	09234534567	2019-02-21
108	green	verde	aguisan	09128654345	2019-02-20
109	blue	trinio	saraet	09123456775	2019-02-27
110	john	blauro	tuoy	09361234568	2019-02-15
111	jude	tenido	cabanbanan	09341234566	2019-02-18
112	mark	talaman	talaban	09368787217	2019-02-27
114	john	caro	saraet	09127654456	2019-02-26
115	ralph	belbar	menez	09361233456	2019-02-26

a. Solution

```
INSERT INTO `customer` (`CUSTOMER_ID`, `F_NAME`, `L_NAME`, `ADDRESS`,
`CONTACT_NUM`, `DATE`) VALUES ('116', 'Jason', 'Tenido', 'Brgy.4',
'09122345757', CURRENT_TIMESTAMP), ('117', 'Carlos', 'Hilado',
'Brgy.Carabalan', '09367281123', CURRENT_TIMESTAMP);
```

b. Result

CUSTOMER_ID	F_NAME	L_NAME	ADDRESS	CONTACT_NUM	DATE
113	kim	paterno	carabalan	09128634566	2019-02-23 18:04:39
114	john	caro	saraet	09127654456	2019-02-26 18:05:17
115	ralph	belbar	menez	09361233456	2019-02-26 18:06:02
116	Jason	Tenido	Brgy.4	09122345757	2019-03-07 17:03:45
117	Carlos	Hilado	Brgy.Carabalan	09367281123	2019-03-07 17:03:45

2 record(s) inserted.

3. Create a query to insert new employee whose name is roel gecosala 27 years old who lived in bgry.enclaro and has a contact number of 09214567999.

EMPLOYEE_ID	F_NAME	LAST_NAME	AGE	ADDRESS	CONTACT_NUM
50	aurelio	laja	21	binalbagan	09126534451
51	wilmae	camanso	24	kabankalan	09467873287
52	carl	marcos	27	himamaylan	09369296755
53	MARIO	tanio	30	binalbagan	09461263436
54	Noel	Dalagan	30	Enclaro	09214567999

- a. Solution

```
INSERT INTO `employee` (`EMPLOYEE_ID`, `F_NAME`, `LAST_NAME`,
`AGE`, `ADDRESS`, `CONTACT_NUM`) VALUES ('55', 'Roel', 'Gecosala',
'27', 'BRGY.ENCLARO', '09214567999');
```

- b. Result

EMPLOYEE_ID	F_NAME	LAST_NAME	AGE	ADDRESS	CONTACT_NUM
5	BRUNO	MARS	26	HIMAMAYLAN	09369296777
50	aurelio	laja	21	binalbagan	09126534451
51	wilmae	camanso	24	kabankalan	09467873287
52	carl	marcos	27	himamaylan	09369296755
53	MARIO	tanio	30	binalbagan	09461263436
54	Noel		0		
55	Roel	Gecosala	27	BRGY.ENCLARO	09214567999

2 record(s) inserted.

4. Create a query to insert new 10 round container and cost of 125.

CONTAINER_ID	CONTAINER_NAME	QUANTITY	UNIT_PRICE	DATE
1	round gallon	1	125	2019-01-28
3	round gallon	1	125	2019-02-28
4	round gallon	1	125	2019-03-09
5	round gallon	1	125	2019-03-11
6	gallon with faucet	1	125	2019-02-14
7	gallon with faucet	1	125	2019-02-13
8	Gallon with faucet	1	125	2019-02-12

a. Solution

```
INSERT INTO `containers` (`CONTAINER_ID`, `CONTAINER_NAME`,
`QUANTITY`, `UNIT_PRICE`) VALUES ('11', 'round container', '10', '125');
```

b. Result

CONTAINER_ID	CONTAINER_NAME	QUANTITY	UNIT_PRICE
1	round gallon	1	125
3	round gallon	1	125
4	round gallon	1	125
5	round gallon	1	125
6	gallon with faucet	1	125
7	gallon with faucet	1	125
8	Gallon with faucet	1	125
9	bottle	1	75
10	bottle	1	75
11	round container	10	125

1 Record(s) inserted.

5. Create a query to insert new transaction such that transasction id is 10112, product id is 1012 ,quantity is 5, transaction type is refill, price is 25 and has a total cost of 125.

TRANSACTION_ID	PRODUCT_ID	QUANTITY	Product_name	TRANS_TYPE	UNIT_PRICE	TOTAL	DATE
10101	1001	12		REFILL	25	300	2019-02-28 18:39:1
10103	1003	2		WATER WITH GALLON	150	300	2019-02-28 18:51:4
10104	1004	1		REFILL	25	25	2019-02-28 18:53:0
10105	1005	1		REFILL	25	25	2019-02-28 18:57:4
10106	1006	2		REFILL	25	25	2019-02-28 18:59:0
10107	1007	1		REFILL	25	25	2019-02-28 18:59:3
10108	1008	3		WATER WITH GALLON	150	450	2019-02-28 19:00:1
10109	1009	2		WATER WITH GALLON	150	300	2019-02-28 19:00:3
10110	1010	1		REFILL	25	25	2019-02-28 19:01:5
10111	1011	1		WATER WITH GALLON	150	150	2019-02-28 19:23:2

a. Solution

```
INSERT INTO `transaction` (`TRANSACTION_ID`, `PRODUCT_ID`,
`QUANTITY`, `TRANS_TYPE`, `UNIT_PRICE`, `TOTAL`, `DATE`) VALUES
('10112', '1012', '5', 'refill', '25', '125', CURRENT_TIMESTAMP);
```

b. Result

TRANSACTION_ID	PRODUCT_ID	QUANTITY	TRANS_TYPE	UNIT_PRICE	TOTAL	DATE
10101	1001	12	REFILL	25	300	2019-02-28 18:39:15
10102	1002	1	WATER WITH GALLON	150	150	2019-02-28 18:48:56
10103	1003	2	WATER WITH GALLON	150	300	2019-02-28 18:51:43
10104	1004	1	REFILL	25	25	2019-02-28 18:53:07
10105	1005	1	REFILL	25	25	2019-02-28 18:57:45
10106	1006	2	REFILL	25	25	2019-02-28 18:59:05
10107	1007	1	REFILL	25	25	2019-02-28 18:59:38
10108	1008	3	WATER WITH GALLON	150	450	2019-02-28 19:00:19
10109	1009	2	WATER WITH GALLON	150	300	2019-02-28 19:00:38
10110	1010	1	REFILL	25	25	2019-02-28 19:01:58
10111	1011	1	WATER WITH GALLON	150	150	2019-02-28 19:23:28
10112	1012	5	refill	25	125	2019-03-08 19:37:33
10113	1012	5	refill	25	125	2019-03-08 19:39:29

2 record(s) inserted.

## Update Statement

1. Create a query to change the last name of noel dalagan to delagon.

EMPLOYEE_ID	F_NAME	LAST_NAME	AGE	ADDRESS	CONTACT_NUM
5	BRUNO	MARS	26	HIMAMAYLAN	09369296777
50	aurelio	laja	21	binalbagan	09126534451
51	wilmae	camanso	24	kabankalan	09467873287
52	carl	marcos	27	himamaylan	09369296755
53	MARIO	tanio	30	binalbagan	09461263436
54	Noel	Dalagan	30	Enclaro	09214567999
55	Roel	Gecosala	27	BRGY.ENCLARO	09214567999
111	May	June	27	Talaban	09214567999

- a. Solution

```
UPDATE `employee` SET `LAST_NAME` = 'Delagon', `WHERE`  
`employee`.`EMPLOYEE_ID` = 54;
```

- b. Result

EMPLOYEE_ID	F_NAME	LAST_NAME	AGE	ADDRESS	CONTACT_NUM
5	BRUNO	MARS	26	HIMAMAYLAN	09369296777
50	aurelio	laja	21	binalbagan	09126534451
51	wilmae	camanso	24	kabankalan	09467873287
52	carl	marcos	27	himamaylan	09369296755
53	MARIO	tanio	30	binalbagan	09461263436
54	Noel	Delagon	30	Enclaro	09214567999
55	Roel	Gecosala	27	BRGY.ENCLARO	09214567999
111	may	june	27	talaban	09214567999

1 Record(s) updated.



2. Create a query to change the information on the product id 1022 .

product_id	Product_name	Quantity	Unit_price
1015	MINERAL	1	35
1016	MINERAL	1	35
1017	MINERAL	1	35
1018	ALKALINE	1	25
1019	ALKALINE	1	25
1020	ALKALINE	1	25
1021	alkaline	5	25
1022		0	0
1023	alkaline	10	25

- a. Solution

```
UPDATE `product` SET `Product_name` = 'purified', `Quantity` = '2',  
`Unit_price` = '45' WHERE `product`.`product_id` = 1022;
```

- b. Result

product_id	Product_name	Quantity	Unit_price
1015	MINERAL	1	35
1016	MINERAL	1	35
1017	MINERAL	1	35
1018	ALKALINE	1	25
1019	ALKALINE	1	25
1020	ALKALINE	1	25
1021	alkaline	5	25
1022	purified	2	45
1023	alkaline	10	25

1 record(s) updated.

3. Write a query to change the amount render by the customer having the payment id of 110A.

PAYMENT_ID	CUSTOMER_ID	TRANSACTION_ID	TOTAL_COST	AMOUNT_RECEIVED	CHANGED	DATE
100A	102	10102	150	200	50	2019-02-28 19:12:38
101A	101	10101	300	500	200	2019-02-28 19:06:57
102A	103	10103	300	1000	800	2019-02-28 19:13:37
103A	104	10104	25	25	0	2019-02-28 19:13:42
104A	105	10105	25	50	25	2019-02-28 19:13:47
105A	106	10106	25	100	75	2019-02-28 19:13:53
106A	107	10107	25	25	0	2019-02-28 19:13:59
107A	108	10108	450	500	50	2019-02-28 19:14:04
108A	109	10109	300	300	0	2019-02-28 19:14:09
109A	110	10110	25	30	5	2019-02-28 19:14:14
110A	111	10111	150	200	50	2019-02-28 19:14:34

a. Solution

```
UPDATE `product` SET `Product_name` = 'purified', `Quantity` = '2', `Unit_price` = '45' WHERE `product`.`product_id` = 1022;
```

b. Result

PAYMENT_ID	CUSTOMER_ID	TRANSACTION_ID	TOTAL_COST	AMOUNT_RECEIVED	CHANGED	DATE
100A	102	10102	150	200	50	2019-02-28 19:12:38
101A	101	10101	300	500	200	2019-02-28 19:06:57
102A	103	10103	300	1000	800	2019-02-28 19:13:37
103A	104	10104	25	25	0	2019-02-28 19:13:42
104A	105	10105	25	50	25	2019-02-28 19:13:47
105A	106	10106	25	100	75	2019-02-28 19:13:53
106A	107	10107	25	25	0	2019-02-28 19:13:59
107A	108	10108	450	500	50	2019-02-28 19:14:04
108A	109	10109	300	300	0	2019-02-28 19:14:09
109A	110	10110	25	30	5	2019-02-28 19:14:14
110A	111	10111	150	300	150	2019-02-28 19:14:34

1 Record(s) updated.

4. Create a query to change the contact number of the employee named May and add suffix on her name.

EMPLOYEE_ID	F_NAME	LAST_NAME	AGE	ADDRESS	CONTACT_NUM
5	BRUNO	MARS	26	HIMAMAYLAN	09369296777
50	aurelio	laja	21	binalbagan	09126534451
51	wilmae	camanso	24	kabankalan	09467873287
52	carl	marcos	27	himamaylan	09369296755
53	MARIO	tanio	30	binalbagan	09461263436
54	Noel	Delagon	30	Enclaro	09214567999
55	Roel	Gecosala	27	BRGY.ENCLARO	09214567999
111	May	June	27	Talaban	09214567999

a. Solution

```
UPDATE `employee` SET `F_NAME` = 'May', `LAST_NAME` = 'June JR.', `ADDRESS` = 'Talaban', `CONTACT_NUM` = '09367144277' WHERE `employee`.`EMPLOYEE_ID` = 111;
```

b. Result

EMPLOYEE_ID	F_NAME	LAST_NAME	AGE	ADDRESS	CONTACT_NUM
5	BRUNO	MARS	26	HIMAMAYLAN	09369296777
50	aurelio	laja	21	binalbagan	09126534451
51	wilmae	camanso	24	kabankalan	09467873287
52	carl	marcos	27	himamaylan	09369296755
53	MARIO	tanio	30	binalbagan	09461263436
54	Noel	Delagon	30	Enclaro	09214567999
55	Roel	Gecosala	27	BRGY.ENCLARO	09214567999
111	May	June JR.	27	Talaban	09367144277

1 record(s) updated

5. Create a query to change the product id and the price of the newly inserted product .

product_id	Product_name	Quantity	Unit_price
0	mineral	2	25
1001	alkaline	1	25

a. Solution

```
UPDATE `product` SET `product_id` = '1024', `Unit_price` = '35'
WHERE `product`.`product_id` = 0;
```

b. Result

product_id	Product_name	Quantity	Unit_price
1016	MINERAL	1	35
1017	MINERAL	1	35
1018	ALKALINE	1	25
1019	ALKALINE	1	25
1020	ALKALINE	1	25
1021	alkaline	5	25
1022	purified	2	45
1023	alkaline	10	25
1024	mineral	2	35

1 Record(s) updated

\

### Delete Statement

1. Create a query to delete one data in customer record who transact on February 23, 2019.

CUSTOMER_ID	F_NAME	L_NAME	ADDRESS	CONTACT_NUM	DATE
101	maria	magdalena	talaban	09123556467	2019-02-28 17:54:40
102	rex	antonio	talaban	09126746426	2019-02-28 17:55:35
103	ben	ten	talaban	09121321677	2019-02-28 17:55:51
104	chris	grey	aguisan	09461243372	2019-02-28 17:57:31
105	nomer	salde	aguisan	09234556776	2019-02-28 17:58:25
106	japit	romano	mambagaton	09751234432	2019-02-28 17:59:05
107	hazel	notar	mambagaton	09234534567	2019-02-21 17:59:50
108	green	verde	aguisan	09128654345	2019-02-20 18:00:23
109	blue	trinio	saraet	09123456775	2019-02-27 18:01:05
110	john	blauro	tuoy	09361234568	2019-02-15 01:00:00
111	jude	tenido	cabanbanan	09341234566	2019-02-18 18:02:45
112	mark	talaman	talaban	09368787217	2019-02-27 18:03:36
113	kim	paterno	carabalan	09128634566	2019-02-23 18:04:39
114	john	caro	saraet	09127654456	2019-02-26 18:05:17

a. Solution

DELETE FROM customer WHERE DATE='2019-02-23'

b. Result

1 row(s) deleted.

2. Create a query to delete one employee.

EMPLOYEE_ID	F_NAME	LAST_NAME	AGE	ADDRESS	CONTACT_NUM
5	BRUNO	MARS	26	HIMAMAYLAN	09369296777
50	aurelio	laja	21	binalbagan	09126534451
51	wilmae	camanso	24	kabankalan	09467873287
52	carl	marcos	27	himamaylan	09369296755
53	MARIO	tanio	30	binalbagan	09461263436
54	Noel	Dalagan	30	Enclaro	09214567999
55	Roel	Gecosala	27	BRGY.ENCLARO	09214567999
111	May	June	27	Talaban	09214567999

a. Solution

DELETE FROM employee where employee\_id=5;

b. Result

1 row(s) deleted

3. Write a query that will delete a one data on transaction.

TRANSACTION_ID	PRODUCT_ID	QUANTITY	TRANS_TYPE	UNIT_PRICE	TOTAL	DATE
10101	1001	12	REFILL	25	300	2019-02-28 18:39:15
10102	1002	1	WATER WITH GALLON	150	150	2019-02-28 18:48:56
10103	1003	2	WATER WITH GALLON	150	300	2019-02-28 18:51:43

a. Solution

DELETE FROM transaction WHERE PRODUCT\_ID = 1002;

b. Result

1 row(s) deleted.

4. Write a query that will remove the product has the product id of 1006.

product_id	Product_name	Quantity	Unit_price
1001	alkaline	1	25
1002	mineral	1	35
1003	MINERAL	1	35
1004	MINERAL	1	35
1005	PURIFIED	1	45
1006	ALKALINE	1	25
1007	ALKALINE	1	25

a. Solution

DELETE FROM product WHERE Product\_name=1006;

b. Result

1 row(s) deleted.

5. Remove the container named bottle .

CONTAINER_ID	CONTAINER_NAME	QUANTITY	UNIT_PRICE
1	round gallon	1	125
3	round gallon	1	125
4	round gallon	1	125
5	round gallon	1	125
6	gallon with faucet	1	125
7	gallon with faucet	1	125
8	Gallon with faucet	1	125
9	bottle	1	75
10	bottle	1	75
11	round container	10	125

a. Solution

DELETE FROM containers WHERE CONTAINER\_NAME='bottle';

b. Result

2 row(s) deleted.

## Sample Records

1. Write a query to show all the records of customers.

- a. Solution

```
SELECT * FROM `customer`;
```

- b. Result

CUSTOMER_ID	F_NAME	L_NAME	ADDRESS	CONTACT_NUM	DATE
101	maria	magdalena	talaban	09123556467	2019-02-28
102	rex	antonio	talaban	09126746426	2019-02-28
103	ben	ten	talaban	09121321677	2019-02-28
104	chris	grey	aguisan	09461243372	2019-02-28
105	nomer	salde	aguisan	09234556776	2019-02-28
106	japit	romano	mambagaton	09751234432	2019-02-28
107	hazel	notar	mambagaton	09234534567	2019-02-21
108	green	verde	aguisan	09128654345	2019-02-20
109	blue	trinio	saraet	09123456775	2019-02-27
110	john	blauro	tuoy	09361234568	2019-02-15
111	jude	tenido	cabanbanan	09341234566	2019-02-18

21 result(s) found.

2. Write a query to show all the records of containers.

- a. Solution

```
SELECT * FROM `containers`;
```

- b. Result

CONTAINER_ID	CONTAINER_NAME	QUANTITY	UNIT_PRICE	DATE
1	round gallon	1	125	2019-01-28
3	round gallon	1	125	2019-02-28
4	round gallon	1	125	2019-03-09
5	round gallon	1	125	2019-03-11
6	gallon with faucet	1	125	2019-02-14
7	gallon with faucet	1	125	2019-02-13
8	Gallon with faucet	1	125	2019-02-12
11	round container	10	125	2019-02-11
12	gallon with faucet	10	125	2019-03-09

21 result(s) found.

3. Write a query to show all the records of product.

a. Solution

SELECT \* FROM `product`;

b. Result

product_id	Product_name	Quantity	Unit_price	Date
1001	alkaline	1	25	2019-03-09
1002	mineral	1	35	2019-03-09
1003	MINERAL	1	35	2019-03-09
1004	MINERAL	1	35	2019-03-14
1005	PURIFIED	1	45	2019-03-12
1007	ALKALINE	1	25	2019-03-14
1008	ALKALINE	1	25	2019-03-13
1009	PURIFIED	1	45	2019-03-14
1010	PURIFIED	1	45	2019-03-14
1011	PURIFIED	1	45	2019-03-14
1012	PURIFIED	1	45	2019-03-14

22 result(s) found.

4. Write a query to show all the records of employee.

a. Solution

SELECT \* FROM `employee`;

b. Result

EMPLOYEE_ID	F_NAME	LAST_NAME	AGE	ADDRESS	CONTACT_NUM
50	aurelio	laja	21	binalbagan	09126534451
51	wilmae	camanso	24	kabankalan	09467873287
52	carl	marcos	27	himamaylan	09369296755
53	MARIO	tanio	30	binalbagan	09461263436
54	Noel	Dalagan	30	Enclaro	09214567999
55	Roel	Gecosala	27	BRGY.ENCLARO	09214567999
111	May	June	27	Talaban	09214567999

7 result(s) found.



5. Write a query to show all the records of transaction.

a. Solution

SELECT \* FROM `transaction`;

b. Result

TRANSACTION_ID	PRODUCT_ID	QUANTITY	Product_name	TRANS_TYPE	UNIT_PRICE	TOTAL	DATE
10101	1001	12		REFILL	25	300	2019-02-28 18:39:15
10103	1003	2		WATER WITH GALLON	150	300	2019-02-28 18:51:43
10104	1004	1		REFILL	25	25	2019-02-28 18:53:07
10105	1005	1		REFILL	25	25	2019-02-28 18:57:45
10106	1006	2		REFILL	25	25	2019-02-28 18:59:05
10107	1007	1		REFILL	25	25	2019-02-28 18:59:38
10108	1008	3		WATER WITH GALLON	150	450	2019-02-28 19:00:19
10109	1009	2		WATER WITH GALLON	150	300	2019-02-28 19:00:38
10110	1010	1		REFILL	25	25	2019-02-28 19:01:58

12 result(s) found.

6. Write a query to show all the records of payment.

a. Solution

SELECT \* FROM `payment`;

b. Result

PAYMENT_ID	CUSTOMER_ID	TRANSACTION_ID	TOTAL_COST	AMOUNT_RECEIVED	CHANGED	DATE
100A	102	10102	150	200	50	2019-02-28 19:12:36
101A	101	10101	300	500	200	2019-02-28 19:06:57
102A	103	10103	300	1000	800	2019-02-28 19:13:37
103A	104	10104	25	25	0	2019-02-28 19:13:42
104A	105	10105	25	50	25	2019-02-28 19:13:47
105A	106	10106	25	100	75	2019-02-28 19:13:53
106A	107	10107	25	25	0	2019-02-28 19:13:59
107A	108	10108	450	500	50	2019-02-28 19:14:04
108A	109	10109	300	300	0	2019-02-28 19:14:09
109A	110	10110	25	30	5	2019-02-28 19:14:14
110A	111	10111	150	200	50	2019-02-28 19:14:34

11 result(s) found.

7. Write a query to show all the records of delivery.

a. Solution

SELECT \* FROM `delivery`;

b. Result

DELIVERY_ID	CUSTOMER_ID	PRODUCT_ID	TRANSACTION_ID	QUANTITY	DELIVERY_MAN	DATE
200	101	1001	10101	12	MARIO TANIO	2019-02-28 19:24:54
201	103	1003	10108	3	MARIO TANIO	2019-02-28 19:29:28
202	102	1002	10109	2	MARIO TANIO	2019-02-28 19:29:36
203	103	1004	10110	1	MARIO TANIO	2019-02-28 19:29:43
204	105	1005	10102	1	BRUNO MARS	2019-02-28 19:29:51
205	106	1006	10106	2	CARL MARCOS	2019-02-28 19:29:59
206	107	1007	10107	1	CARL MARCOS	2019-02-28 19:30:08
207	108	1008	10103	2	BRUNO MARS	2019-02-28 19:30:17
208	109	1009	10111	1	MARIO TANIO	2019-02-28 19:30:36
209	110	1010	10104	1	BRUNO MARS	2019-02-28 19:31:06
210	111	1011	10105	1	BRUNO MARS	2019-02-28 19:31:23

11

result(s) found.