**Part1**

Q1.1

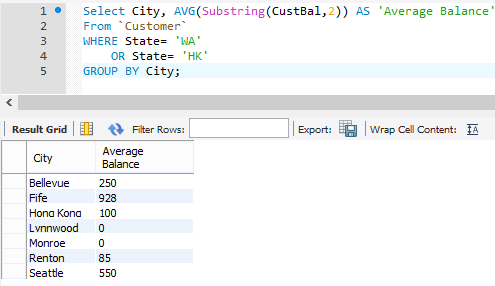
Select City, AVG(Substring(CustBal,2)) AS 'Average Balance'

From `Customer`

WHERE State= 'WA'

OR State= 'HK'

GROUP BY City;



Q1.2.

Select City, AVG(Substring(CustBal,2)) AS 'Average Balance' ,

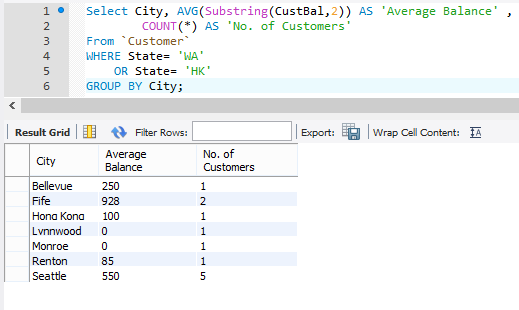
COUNT(\*) AS 'No. of Customers'

From `Customer`

WHERE State= 'WA'

OR State= 'HK'

GROUP BY City;



Q1.3.

Select OrderNo, (Sum(SubString(Price,2) \* Qty)) AS 'Amount'

From `Order` INNER JOIN

(`Product` INNER JOIN

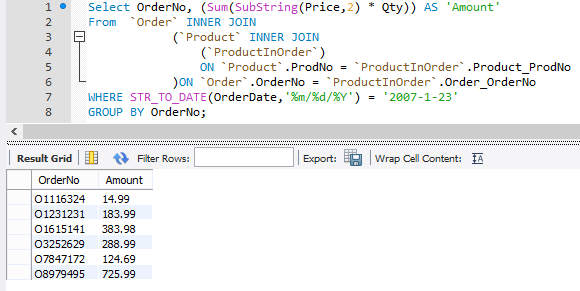
(`ProductInOrder`)

ON `Product`.ProdNo = `ProductInOrder`.Product\_ProdNo

)ON `Order`.OrderNo = `ProductInOrder`.Order\_OrderNo

WHERE STR\_TO\_DATE(OrderDate,'%m/%d/%Y') = '2007-1-23'

GROUP BY OrderNo;



Q1.4

Select OrderNo, Customer.FirstName, Customer.LastName, (Sum(SubString(Price,2) \* Qty)) AS 'Amount'

From `Customer` INNER JOIN

(`Order` INNER JOIN

(`Product` INNER JOIN

(`ProductInOrder`)

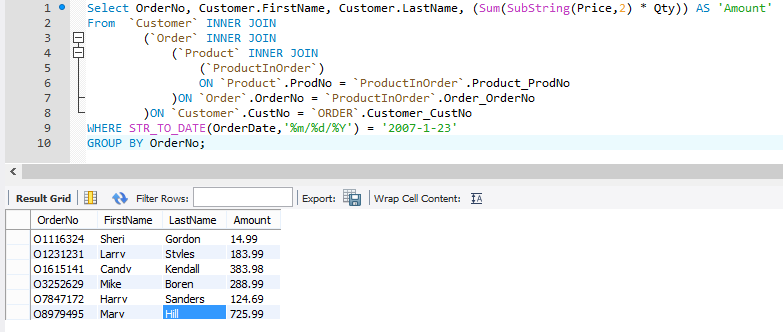
ON `Product`.ProdNo = `ProductInOrder`.Product\_ProdNo

)ON `Order`.OrderNo = `ProductInOrder`.Order\_OrderNo

)ON `Customer`.CustNo = `ORDER`.Customer\_CustNo

WHERE STR\_TO\_DATE(OrderDate,'%m/%d/%Y') = '2007-1-23'

GROUP BY OrderNo;



Q1.5.

Select CustNo, Customer.FirstName, Customer.LastName,

Sum(Qty) AS 'No. of Products',(Sum(SubString(Price,2) \* Qty)) AS 'Cost'

From `Customer` INNER JOIN

(`Order` INNER JOIN

(`Product` INNER JOIN

(`ProductInOrder`)

ON `Product`.ProdNo = `ProductInOrder`.Product\_ProdNo

)ON `Order`.OrderNo = `ProductInOrder`.Order\_OrderNo

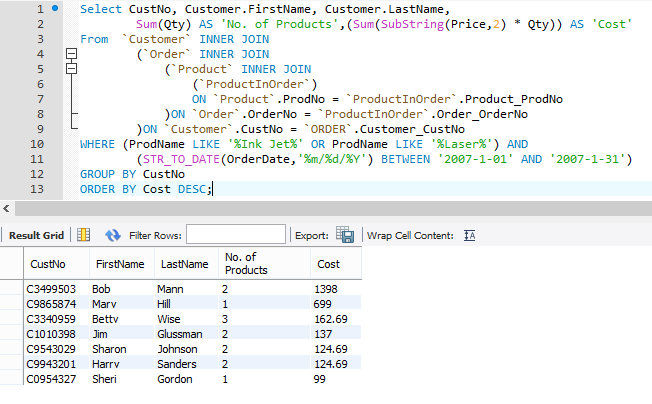
)ON `Customer`.CustNo = `ORDER`.Customer\_CustNo

WHERE (ProdName LIKE '%Ink Jet%' OR ProdName LIKE '%Laser%') AND

(STR\_TO\_DATE(OrderDate,'%m/%d/%Y') BETWEEN '2007-1-01' AND '2007-1-31')

GROUP BY CustNo

ORDER BY Cost DESC;



Q1.6

Select ProdNo, ProdName, Sum(Qty) AS 'Total Quantity',

(Sum(SubString(Price,2) \* Qty)) AS 'Cost'

From `Customer` INNER JOIN

(`Order` INNER JOIN

(`Product` INNER JOIN

(`ProductInOrder`)

ON `Product`.ProdNo = `ProductInOrder`.Product\_ProdNo

)ON `Order`.OrderNo = `ProductInOrder`.Order\_OrderNo

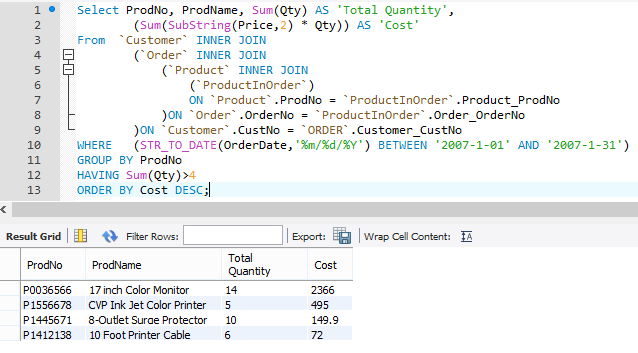
)ON `Customer`.CustNo = `ORDER`.Customer\_CustNo

WHERE (STR\_TO\_DATE(OrderDate,'%m/%d/%Y') BETWEEN '2007-1-01' AND '2007-1-31')

GROUP BY ProdNo

HAVING Sum(Qty)>4

ORDER BY Cost DESC;



Q1.7.

Select CustNo, Customer.FirstName, Customer.LastName,

COUNT(DISTINCT OrderNo) AS 'No. of Orders'

From `Customer` INNER JOIN

(`Order` INNER JOIN

(`Product` INNER JOIN

(`ProductInOrder`)

ON `Product`.ProdNo = `ProductInOrder`.Product\_ProdNo

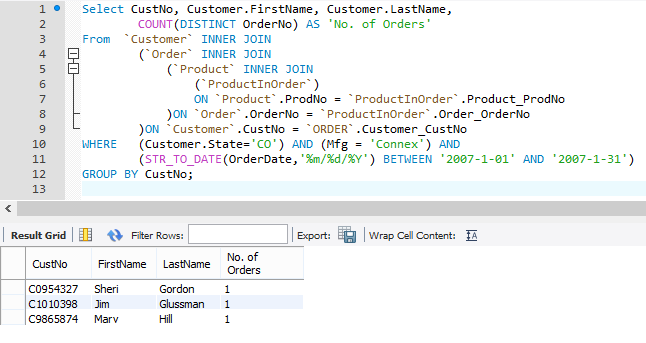
)ON `Order`.OrderNo = `ProductInOrder`.Order\_OrderNo

)ON `Customer`.CustNo = `ORDER`.Customer\_CustNo

WHERE (Customer.State='CO') AND (Mfg = 'Connex') AND

(STR\_TO\_DATE(OrderDate,'%m/%d/%Y') BETWEEN '2007-1-01' AND '2007-1-31')

GROUP BY CustNo;



Q1.8.

Select MONTHNAME(STR\_TO\_DATE(OrderDate,'%m/%d/%Y')) AS 'Month', (Sum(SubString(Price,2) \* Qty)) AS 'Amount'

From `Order` INNER JOIN

(`Product` INNER JOIN

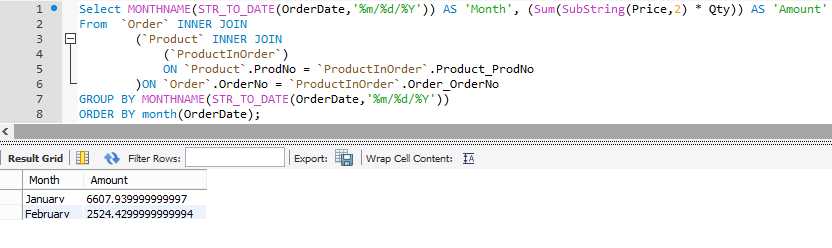
(`ProductInOrder`)

ON `Product`.ProdNo = `ProductInOrder`.Product\_ProdNo

)ON `Order`.OrderNo = `ProductInOrder`.Order\_OrderNo

GROUP BY MONTHNAME(STR\_TO\_DATE(OrderDate,'%m/%d/%Y'))

ORDER BY month(OrderDate);



**Part2**

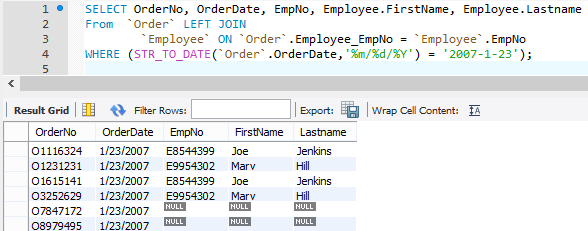
2.1.

SELECT OrderNo, OrderDate, EmpNo, Employee.FirstName, Employee.Lastname

From `Order` LEFT JOIN

`Employee` ON `Order`.Employee\_EmpNo = `Employee`.EmpNo

WHERE (STR\_TO\_DATE(`Order`.OrderDate,'%m/%d/%Y') = '2007-1-23');



2.2

SELECT OrderNo, OrderDate, EmpNo, Employee.FirstName, Employee.LastName,

CustNo, Customer.FirstName, Customer.Lastname

From `Order` LEFT JOIN

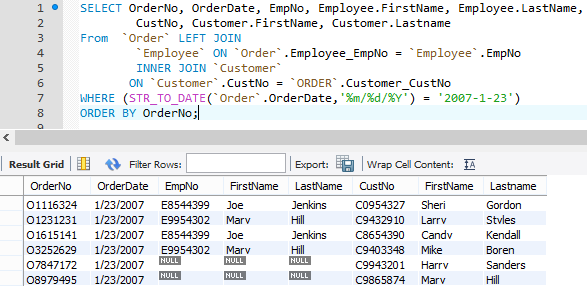
`Employee` ON `Order`.Employee\_EmpNo = `Employee`.EmpNo

INNER JOIN `Customer`

ON `Customer`.CustNo = `ORDER`.Customer\_CustNo

WHERE (STR\_TO\_DATE(`Order`.OrderDate,'%m/%d/%Y') = '2007-1-23')

ORDER BY OrderNo;



Q2.3.

SELECT \* FROM `Customer` LEFT JOIN `Employee`

ON (`Customer`.FirstName = `Employee`.FirstName

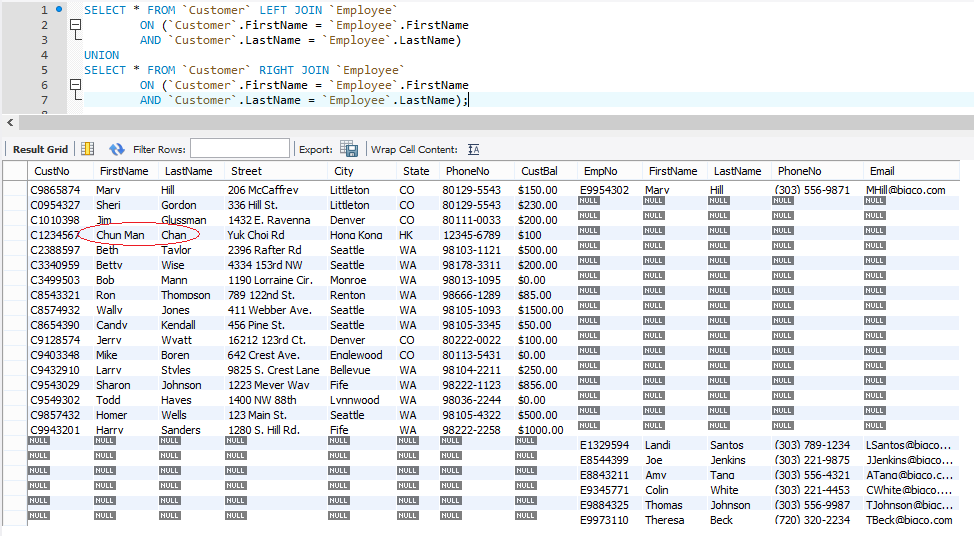
AND `Customer`.LastName = `Employee`.LastName)

UNION

SELECT \* FROM `Customer` RIGHT JOIN `Employee`

ON (`Customer`.FirstName = `Employee`.FirstName

AND `Customer`.LastName = `Employee`.LastName);



**Part3**

3.1

Select CustNo, FirstName, LastName,City

FROM `Customer`

WHERE (SUBSTRING(Custbal,2)>150) AND

Customer.CustNo IN

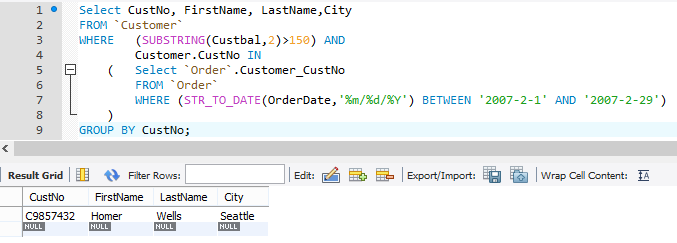
( Select `Order`.Customer\_CustNo

FROM `Order`

WHERE (STR\_TO\_DATE(OrderDate,'%m/%d/%Y') BETWEEN '2007-2-1' AND '2007-2-29')

)

GROUP BY CustNo;



3.2

Select CustNo, FirstName, LastName

FROM `Customer`

WHERE (State = 'CO') AND

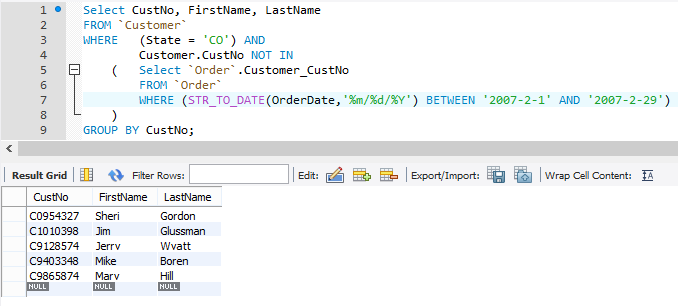
Customer.CustNo NOT IN

( Select `Order`.Customer\_CustNo

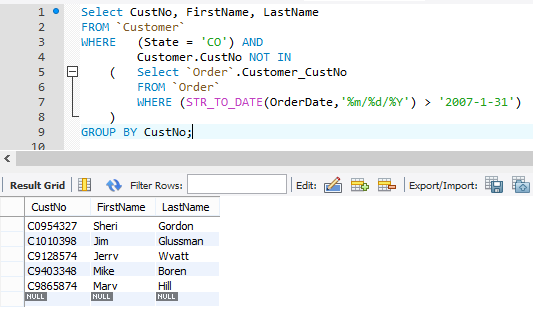
FROM `Order`

WHERE (STR\_TO\_DATE(OrderDate,'%m/%d/%Y') BETWEEN '2007-2-1' AND '2007-2-29')

)

GROUP BY CustNo;

3.3



Q3.3

SELECT OrderNo, OrderDate, CustName

From `Order`

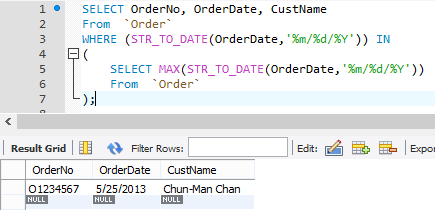
WHERE (STR\_TO\_DATE(OrderDate,'%m/%d/%Y')) IN

(

SELECT MAX(STR\_TO\_DATE(OrderDate,'%m/%d/%Y'))

From `Order`

);



**Part4**

DELIMITER $$

DROP PROCEDURE IF EXISTS add\_customer $$

CREATE PROCEDURE add\_customer(IN p\_CustNo CHAR(8), IN p\_LastName VARCHAR(45), IN p\_FirstName VARCHAR(45), OUT p\_Message VARCHAR(45))

BEGIN

DECLARE error\_key INT DEFAULT 0;

BEGIN

DECLARE EXIT HANDLER FOR 1062

SET error\_key = 1062;

DECLARE EXIT HANDLER FOR 1048

SET error\_key = 1048;

INSERT INTO `Customer` (CustNo, FirstName, LastName)

VALUES (p\_CustNo, p\_FirstName, p\_LastName);

END;

CASE

WHEN error\_key = 1062

THEN SET P\_Message = CONCAT('Duplicated key for Customer ', p\_CustNo);

WHEN error\_key = 1048

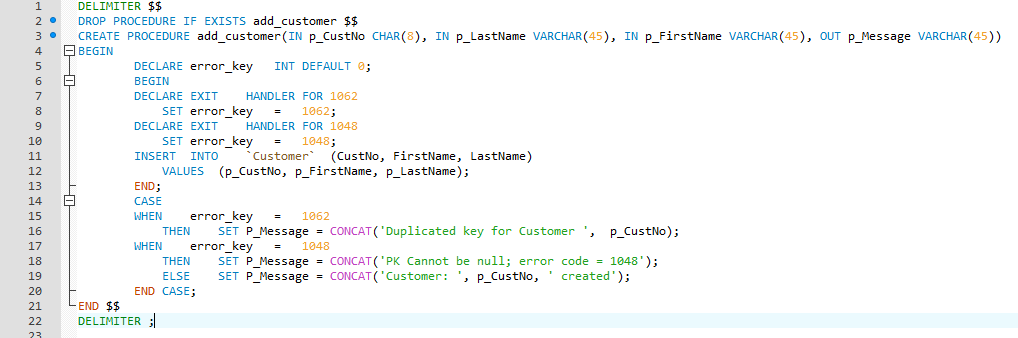
THEN SET P\_Message = CONCAT('PK Cannot be null; error code = 1048');

ELSE SET P\_Message = CONCAT('Customer: ', p\_CustNo, ' created');

END CASE;

END $$

DELIMITER ;

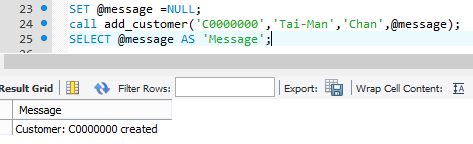


Q4.1(i)

SET @message =NULL;

call add\_customer('C0000000','Tai-Man','Chan',@message);

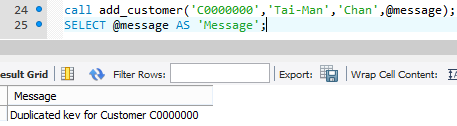
SELECT @message AS 'Message';



Q4.1(ii)

call add\_customer('C0000000','Tai-Man','Chan',@message);

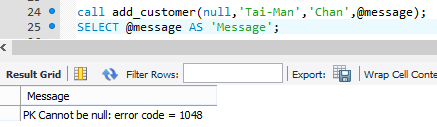
SELECT @message AS 'Message';



Q4.1(iii)

call add\_customer(null,'Tai-Man','Chan',@message);

SELECT @message AS 'Message';



Q4.2

DELIMITER $$

DROP TRIGGER IF EXISTS `update\_product` $$

CREATE TRIGGER `update\_product`

BEFORE UPDATE ON `Product`

FOR EACH ROW

BEGIN

DECLARE Price\_old DECIMAL(10, 2) DEFAULT 0;

DECLARE Price\_new DECIMAL(10, 2) DEFAULT 0;

DECLARE Stock\_diff DECIMAL(10, 2) DEFAULT 0;

DECLARE Revenue\_old DECIMAL(10, 2) DEFAULT 0;

DECLARE Revenue\_new DECIMAL(10, 2) DEFAULT 0;

SET Price\_old = SUBSTRING(OLD.Price, 2);

SET Stock\_diff = (NEW.stock - OLD.Stock) / OLD.Stock;

SET Price\_new = Price\_old - (Price\_old \* Stock\_diff);

SET Revenue\_old = (SELECT SUM(SUBSTRING(Price, 2) \* Stock) FROM `Product`);

SET Revenue\_new = Revenue\_old - (Price\_old \* OLD.Stock) + (Price\_New \* NEW.Stock);

IF (Revenue\_new > 17000)

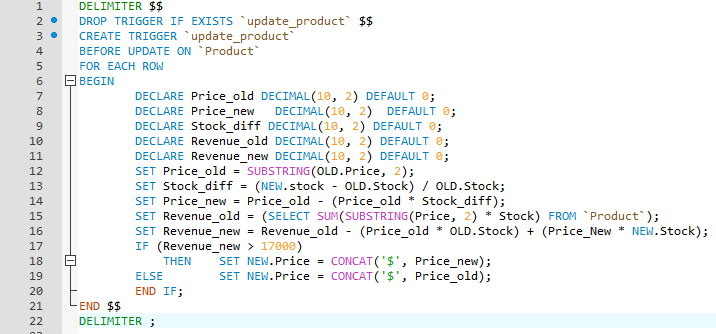
THEN SET NEW.Price = CONCAT('$', Price\_new);

ELSE SET NEW.Price = CONCAT('$', Price\_old);

END IF;

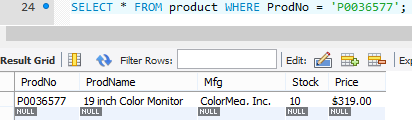
END $$

DELIMITER ;



Q4.2(i)

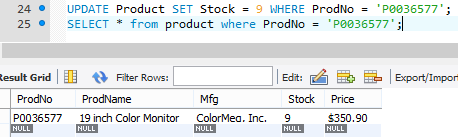
SELECT \* FROM product WHERE ProdNo = 'P0036577';



Q4.2(ii)

UPDATE Product SET Stock = 9 WHERE ProdNo = 'P0036577';

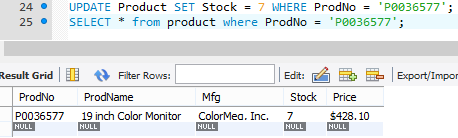
SELECT \* from product where ProdNo = 'P0036577';



Q4.2(iii)

UPDATE Product SET Stock = 7 WHERE ProdNo = 'P0036577';

SELECT \* from product where ProdNo = 'P0036577';



Q4.2(iv)

UPDATE Product SET Stock = 4 WHERE ProdNo = 'P0036577';

SELECT \* from product where ProdNo = 'P0036577';

