Informatics Institute of Technology (IIT)

Database Systems(5COSC020C)

Coursework part A+B

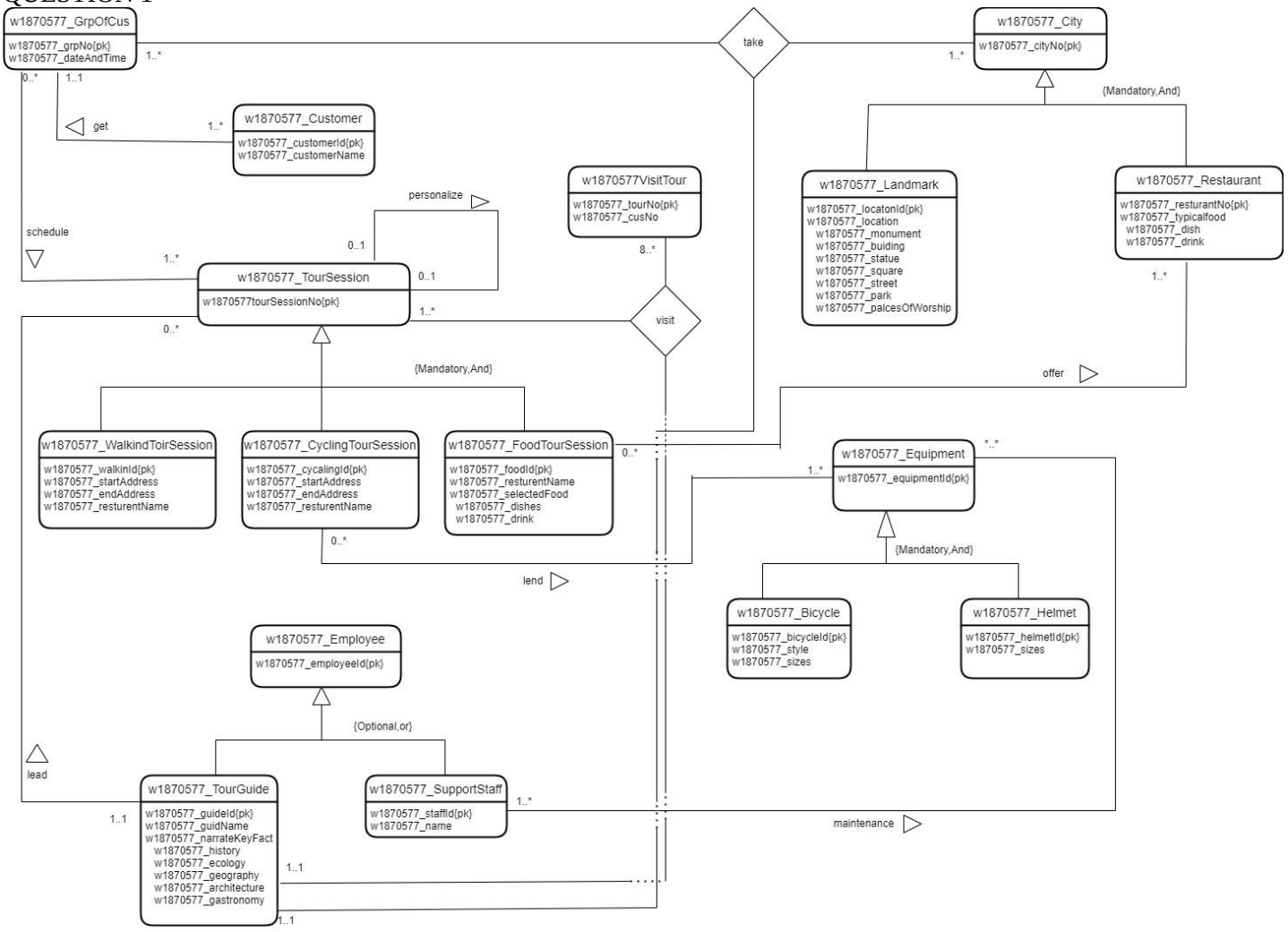
Name: Didula Thaveesha Peduruhewa

Uow No: - w1870577

IIT No: - SE-20210174

Coursework Part A: Conceptual EERD

QUESTION 1



Assumptions

- I assumed that customers would be taken as a group instead of taking each person separately

QUESTION 2

Entity name	Brief Description
W1870577_equipment	Equipment provided by tourmato to carry out the tour
W1870577_City	The term is used to refer to all the cities that the tour guide recommends
W1870577_Customer	The term used to refer to all clients who use tourmato's services generally
W1870577_employee	The term used to refer to all tour guides and staff in general
W1870577_Tour Session	The term is used to refer to all booked trips that the tour operator organizes.
w1870577VisitTour	The term is used to refer to providing all customizable tour sessions.
w1870577_GrpOfCus	Small groups assigned to customers.

General name	Specialized entity	Brief explanation
W1870577_City	W1870577_Landmarks	A general word used to describe all the easily accessible relevant locations in a city.
	W1870577_restaurants	A general word used to describe all the easily accessible relevant locations in a city.
W1870577_TourSession	W1870577_WarkingSession	Customers assigned the city by moving from one destination to another while on foot, according to a general word for the activity.
	W1870577_Cycling Session	A general word for the cycling experience is when visitors ride bicycles between attractions.
	W1870577_FoodSession	A general word used to describe dining experiences where patrons can visit several restaurants and try particular cuisines.
W1870577_Employee	W1870577_tourGuide	The general description of tour guides is that they naturally conduct the tour sessions.
	W1870577_SupStaff	A term used to describe the support staff's stringent upkeep of all Tourmato-related equipment
W1870577_Equipment	W1870577_Bicycle	The bicycle is given to people to go to tour cycling sessions.
	W1870577_Helmet	The Helmet that is given to people to go to tour cycle session

Question 3

Entity name	Multiplicity	Relationship	Multiplicity	Entity name	Brief justifications for the multiplicity
W1870577_TourGuid	1..1	lead	0..*	W1870577_TourSession	1. One tour guide may leaded one tour session. 2. One tour guide may leaded many tour sessions. 3. One tour session must leaded one tour guide. 4. One tour session must leaded one tour guide
W1870577_CyclingTourSession	0..*	lend	1..*	W1870577_Equipment	1. One Cycling TourSession has landed at least one equipment. 2. One Cycling TourSession may have equipment. 3. One equipment may not be lent to any Cycling TourSession. 4. One equipment may have landed many tour sessions.
W1870577_SupportStaff	1..*	maintenance	*..*	W1870577_Equipmentnt	1. one support staff maintains many equipment. 2. one support staff may have maintenance many equipment. 3. One equipment at least maintained by one support staff. 4. One equipment may have maintenance by many support staff.
W1870577_FoodTourSession	0..*	offer	1..*	W1870577_Restaurant	1. One restaurant may haven't offered a food tour session. 2. One restaurant may have offered many food tour sessions. 3. One food tour session offer at least one restaurant. 4. One food tour session may be offered to many restaurants.

w1870577_Customer	1..*	get	1..1	w1870577_GrpOfCus	1. Customersmer get at least one group. 2. Customersmer get at most one group. 3. A group gets at least one customer. 4. A group may have to get many customers.
w1870577_GrpOfCus	0..*	schedule	1..*	w1870577_TourSession	1. One group can schedule one tour session. 2. One group may schedule many tour sessions. 3. One tour session may not schedule by a group. 4. One tour session may schedule by many groups.
w1870577_TourSession	0..1	personalize	0..*	w1870577_TourSession	1. A tour session may not be personalized any tour session. 2. A tour session may have personalized many tour sessions.
w1870577_TourGuide	1..1	take	1..*	w1870577_City	1. One tour guide takes a group to at least one city. 2. One tour guide takes a group to maybe many cities. 3. One customer can be taken by a tour guide to a city. 4. One customer can be taken by a tour guide to many cities. 5. One city can be taken by a tour guide with a group of customers. 6. One city can be taken by a tour guide with many customers.
			1..*	w1870577_GrpOfCus	
w1870577_TourSession	1..*	visit	1..1	w1870577_TourGuide	1. Tour session visit to visit tour at least one tour guide. 2. A tour session visit to one visit tour at most one tour guide. 3. A tour guide who visited one tour session should have at least eight visit tour. 4. A tour guide who visited one tour
			8..*	w1870577VisitTour	

					session may have many visit tour. 5. A tour guide who visited one visit tour may have at least one tour session. 6. A tour guide who visited one visit tour may have at most many tour sessions

QUESTION 4

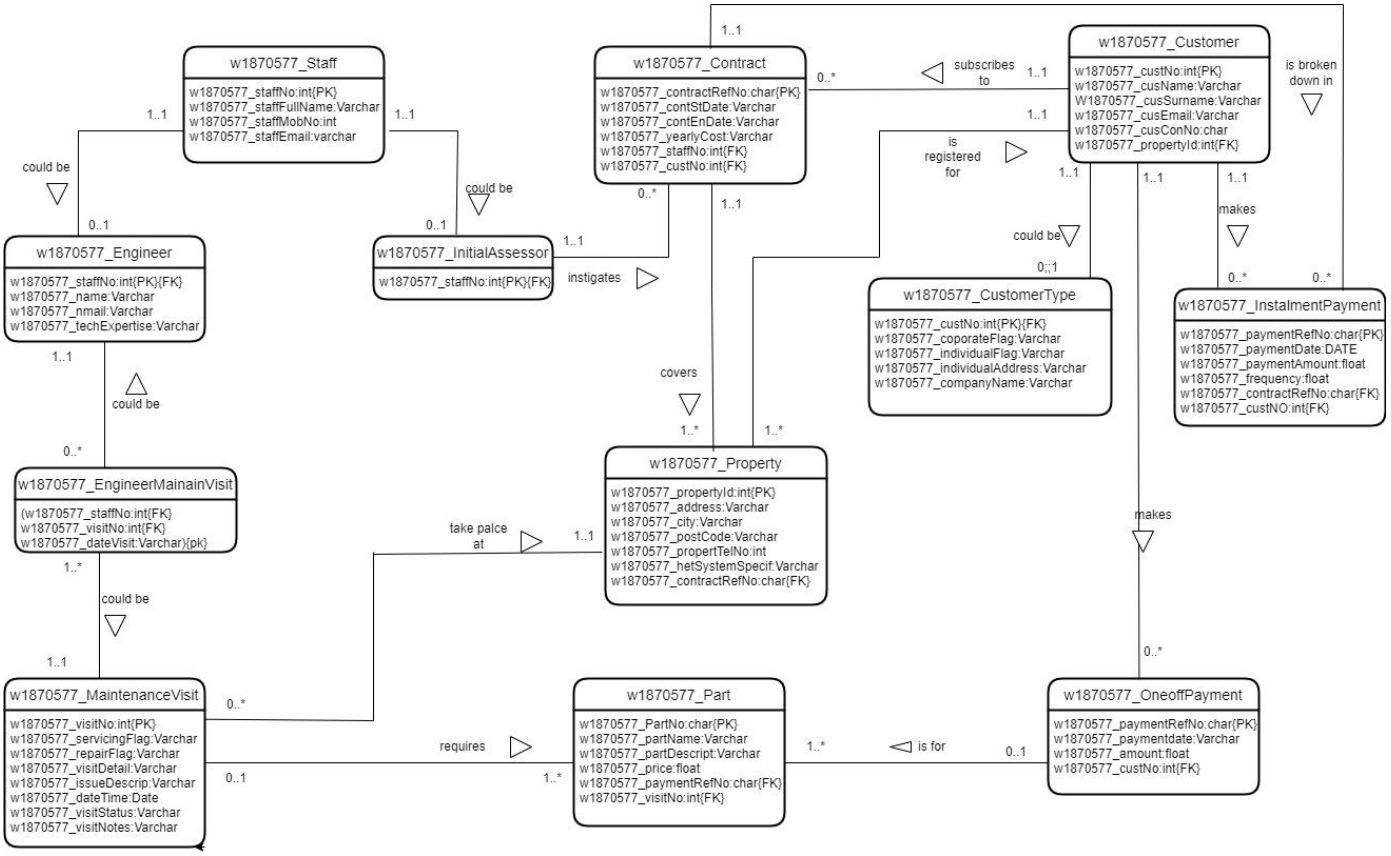
Entity name	Attributes for this entity (include PK)	Brief explanation
W1870577_Equipment	W1870577_equipmentId{pk}	Uniquely identify what the equipment has in tourmato
W1870577_Bicycle	W1870577_bicycleId W1870577_style W1870577_size	Uniquely identify what bicycle Style of bicycle Size of bicycle
W1870577_CycleHelmt	W1870577_helmetId W1870577_size	Uniquely identify what helmet Size of helmet
W1870577_Employee	W1870577_employeeId{pk}	Uniquely identify staff members who work intourmato
W1870577_TourGuid	W1870577_guidId{pk} W1870577_guidName W1870577_narrateKeyFact W1870577_ history W1870577_ecology W1870577_geography W1870577_architecture W1870577_gastronomy	distinctly identify the tour guide named tour guide. Key historical details explained in guidance Key geographic information are explained. Guidance on important architectural facts Key ecological facts are explained. Guidance on crucial gastronomic facts
W1870577_SupportStaff	W1870577_staffId{pk} W1870577_name	To uniquely identify the support staff, their ID and their name

W1870577_City	W1870577_cityNo{pk}	Uniquely identify what are the cities covered by tourmato
W1870577_Customer	W1870577_customerId {pk} W1870577_customerName	Uniquely identify customerswho are joined with tourmato,and his name.
W1870577_TourSession	w1870577tourSessionNo{pk}	Uniquely identify Tour session
W1870577_WalkingTourSession	W1870577_walkingId{pk} W1870577_startAddress W1870577_endAddress W1870577_restarenatName	The starting address of tour session Customer identity for specific identification
W1870577_CyclingTourSession	W1870577_cyclingId{pk} W1870577_startAddress W1870577_endAddress W1870577_restaurantName	The starting address of the tour session Customer identity for specific identification
W1870577_FoodTourSession	W1870577_foodId{pk} W1870577_restaurantName W1870577_selectedFood W1870577_dishes W1870577_drink	It needs a unique code to correctly identify the relevant food. The name of the restaurant and the food selection available to people
W1870577_Landmarks	W1870577_locationId {pk} W1870577_locations W1870577_monument W1870577_building W1870577_statue W1870577_square W1870577_street W1870577_park W1870577_placesOfWorship	Uniquely identify locations What are the monument that tourists can visit? What are the buildings that can visit in customers. What are the statue that can visit in customers. What are the square that can visit in customers. What are the street that can visit in customers. What are the parks that can visit in customers. What are the palace of worship that can visit in customers.

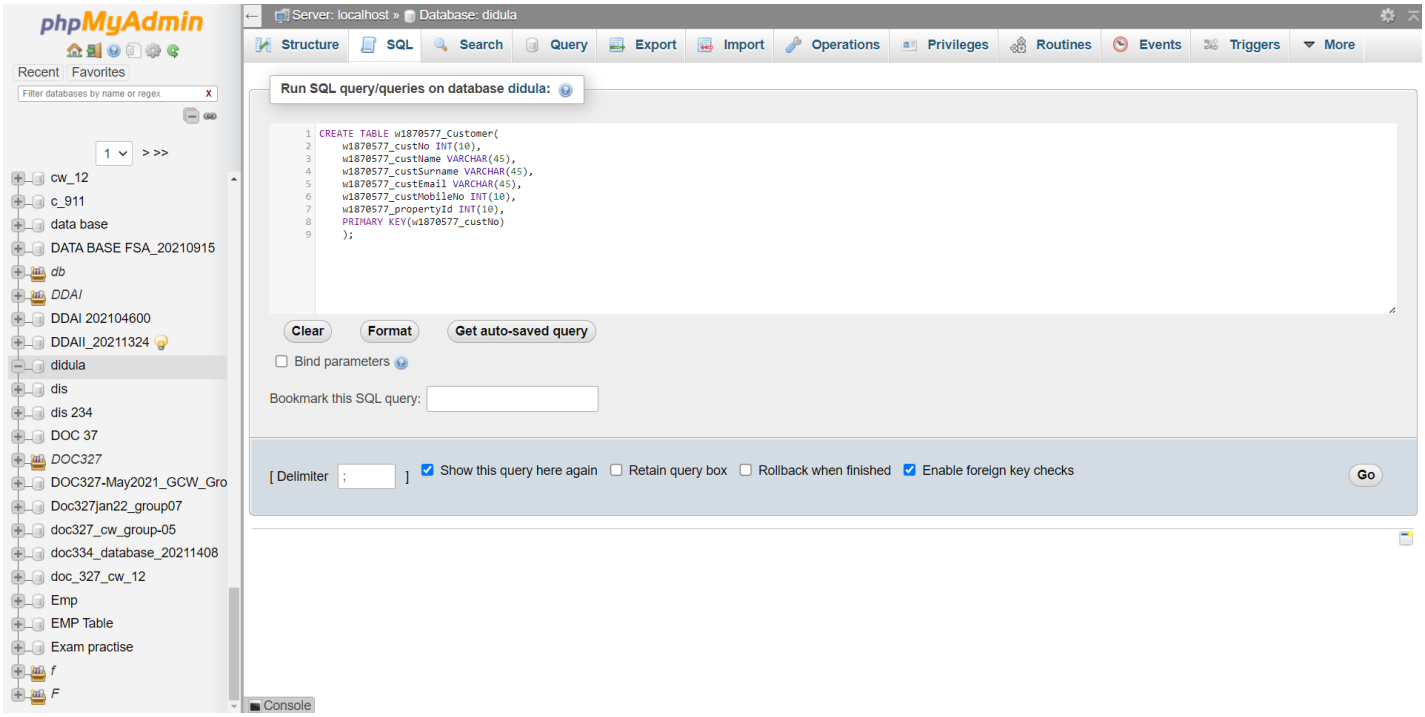
W1870577_Restaurant	W1870577_restaurantNo{pk} W1870577_typicalFood W1870577_dish W1870577_ drink	Uniquely identify restaurants that are booked by customer. Dishes that have available Drinks that have available
w1870577VisitTour	w1870577_tourNo{pk} w1870577_cusNo	Uniquely identify the tour session number. Identify the customized tour session or not.
w1870577_GrpOfCus	w1870577_grpNo{pk} w1870577_dateAndTime	Uniquely identify the group of customers group number. Unique identify the

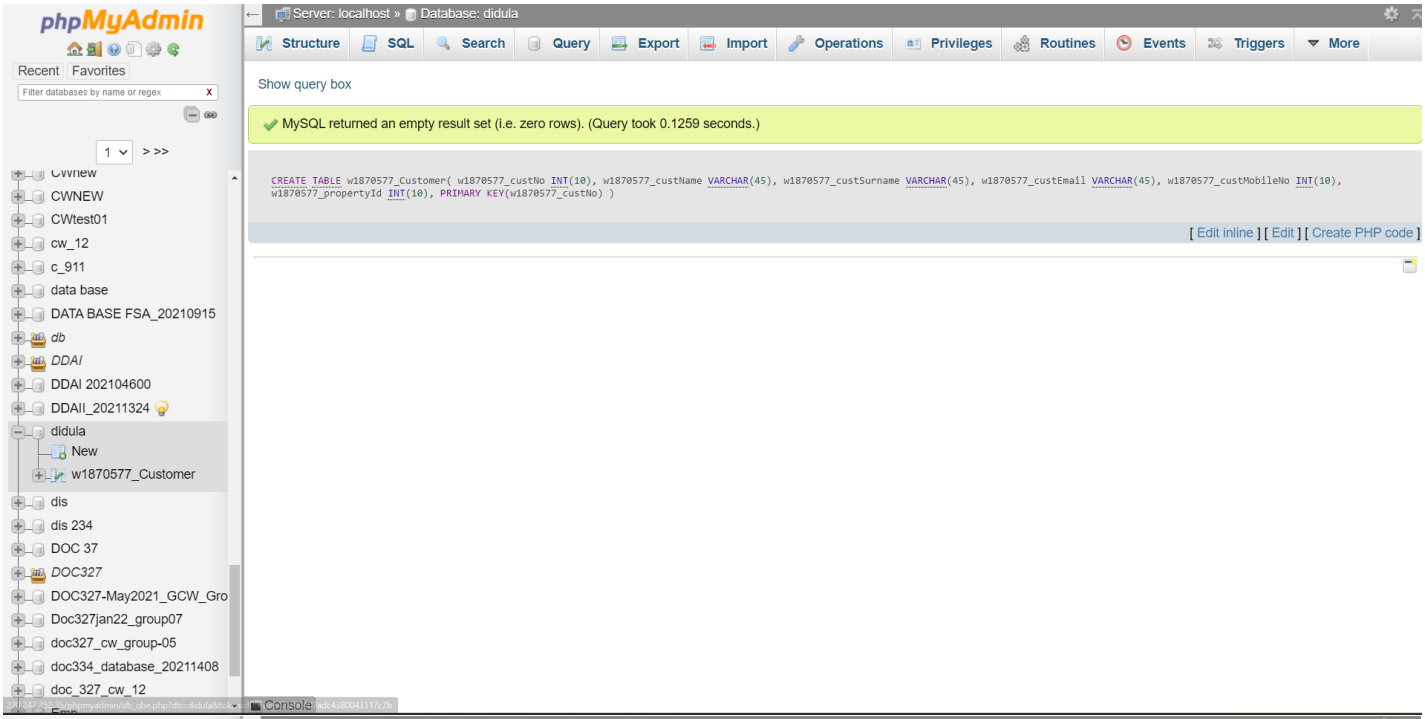
Coursework Part B: Logical ERD, SQL, and Discussion

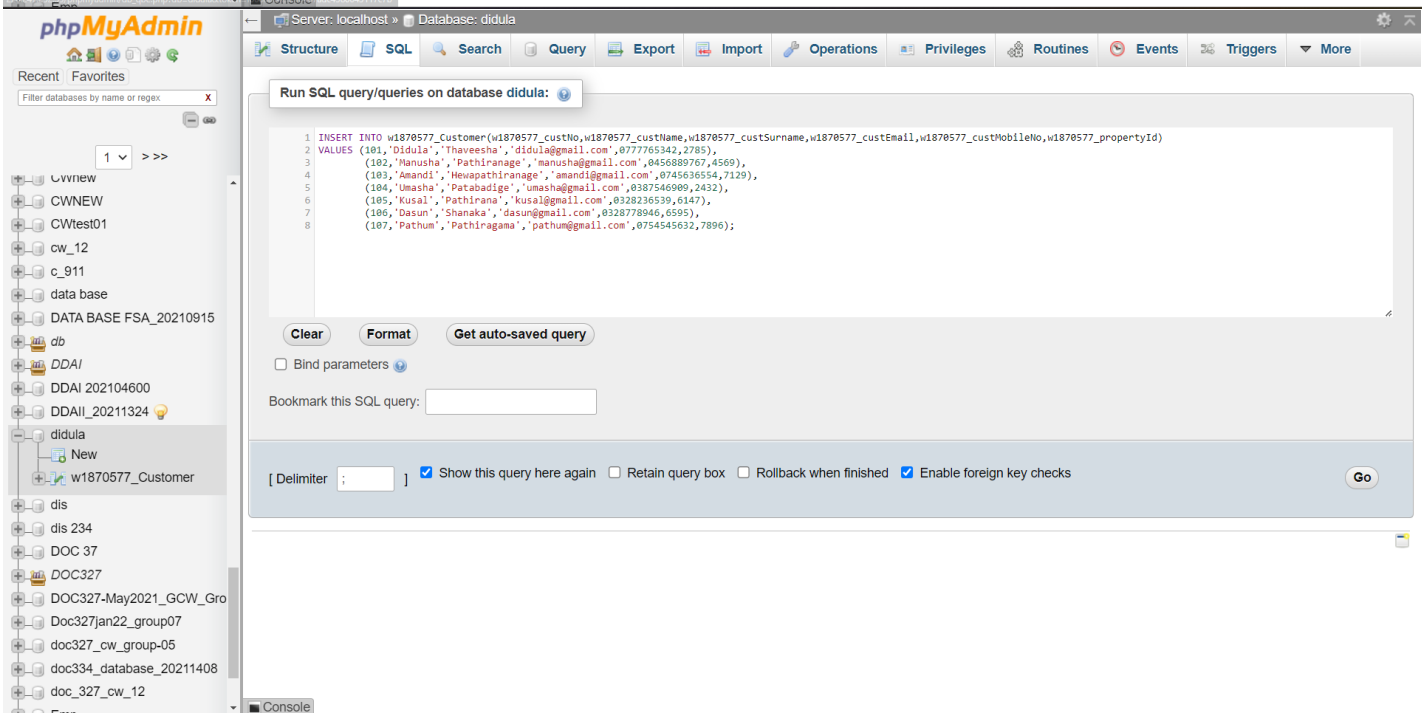
Question 5

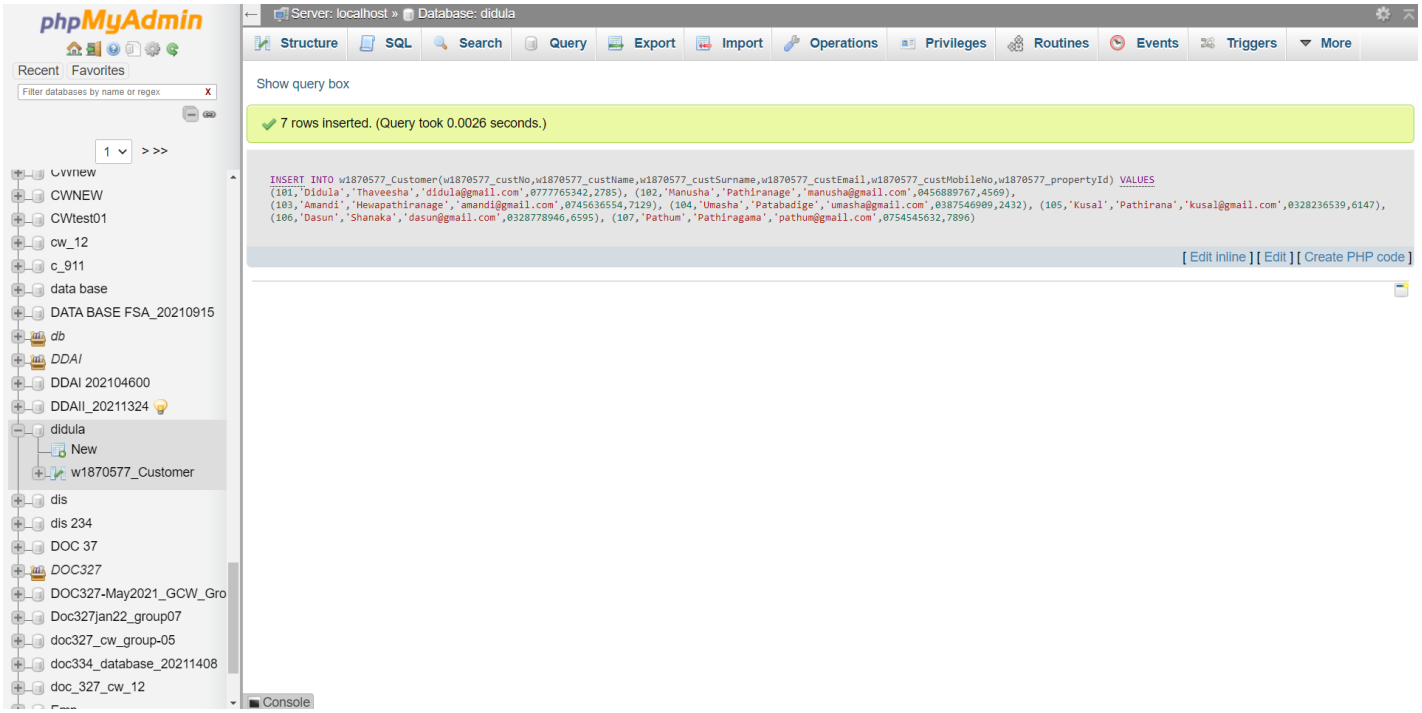


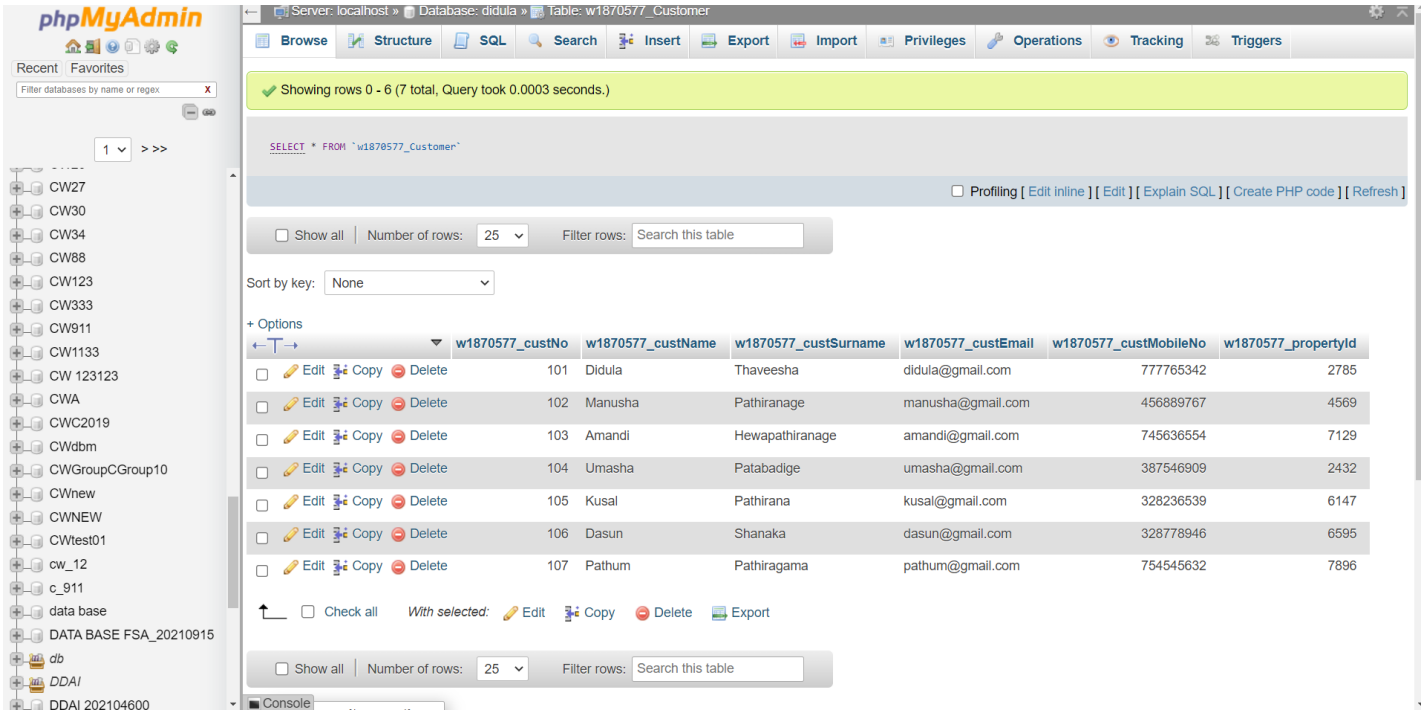
Question 6

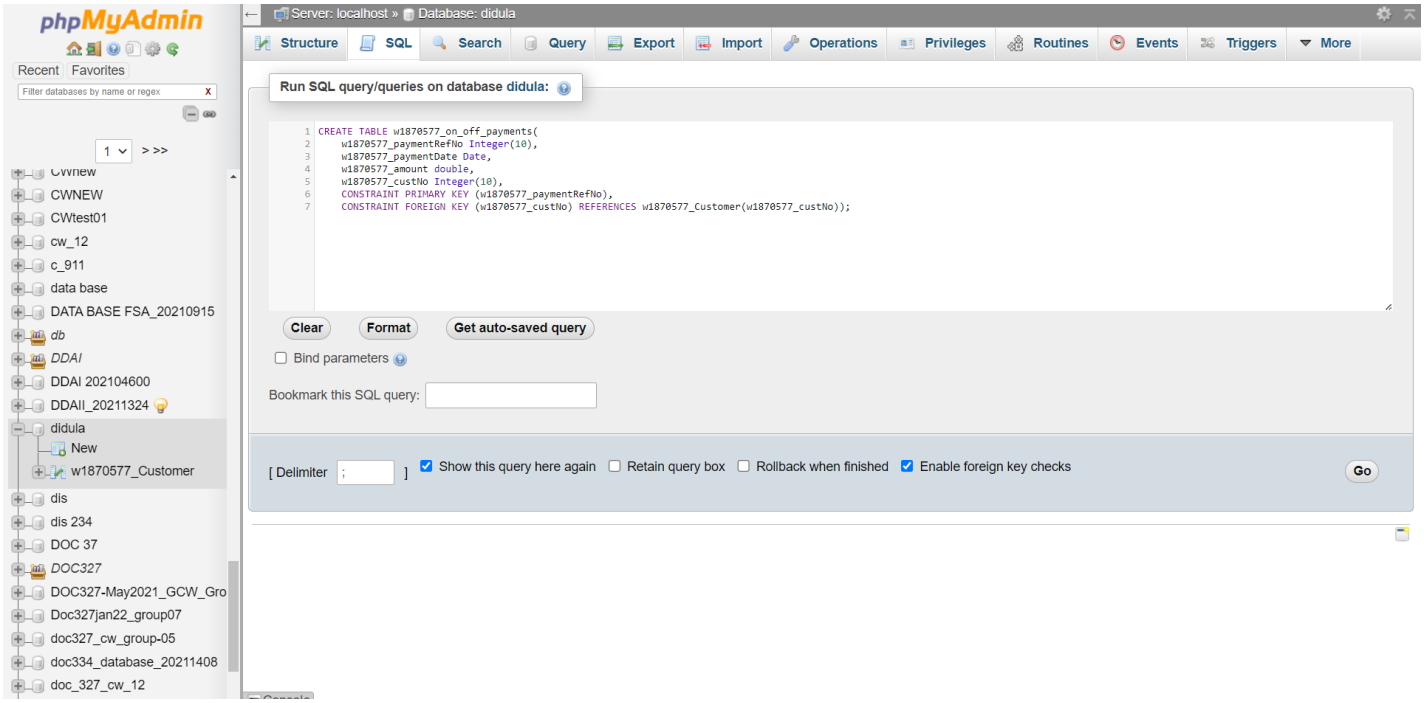

phpMyAdmin
Recent
Favorites
Filter databases by name or regex
cw_12
c_911
data base
DATA BASE FSA_20210915
db
DDAI
DDAI 202104600
DDAII_20211324
didula
dis
dis 234
DOC 37
DOC327
DOC327-May2021_GCW_Gro
Doc327jan22_group07
doc327_cw_group-05
doc334_database_20211408
doc_327_cw_12
Emp
EMP Table
Exam practise
f
F
Server: localhost » Database: didula
Structure
SQL
Search
Query
Export
Import
Operations
Privileges
Routines
Events
Triggers
More
Run SQL query/queries on database didula:
CREATE TABLE w1870577_Customer(
w1870577_custNo INT(10),
w1870577_custName VARCHAR(45),
w1870577_custSurname VARCHAR(45),
w1870577_custEmail VARCHAR(45),
w1870577_custMobileNo INT(10),
w1870577_propertyId INT(10),
PRIMARY KEY(w1870577_custNo)
);
Clear
Format
Get auto-saved query
Bind parameters
Bookmark this SQL query:
[Delimiter ;]
Show this query here again
Retain query box
Rollback when finished
Enable foreign key checks
Go
Console

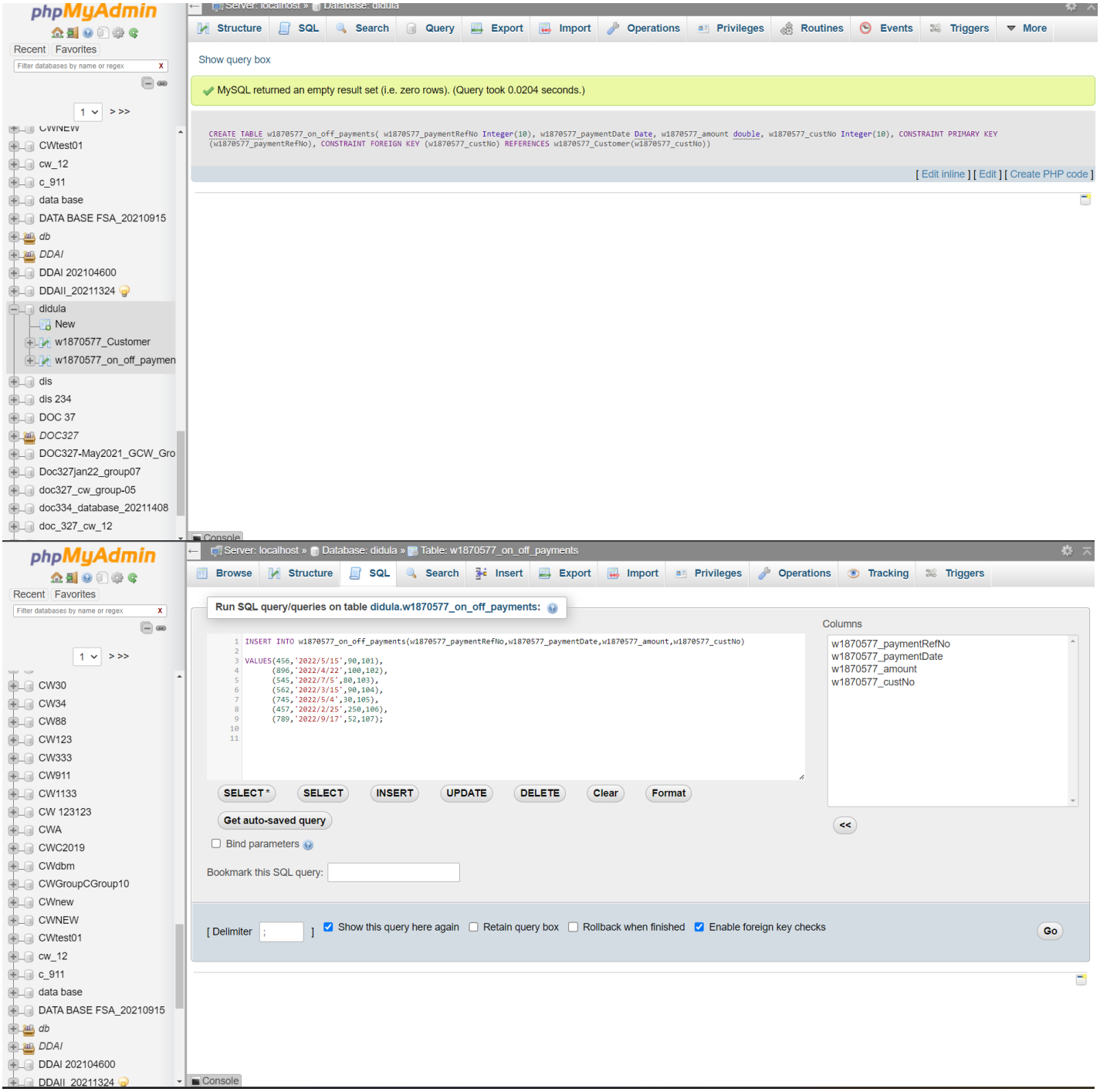

phpMyAdmin
Recent Favorites
Filter databases by name or regex
CWNEW
CWtest01
cw_12
c_911
data base
DATA BASE FSA_20210915
db
DDAI
DDAI 202104600
DDAII_20211324
didula
New
w1870577_Customer
dis
dis 234
DOC 37
DOC327
DOC327-May2021_GCW_Gro
Doc327jan22_group07
doc327_cw_group-05
doc334_database_20211408
doc_327_cw_12
Server: localhost » Database: didula
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers More
Show query box
MySQL returned an empty result set (i.e. zero rows). (Query took 0.1259 seconds.)
CREATE TABLE w1870577_Customer(w1870577_custNo INT(10), w1870577_custName VARCHAR(45), w1870577_custSurname VARCHAR(45), w1870577_custEmail VARCHAR(45), w1870577_custMobileNo INT(10), w1870577_propertyId INT(10), PRIMARY KEY(w1870577_custNo))
[Edit inline] [Edit] [Create PHP code]
Console

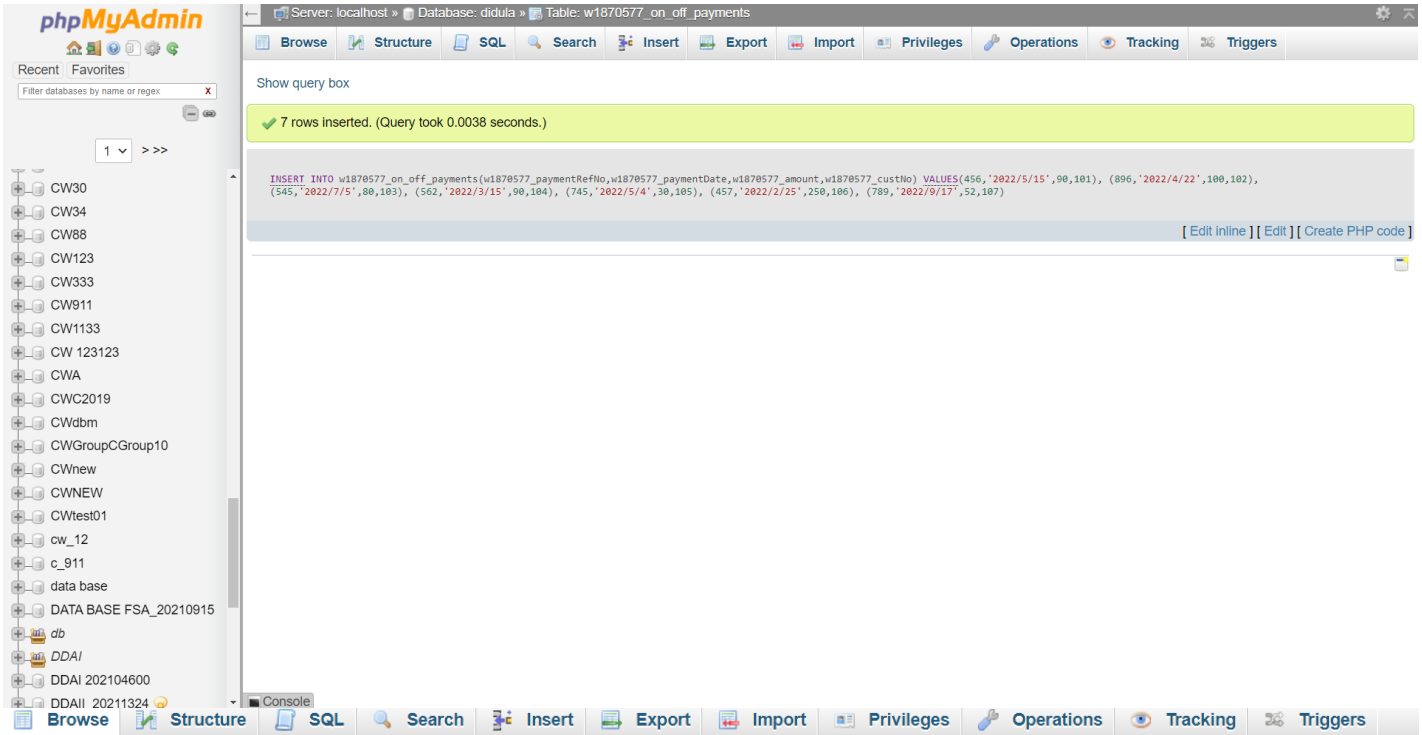

phpMyAdmin
Recent Favorites
Filter databases by name or regex
CWNEW
CWtest01
cw_12
c_911
data base
DATA BASE FSA_20210915
db
DDAI
DDAI 202104600
DDAII_20211324
didula
New
w1870577_Customer
dis
dis 234
DOC 37
DOC327
DOC327-May2021_GCW_Gro
Doc327jan22_group07
doc327_cw_group-05
doc334_database_20211408
doc_327_cw_12
Server: localhost » Database: didula
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers More
Run SQL query/queries on database didula:
INSERT INTO w1870577_Customer(w1870577_custNo,w1870577_custName,w1870577_custSurname,w1870577_custEmail,w1870577_custMobileNo,w1870577_propertyId)
VALUES (101,'Didula','Thaveesha','didula@gmail.com',0777765342,2785),
(102,'Manusha','Pathiranage','manusha@gmail.com',0456889767,4569),
(103,'Amandi','Hewapathiranage','amandi@gmail.com',0745636554,7129),
(104,'Umasha','Patabadige','umasha@gmail.com',0387546909,2432),
(105,'Kusal','Pathirana','kusal@gmail.com',0328236539,6147),
(106,'Dasun','Shanaka','dasun@gmail.com',0328778946,6595),
(107,'Pathum','Pathiragama','pathum@gmail.com',0754545632,7896);
Clear Format Get auto-saved query
Bind parameters
Bookmark this SQL query:
[Delimiter ;] Show this query here again Retain query box Rollback when finished Enable foreign key checks
Go
Console


phpMyAdmin
Recent Favorites
Filter databases by name or regex
CWNEW
CWtest01
cw_12
c_911
data base
DATA BASE FSA_20210915
db
DDAI
DDAI 202104600
DDAII_20211324
didula
New
w1870577_Customer
dis
dis 234
DOC 37
DOC327
DOC327-May2021_GCW_Gro
Doc327jan22_group07
doc327_cw_group-05
doc334_database_20211408
doc_327_cw_12
Server: localhost » Database: didula
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers More
Show query box
7 rows inserted. (Query took 0.0026 seconds.)
INSERT INTO w1870577_Customer(w1870577_custNo,w1870577_custName,w1870577_custSurname,w1870577_custEmail,w1870577_custMobileNo,w1870577_propertyId) VALUES
(101,'Didula','Thaveesha','didula@gmail.com',0777765342,2785), (102,'Manusha','Pathiranage','manusha@gmail.com',0456889767,4569),
(103,'Amandi','Hewapathiranage','amandi@gmail.com',0745636554,7129), (104,'Umasha','Patabadige','umasha@gmail.com',0387546909,2432), (105,'Kusal','Pathirana','kusal@gmail.com',0328236539,6147),
(106,'Dasun','Shanaka','dasun@gmail.com',0328778946,6595), (107,'Pathum','Pathiragama','pathum@gmail.com',0754545632,7896)
[Edit inline] [Edit] [Create PHP code]
Console


phpMyAdmin
Recent Favorites
Filter databases by name or regex
CW27
CW30
CW34
CW88
CW123
CW333
CW911
CW1133
CW 123123
CWA
CWC2019
CWdbm
CWGroupCGroup10
CWnew
CWNEW
CWtest01
cw_12
c_911
data base
DATA BASE FSA_20210915
db
DDAI
DDAI 202104600
Server: localhost » Database: didula » Table: w1870577_Customer
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Showing rows 0 - 6 (7 total, Query took 0.0003 seconds.)
SELECT * FROM `w1870577_Customer`
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all Number of rows: 25 Filter rows: Search this table
Sort by key: None
+ Options
w1870577_custNo w1870577_custName w1870577_custSurname w1870577_custEmail w1870577_custMobileNo w1870577_propertyId
Edit Copy Delete 101 Didula Thaveesha didula@gmail.com 777765342 2785
Edit Copy Delete 102 Manusha Pathiranage manusha@gmail.com 456889767 4569
Edit Copy Delete 103 Amandi Hewapathiranage amandi@gmail.com 745636554 7129
Edit Copy Delete 104 Umasha Patabadige umasha@gmail.com 387546909 2432
Edit Copy Delete 105 Kusal Pathirana kusal@gmail.com 328236539 6147
Edit Copy Delete 106 Dasun Shanaka dasun@gmail.com 328778946 6595
Edit Copy Delete 107 Pathum Pathiragama pathum@gmail.com 754545632 7896
Check all With selected: Edit Copy Delete Export
Show all Number of rows: 25 Filter rows: Search this table
Console


phpMyAdmin
Recent
Favorites
Filter databases by name or regex
CWNEW
CWtest01
cw_12
c_911
data base
DATA BASE FSA_20210915
db
DDAI
DDAI 202104600
DDAII_20211324
didula
New
w1870577_Customer
dis
dis 234
DOC 37
DOC327
DOC327-May2021_GCW_Gro
Doc327jan22_group07
doc327_cw_group-05
doc334_database_20211408
doc_327_cw_12
Server: localhost » Database: didula
Structure
SQL
Search
Query
Export
Import
Operations
Privileges
Routines
Events
Triggers
More
Run SQL query/queries on database didula:
CREATE TABLE w1870577_on_off_payments(
w1870577_paymentRefNo Integer(10),
w1870577_paymentDate Date,
w1870577_amount double,
w1870577_custNo Integer(10),
CONSTRAINT PRIMARY KEY (w1870577_paymentRefNo),
CONSTRAINT FOREIGN KEY (w1870577_custNo) REFERENCES w1870577_Customer(w1870577_custNo));
Clear
Format
Get auto-saved query
Bind parameters
Bookmark this SQL query:
[Delimiter ;]
Show this query here again
Retain query box
Rollback when finished
Enable foreign key checks
Go


phpMyAdmin
Recent Favorites
Filter databases by name or regex
CWtest01
cw_12
c_911
data base
DATA BASE FSA_20210915
db
DDAI
DDAI 202104600
DDAII_20211324
didula
New
w1870577_Customer
w1870577_on_off_paymen
dis
dis 234
DOC 37
DOC327
DOC327-May2021_GCW_Gro
Doc327jan22_group07
doc327_cw_group-05
doc334_database_20211408
doc_327_cw_12
Server: localhost » Database: didula
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers More
Show query box
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0204 seconds.)
CREATE TABLE w1870577_on_off_payments(w1870577_paymentRefNo Integer(10), w1870577_paymentDate Date, w1870577_amount double, w1870577_custNo Integer(10), CONSTRAINT PRIMARY KEY (w1870577_paymentRefNo), CONSTRAINT FOREIGN KEY (w1870577_custNo) REFERENCES w1870577_Customer(w1870577_custNo))
[Edit inline] [Edit] [Create PHP code]
Console
phpMyAdmin
Recent Favorites
Filter databases by name or regex
CW30
CW34
CW88
CW123
CW333
CW911
CW1133
CW 123123
CWA
CWC2019
CWdbm
CWGroupCGroup10
CWnew
CWNEW
CWtest01
cw_12
c_911
data base
DATA BASE FSA_20210915
db
DDAI
DDAI 202104600
DDAII_20211324
Server: localhost » Database: didula » Table: w1870577_on_off_payments
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Run SQL query/queries on table didula.w1870577_on_off_payments:
INSERT INTO w1870577_on_off_payments(w1870577_paymentRefNo,w1870577_paymentDate,w1870577_amount,w1870577_custNo)
VALUES(456,'2022/5/15',90,101),
(896,'2022/4/22',100,102),
(545,'2022/7/5',80,103),
(562,'2022/3/15',90,104),
(745,'2022/5/4',30,105),
(457,'2022/2/25',250,106),
(789,'2022/9/17',52,107);
Columns
w1870577_paymentRefNo
w1870577_paymentDate
w1870577_amount
w1870577_custNo
SELECT * SELECT INSERT UPDATE DELETE Clear Format
Get auto-saved query
<<
Bind parameters
Bookmark this SQL query:
[Delimiter ;] Show this query here again Retain query box Rollback when finished Enable foreign key checks
Go
Console



Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Showing rows 0 - 6 (7 total, Query took 0.0004 seconds.)

```
SELECT * FROM `w1870577_on_off_payments`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP co

Show all | Number of rows: 25 Filter rows: Search this table

Sort by key: None

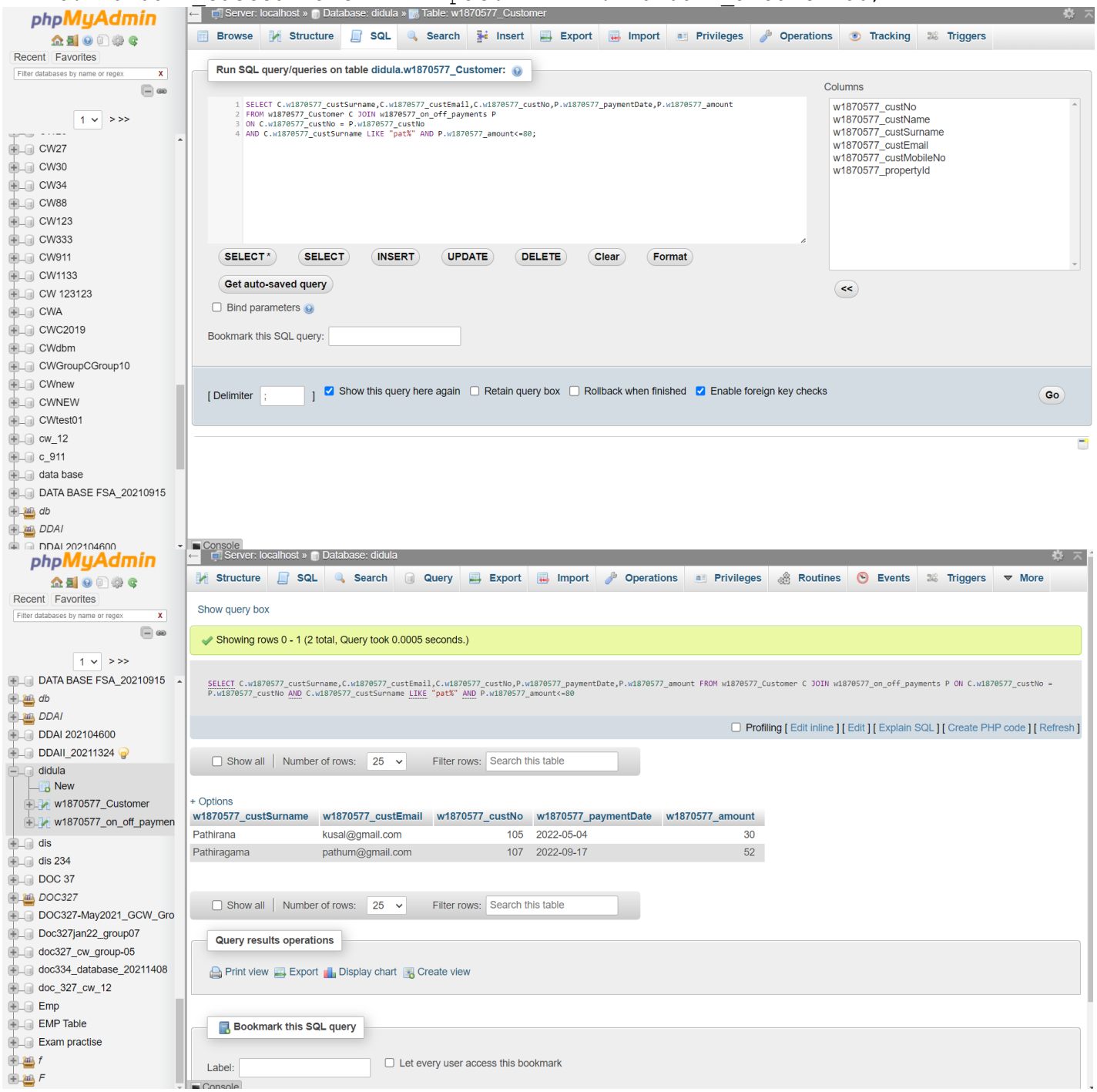
+ Options

	w1870577_paymentRefNo	w1870577_paymentDate	w1870577_amount	w1870577_custNo
Edit Copy Delete	456	2022-05-15	90	101
Edit Copy Delete	457	2022-02-25	250	106
Edit Copy Delete	545	2022-07-05	80	103
Edit Copy Delete	562	2022-03-15	90	104
Edit Copy Delete	745	2022-05-04	30	105
Edit Copy Delete	789	2022-09-17	52	107
Edit Copy Delete	896	2022-04-22	100	102

Check all *With selected:* Edit Copy Delete Export

Show all | Number of rows: 25 Filter rows: Search this table

```
SELECT
C.w1870577_custSurname,C.w1870577_custEmail,C.w1870577_custNo,P.w1870577_payment
Date,P.w1870577_amount
FROM w1870577_Customer C JOIN w1870577_on_off_payments P
ON C.w1870577_custNo = P.w1870577_custNo
AND C.w1870577_custSurname LIKE "pat%" AND P.w1870577_amount<=80;
```



Question 7

Features	Mysql	Mongodb
Performance	MySQL is designed for powerful joins between diverse, properly indexed tables. Additionally, while using MySQL, data should be added row by row. Even though MySQL excels at choosing the next type of data, as the number of records grows, its speed may suffer.	uses a hierarchy-based data design that keeps most of the data in a single document, reducing the requirement for joins across many documents. With a dedicated insert many() API for swiftly inserting data that stresses speed above transaction safety, MongoDB is likewise optimized for write performance. Additionally, inputting or editing several records, is significantly faster. As a consequence, MongoDB is a perfect substitute for MySQL when it comes to adding and updating data more quickly.(Wei-ping, Ming-xin and Huan, 2011)
Schema	A schema must be created for each table in a MySQL database since it is a common electronic information service. Mysql is a trustworthy alternative for structured data as a result.	No prior schema is necessary for MongoDB. Real-time analytics, content management, the Internet of Things, mobile apps, and other uses are perfect for it. It is a great option for data that might grow quickly, whether it is unstructured or organized. In a management organization, MongoDB is the ideal choice for decision-making since it has a more adaptable structure than MySQL.
Data storage	Each entry in MySQL is kept as "rows" in a table. It has a correct, rigid structure that is more difficult to change than MongoDB.	MongoDB stores each record as a separate "document." In other words, it is simple to add new data, remove some data, and change data.(Wei-ping, Ming-xin and Huan, 2011)
Scalability	With the MySQL database architecture, options are somewhat more constrained. There are normally two choices: adding read replicates or vertical scaling. Reading replication is the process of	One key benefit of the MongoDB design is how simple it is to grow the database. A sharded cluster enables the configuration of a database's component as a replica set. A shared cluster's data is spread out among

	adding read-only copies of the database to other servers. However, there are only five copies available. Applications that often write to and read from the database or are write-intensive might have issues with this. Although MySQL now has multi-master replication capabilities, its approach is less flexible than MongoDB's.	several computers. Due to MongoDB's highly flexible design, read and write performance can be expanded horizontally to support applications of any scale.(Damodaran B, Salim and Vargese, 2016)
Architecture	The client-server architecture of MySQL, on the other hand, has multithreading and rapid storage. Additionally, it lists a few setup-focused speed enhancement techniques rather than ones that focus on fine-tuning SQL measures in its handbook.(Győrödi et al., 2015)	Nexus Architecture incorporates relational database functionalities thanks to MongoDB's architectural idea. High scalability, global availability, and a customizable schema enable it to meet the requirements of modern applications. Therefore, altering Its design is easy.

Reference:-

Damodaran B, D., Salim, S. and Vargese, S.M. (2016). Performance Evaluation of MySQL and MongoDB Databases. *International Journal on Cybernetics & Informatics*, 5 (2), 387–394. Available from https://doi.org/10.5121/ijci.2016.5241.

Győrödi, C. et al. (2015). A comparative study: MongoDB vs. MySQL. *2015 13th International Conference on Engineering of Modern Electric Systems (EMES)*. June 2015. 1–6. Available from https://doi.org/10.1109/EMES.2015.7158433.

Patil, M.M. et al. (2017). A qualitative analysis of the performance of MongoDB vs MySQL database based on insertion and retriewal operations using a web/android application to explore load balancing — Sharding in MongoDB and its advantages. *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*. February 2017. 325–330. Available from

https://doi.org/10.1109/I-SMAC.2017.8058365.

Wei-ping, Z., Ming-xin, L. and Huan, C. (2011). Using MongoDB to implement textbook management system instead of MySQL. *2011 IEEE 3rd International Conference on Communication Software and Networks*. May 2011. 303–305. Available from https://doi.org/10.1109/ICCSN.2011.6013720.