



2814ICT – DATA MANAGEMENT
7003ICT – DATABASE DESIGN
School of Information & Communication Technology
Trimester _1_, 2021_

Assignment Part 2: Implementing a Database for BigM

ASSIGNMENT TITLE: Designing a Database for BigM

Group ID	065	
Student 1	s-number: s5210032	Full name: Jingdi Lin
Student 2	s-number: s5204340	Full name: Heang SOK
Course Code: 7003ICT		Workshop/Lab day & time: 8am Thu/ 8am Wed
Tutor's name: John Wang/Fahimeh Alaei		Date submitted: 27/05/2021

Table of Contents

Table of Contents	2
List of Illustrations	2
Statement of Completion	2
Acknowledgements	2
Reports on SQL Queries in Task 3	3
Bibliography	10

List of Illustrations

Not applicable.

Statement of Completion

All tasks have been completed.

Acknowledgements

- 1) Mohammad Awrangjeb
- 2) John Wang
- 3) Fahimeh Alaei

Reports on SQL Queries in Task 3

Query 1:

-- I. List of names of all employees sorted by their hourly salary.

```
SELECT CONCAT (Emp_FName,' ',Emp_LName) AS 'Employee Name',  
CONCAT ('$ ', Emp_HourlySalary) AS 'Hourly Salary' from EMPLOYEE  
ORDER BY Emp_HourlySalary ASC;
```

Output table:

Employee Name	Hourly Salary
Hero Gardner	\$20.00
Rock Lee	\$20.00
Logan Lee	\$20.00
Joma Homa	\$20.00
Debra Morgant	\$20.00
Inoha Shiha	\$20.00
Touya Hima	\$20.00
Ayaka Kim	\$20.00
Big Mark	\$20.00
Barry Allen	\$20.00
Clark Kent	\$20.00
Sheldon Tien	\$20.00
Yui Yuigahama	\$23.00
Himaru Ishin	\$23.00
Hachiman Hikigaya	\$23.00
Jordan Buffet	\$23.00
Tracey Ly	\$25.00
Tele Cross	\$25.00
Paige Gate	\$28.00
Alice Cook	\$28.00
Winter Soldier	\$28.00
John Gardner	\$30.00
Joseph Smith	\$30.00
Cole Mac	\$30.00
Richard Homan	\$30.00
John Gardner	\$30.00
Char Kara	\$30.00
Minghun Lee	\$35.00
Ichigo Morgant	\$35.00
Steve Rogger	\$35.00
Kong Bing	\$45.00
June LY	\$45.00
Petter Parker	\$45.00
Tom Musk	\$45.00
Jessica Lin	\$45.00

Query 2:

-- II. The date on which the most recent customer order has been made.

```
SELECT CONCAT (CUSTOMER.Cust_FName,' ', CUSTOMER.Cust_LName) AS  
'Customer Name', CUSTOMERORDER.CustOrd_Date AS 'Date'  
FROM CUSTOMER, CUSTOMERORDER WHERE CUSTOMER.Cust_Number =  
CUSTOMERORDER.Cust_Number  
ORDER BY CustOrd_Date DESC LIMIT 0,1;
```

Output table:

+ Options

	Customer Name	Date
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	Monica Geller	2018-05-24
















Query 3:

-- III. List all the store names and their manager names, sorted in dictionary-order of the store name.

```
SELECT STORE.Str_Name AS 'Store Name', CONCAT (EMPLOYEE.Emp_FName, ' ',  
EMPLOYEE.Emp_LName) AS 'Manager Name' FROM STORE, EMPLOYEE  
WHERE STORE.StoreManagerID = EMPLOYEE.Emp_ID  
ORDER BY Str_Name ASC;
```

Output table:

+ Options

<div><div>←</div><div>T</div><div>→</div></div> <div></div>				Store Name	Manager Name
<input type="checkbox"/>	 Edit	 Copy	 Delete	Arundel	Petter Parker
<input type="checkbox"/>	 Edit	 Copy	 Delete	Ashmore	June LY
<input type="checkbox"/>	 Edit	 Copy	 Delete	Koala	Kong Bing
<input type="checkbox"/>	 Edit	 Copy	 Delete	Nathan	Tom Musk
<input type="checkbox"/>	 Edit	 Copy	 Delete	Parkwood	Jessica Lin

Query 4:

-- IV. List all customers that have not placed an order yet.

```
SELECT CUSTOMER.Cust_Number AS 'Customer Number', CONCAT  
(CUSTOMER.Cust_FName, ' ', CUSTOMER.Cust_LName) AS 'Customer Name'  
FROM CUSTOMER  
LEFT JOIN CUSTOMERORDER  
ON CUSTOMER.Cust_Number = CUSTOMERORDER.Cust_Number  
WHERE CUSTOMER.Cust_Number NOT IN (SELECT CUSTOMERORDER.Cust_Number  
from CUSTOMERORDER);
```

Output table:

Customer Number	Customer Name
3	Rachel Green
5	Joey Tribbiani
7	Natalie Wang
9	Jingly Jim
10	Yaka Kim

Query 5:

-- V. A list containing the name of employees, who work as supervisors for 'Sports' departments in various stores.

```
SELECT S.Str_Name As 'Store Name', CONCAT (Emp_FName,',',Emp_LName) AS  
'Supervisors Of Sports Departments'  
FROM STORE AS S  
JOIN STOREDEPARTMENT AS SD ON S.Str_Num = SD.Str_Num  
JOIN EMPLOYEE AS E ON E.Emp_ID = SD.DeptSupervisorID  
JOIN DEPARTMENT AS D ON D.Dept_ID = SD.Dept_ID  
WHERE D.Dept_Name = 'Sports';
```

Output table:

+ Options

Store Name	Supervisors Of Sports Departments
Parkwood	Touya Hima
Nathan	Steve Rogger
Arundel	Joma Homa
Ashmore	Minghun Lee
Koala	Hachiman Hikigaya

Query 6:

-- VI. A list containing the total quantity on hand for each product regardless of stores

```
SELECT P.Prod_Num AS 'Product Number', P.Prod_Desc AS 'Description',  
SUM(I.Inv_QntyOnHand) AS 'Quantity On Hand'  
FROM INVENTORY AS I  
JOIN PRODUCT AS P ON P.Prod_Num = I.ProductNum  
GROUP BY I.ProductNum;
```

Output table:

Product Number	Description	Quantity On Hand
1	Women Shirt - S	73
2	Women Shirt - M	60
3	Women Shirt - L	64
4	Women Jeans - S	75
5	Women Jeans - M	74
6	Women Jeans - L	77
7	Men Shirt - S	77
8	Men Shirt - M	62
9	Men Shirt - L	66
10	Men Jeans - S	59
11	Men Jeans - M	74
12	Men Jeans - L	63

Query 7:

-- VII. A list showing each product sold (picked) on or before May 20, 2018. Show product number, name and quantity sold, sorted by product number and then quantity sold

```
SELECT P.Prod_Num AS 'Product Number', P.Prod_Desc AS 'Name',  
SUM(OL.OrdLn_Qnty) AS 'Quantity Sold'  
FROM PRODUCT AS P  
JOIN ORDERLINE AS OL ON P.Prod_Num = OL.Prod_Num  
WHERE OL.OrdLn_DatePicked < '2018-05-20'  
GROUP BY P.Prod_Num  
ORDER BY P.Prod_Num, OL.OrdLn_Qnty;
```

Output table:

+ Options

Product Number	Name	Quantity Sold
1	Women Shirt - S	8
2	Women Shirt - M	6
3	Women Shirt - L	4
5	Women Jeans - M	9
7	Men Shirt - S	3
8	Men Shirt - M	2
10	Men Jeans - S	13
11	Men Jeans - M	8
12	Men Jeans - L	3

Query 8:

-- VIII. A list of products (show product number, description and price) whose price is less than or equal to the average product price

```
SELECT Prod_Num AS 'Product Number', Prod_Desc AS 'Description', CONCAT('$',  
Prod_Price) AS 'Price'  
FROM PRODUCT  
WHERE Prod_Price <= (SELECT AVG(Prod_Price) FROM PRODUCT);
```

Output table:

				Product Number	Description	Price
<input type="checkbox"/>				1	Women Shirt - S	\$15.00
<input type="checkbox"/>				2	Women Shirt - M	\$16.00
<input type="checkbox"/>				7	Men Shirt - S	\$10.00
<input type="checkbox"/>				8	Men Shirt - M	\$11.00
<input type="checkbox"/>				9	Men Shirt - L	\$12.00
<input type="checkbox"/>				10	Men Jeans - S	\$16.00

Query 9:

-- IX. Increase each employee's salary by 7.5% and show the updated salary of all employees (name and salary)

```
SELECT CONCAT(Emp_FName, ' ', Emp_LName) AS 'Employee Name', CONCAT('$',  
CAST(1.075 * Emp_HourlySalary AS DECIMAL(7,2))) AS 'Salary Increased by 7.5%'  
FROM EMPLOYEE;
```

Output table:

Employee Name	Salary Increased by 7.5%
Jessica Lin	\$48.38
Clark Kent	\$21.50
Barry Allen	\$21.50
John Gardner	\$32.25
Joseph Smith	\$32.25
Ayaka Kin	\$21.50
Touya Hima	\$21.50
Tom Musk	\$48.38
Debra Morgant	\$21.50
Cole Mac	\$32.25
Richard Homan	\$32.25
Stracey Ly	\$26.88
Rock Lee	\$21.50
Steve Rogger	\$37.63
Petter Parker	\$48.38
Tom Cruise	\$32.25
Himaru Ishin	\$24.73
Ichigo Morgant	\$37.63
Logan Lee	\$21.50
Alice Cook	\$30.10
Joma Homa	\$21.50
June LY	\$48.38
Tele Cross	\$26.88
Winter Soldier	\$30.10
Big Mark	\$21.50
Employee Name	Salary Increased by 7.5%
Hero Gardner	\$21.50
Char Kara	\$32.25
Minghun Lee	\$37.63
Kong Bing	\$48.38
Yui Yuigahama	\$24.73
Sheldon Tien	\$21.50
Inoha Shiha	\$21.50
Paige Gate	\$30.10
Jordan Buffet	\$24.73
Hachiman Hikigaya	\$24.73

Query 10:

-- X. Show the pay information (employee name, hours paid, amount paid) of all employees in the most recent pay date

```
SELECT CONCAT(E.Emp_FName, ' ', E.Emp_LName) AS 'Employee Name', CONCAT  
('$', E.Emp_HourlySalary) AS 'Hours paid', CONCAT('$', PA.Pay_amount_gross) AS  
'Amount Paid'  
FROM EMPLOYEE AS E  
JOIN PAYSLIP AS PA ON E.Emp_ID = PA.Emp_ID  
WHERE Pay_date = (SELECT MAX(Pay_date) FROM PAYSLIP);
```

Output table:

Employee Name	Hours paid	Amount Paid
Jessica Lin	\$45.00	\$7200.00
Tom Musk	\$45.00	\$7200.00
Petter Parker	\$45.00	\$7200.00
June LY	\$45.00	\$7200.00
Kong Bing	\$45.00	\$7200.00

Query 11:

-- XI. Make a list of all "Medium" size Shirts and their price in ascending order of price.
Show the product description and price

```
SELECT Prod_Desc AS 'Product Description', CONCAT('$', Prod_Price) AS 'Price'
FROM PRODUCT
WHERE Prod_Desc LIKE '%Shirt - M'
ORDER BY Prod_Price ASC;
```

Output table:

				Product Description	Price
<input type="checkbox"/>	Edit	Copy	Delete	Men Shirt - M	\$11.00
<input type="checkbox"/>	Edit	Copy	Delete	Women Shirt - M	\$16.00

Query 12:

-- XII.A list of supervisors (employee id, name) and all of their subordinates (employees id, name)

```
SELECT E.Emp_ID AS 'Employee ID', CONCAT(E.Emp_FName, ' ', E.Emp_LName) AS
'Employee Name', SUP.Emp_ID AS 'Supervisor ID', CONCAT(SUP.Emp_FName, '
', SUP.Emp_LName) AS 'Supervisor Name'
FROM EMPLOYEE AS E
JOIN EMPLOYEE AS SUP ON E.SupervisorID = SUP.Emp_ID;
```

Output table:

+ Options			
Employee ID	Employee Name	Supervisor ID	Supervisor Name
2	Clark Kent	1	Jessica Lin
3	Barry Allen	1	Jessica Lin
4	John Gardner	1	Jessica Lin
5	Joseph Smith	1	Jessica Lin
6	Ayaka Kin	1	Jessica Lin
7	Touya Hima	1	Jessica Lin
9	Debra Morgant	8	Tom Musk
10	Cole Mac	8	Tom Musk
11	Richard Homan	8	Tom Musk
12	Stracey Ly	8	Tom Musk
13	Rock Lee	8	Tom Musk
14	Steve Rogger	8	Tom Musk
16	Tom Cruise	15	Petter Parker
17	Himaru Ishin	15	Petter Parker
18	Ichigo Morgant	15	Petter Parker
19	Logan Lee	15	Petter Parker
20	Alice Cook	15	Petter Parker
21	Joma Homa	15	Petter Parker
23	Tele Cross	22	June LY
24	Winter Soldier	22	June LY
25	Big Mark	22	June LY
26	Hero Gardner	22	June LY
27	Char Kara	22	June LY
28	Minghun Lee	22	June LY
30	Yui Yuigahama	29	Kong Bing

+ Options

Employee ID	Employee Name	Supervisor ID	Supervisor Name
31	Sheldon Tien	29	Kong Bing
32	Inoha Shiha	29	Kong Bing
33	Paige Gate	29	Kong Bing
34	Jordan Buffet	29	Kong Bing
35	Hachiman Hikigaya	29	Kong Bing

Bibliography

Coronel. C., & Morris. S. (2017). *Database systems: Design, implementation, and management* (12th ed.). Retrieved from <http://www.cengage.com>