LiveStock

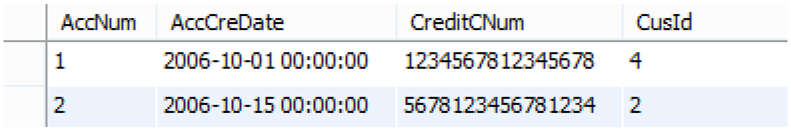
CSE 305: Assignment 2

Wendy Zheng: wendy.zheng@stonybrook.edu
Ryan Goepfert: ryan.goepfert@stonybrook.edu
Yuliya Astapova: yuliya.astapova@stonybrook.edu

TABLES WITH DEMO DATA

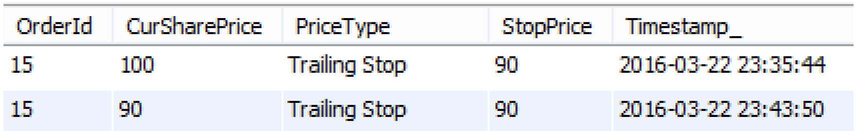
Our database auto-increments all id numbers, so the ids given in the demo data differ from the ones in our database. A customer with the id 'XXX-XX-XXXX' now has id 'X'. Likewise, each employee id, account number, order id, and transaction id auto-increment and are unique. As such, they may differ from the demo data. SSN, telephone number, and credit card no longer have hyphens. All dates have a time set to the time the data was entered, as our database stores both date and time for such fields.

Account



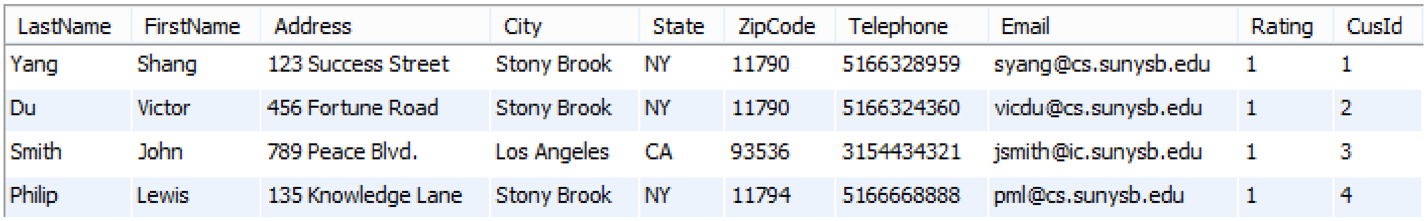
AccNum	AccCreDate	CreditCNum	CusId
1	2006-10-01 00:00:00	1234567812345678	4
2	2006-10-15 00:00:00	5678123456781234	2

ConditionalPriceHistory



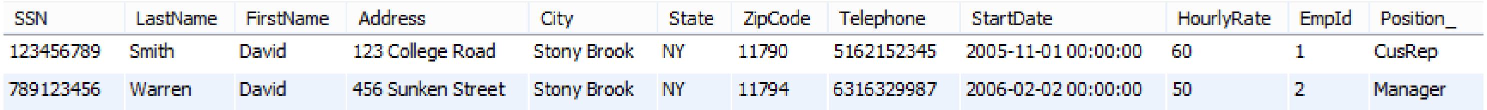
OrderId	CurSharePrice	PriceType	StopPrice	Timestamp_
15	100	Trailing Stop	90	2016-03-22 23:35:44
15	90	Trailing Stop	90	2016-03-22 23:43:50

Customer



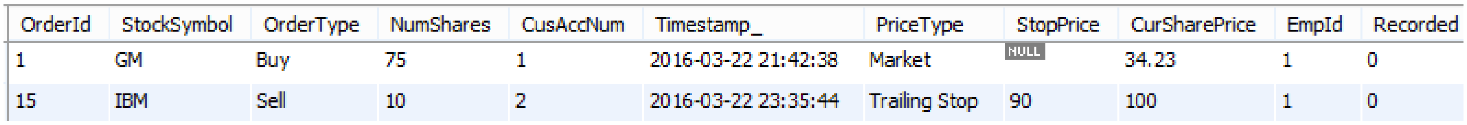
LastName	FirstName	Address	City	State	ZipCode	Telephone	Email	Rating	CusId
Yang	Shang	123 Success Street	Stony Brook	NY	11790	5166328959	syang@cs.sunysb.edu	1	1
Du	Victor	456 Fortune Road	Stony Brook	NY	11790	5166324360	vicdu@cs.sunysb.edu	1	2
Smith	John	789 Peace Blvd.	Los Angeles	CA	93536	3154434321	jsmith@ic.sunysb.edu	1	3
Philip	Lewis	135 Knowledge Lane	Stony Brook	NY	11794	5166668888	pml@cs.sunysb.edu	1	4

Employee



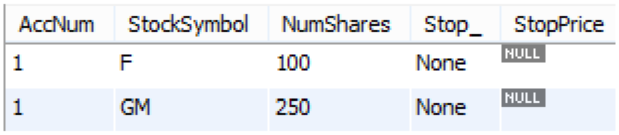
SSN	LastName	FirstName	Address	City	State	ZipCode	Telephone	StartDate	HourlyRate	EmpId	Position_
123456789	Smith	David	123 College Road	Stony Brook	NY	11790	5162152345	2005-11-01 00:00:00	60	1	CusRep
789123456	Warren	David	456 Sunken Street	Stony Brook	NY	11794	6316329987	2006-02-02 00:00:00	50	2	Manager

Order



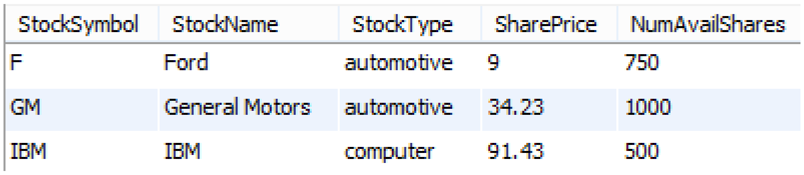
OrderId	StockSymbol	OrderType	NumShares	CusAccNum	Timestamp_	PriceType	StopPrice	CurSharePrice	EmpId	Recorded
1	GM	Buy	75	1	2016-03-22 21:42:38	Market	NULL	34.23	1	0
15	IBM	Sell	10	2	2016-03-22 23:35:44	Trailing Stop	90	100	1	0

Portfolio



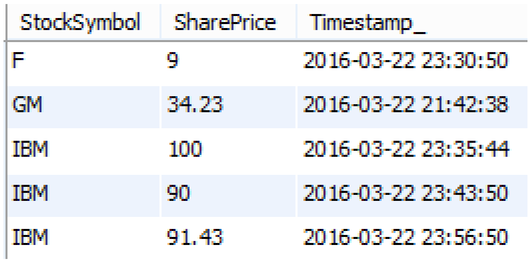
AccNum	StockSymbol	NumShares	Stop_	StopPrice
1	F	100	None	NULL
1	GM	250	None	NULL

Stock



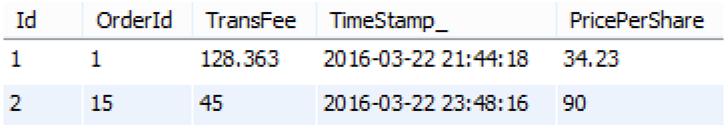
StockSymbol	StockName	StockType	SharePrice	NumAvailShares
F	Ford	automotive	9	750
GM	General Motors	automotive	34.23	1000
IBM	IBM	computer	91.43	500

StockPriceHistory



StockSymbol	SharePrice	Timestamp_
F	9	2016-03-22 23:30:50
GM	34.23	2016-03-22 21:42:38
IBM	100	2016-03-22 23:35:44
IBM	90	2016-03-22 23:43:50
IBM	91.43	2016-03-22 23:56:50

Transact



Id	OrderId	TransFee	TimeStamp_	PricePerShare
1	1	128.363	2016-03-22 21:44:18	34.23
2	15	45	2016-03-22 23:48:16	90

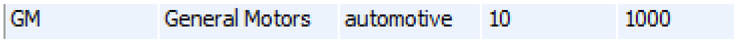
TRANSACTIONS

All fields and values starting with ':' are parameters to be entered by the user. The argument placeholder denotes the required input data. Please note that the manager level transactions dealing with reports have mostly empty views because none of the orders have been recorded yet; therefore they are not considered "reviewed" and do not appear on the resulting table.

Manager-Level Transactions

- Set the share price of a stock (for simulating market fluctuations in a stock's share price)

```
UPDATE STOCK
SET SharePrice = :share_price
WHERE StockSymbol = :stock_symbol

UPDATE STOCK
SET SharePrice = 10
WHERE StockSymbol = 'GM'
```

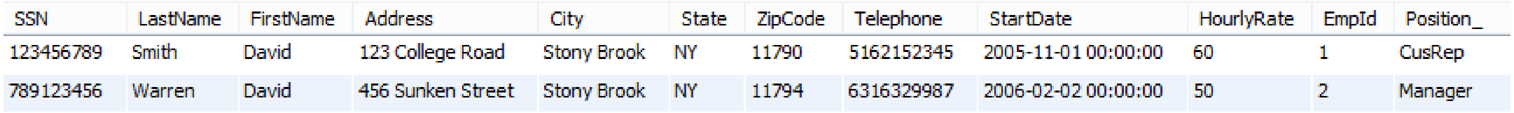


GM	General Motors	automotive	10	1000

- Add, Edit and Delete information for an employee

```
INSERT INTO EMPLOYEE(
SSN, LastName, FirstName, Address, City, State, ZipCode, Telephone, StartDate, HourlyRate, Position)
VALUES
(:ssn, :last_name, :first_name, :address, :city, :state, :zipcode, :telephone, :start_date, :hourly_rate, :position)
```

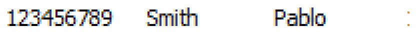
```
INSERT INTO EMPLOYEE(
SSN, LastName, FirstName, Address, City, State, ZipCode, Telephone, StartDate, HourlyRate, Position_)
VALUES ('123456789', 'Smith', 'David', '123 College Road', 'Stony Brook', 'NY', '11790', '5162152345',
'2005-11-01 00:00:00', 60, 'CusRep');
```



SSN	LastName	FirstName	Address	City	State	ZipCode	Telephone	StartDate	HourlyRate	EmpId	Position_
123456789	Smith	David	123 College Road	Stony Brook	NY	11790	5162152345	2005-11-01 00:00:00	60	1	CusRep
789123456	Warren	David	456 Sunken Street	Stony Brook	NY	11794	6316329987	2006-02-02 00:00:00	50	2	Manager

```
UPDATE EMPLOYEE
SET :field = :value
WHERE EmpId = :emp_id
```

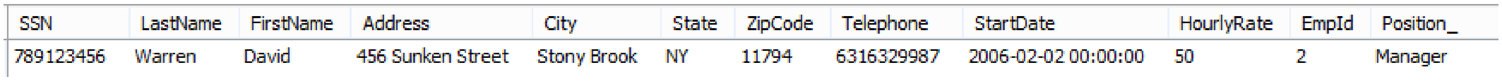
```
UPDATE EMPLOYEE
SET FirstName = 'Pablo'
WHERE EmpId = 1
```



123456789	Smith	Pablo

```
DELETE FROM EMPLOYEE
WHERE EmpId = :emp_id
```

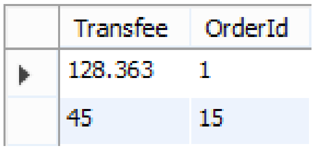
```
DELETE FROM Employee
WHERE EmpId = 1
```



SSN	LastName	FirstName	Address	City	State	ZipCode	Telephone	StartDate	HourlyRate	EmpId	Position_
789123456	Warren	David	456 Sunken Street	Stony Brook	NY	11794	6316329987	2006-02-02 00:00:00	50	2	Manager

- Obtain a sales report for a particular month

```
SELECT T.Transfee, T.OrderId
FROM TRANSACT T
WHERE YEAR(T.Timestamp) = :year
AND MONTH(T.Timestamp) = :month
```

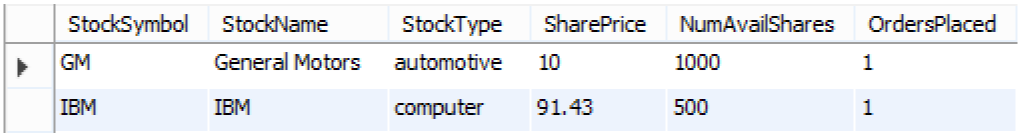


Transfee	OrderId
128.363	1
45	15

```
SELECT T.Transfee, T.OrderId
FROM TRANSACT T
WHERE YEAR(T.Timestamp) = 2016
AND MONTH(T.Timestamp) = 03
```

- Produce a comprehensive listing of all stocks

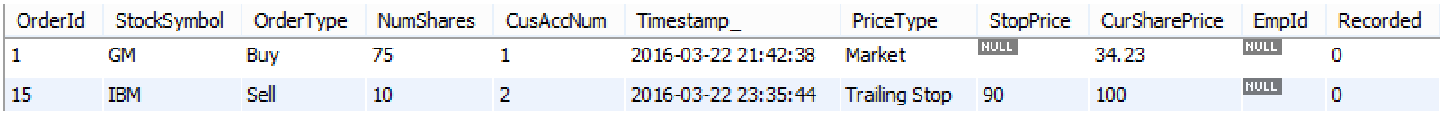
```
SELECT S.*, COUNT (O.StockSymbol) AS OrdersPlaced
FROM STOCK S, ORDER O
WHERE S.StockSymbol = O.StockSymbol
GROUP BY S.StockSymbol
```



StockSymbol	StockName	StockType	SharePrice	NumAvailShares	OrdersPlaced
GM	General Motors	automotive	10	1000	1
IBM	IBM	computer	91.43	500	1

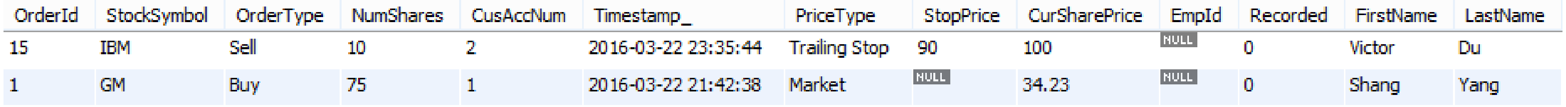
- Produce a list of orders by stock symbol or by customer name

```
SELECT *
FROM ORDER_ O
ORDER BY O.StockSymbol
```



OrderId	StockSymbol	OrderType	NumShares	CusAccNum	Timestamp_	PriceType	StopPrice	CurSharePrice	EmpId	Recorded
1	GM	Buy	75	1	2016-03-22 21:42:38	Market	NULL	34.23	NULL	0
15	IBM	Sell	10	2	2016-03-22 23:35:44	Trailing Stop	90	100	NULL	0

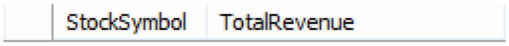
```
SELECT O.*, C.FirstName, C.LastName
FROM ORDER O, CUSTOMER C
WHERE O.CusAccNum = C.CusId
ORDER BY C.LastName, C.FirstName
```



OrderId	StockSymbol	OrderType	NumShares	CusAccNum	Timestamp_	PriceType	StopPrice	CurSharePrice	EmpId	Recorded	FirstName	LastName
15	IBM	Sell	10	2	2016-03-22 23:35:44	Trailing Stop	90	100	NULL	0	Victor	Du
1	GM	Buy	75	1	2016-03-22 21:42:38	Market	NULL	34.23	NULL	0	Shang	Yang

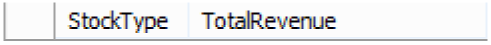
- Produce a summary listing of revenue generated by a particular stock, stock type, or customer

```
CREATE VIEW STOCKREVENUE (StockSymbol, TotalRevenue) AS
        SELECT O.StockSymbol, SUM(T.TransFee)
        FROM Order O, Transact T
        WHERE O.Recorded = 1 AND O.OrderId = T.OrderId
        GROUP BY O.StockSymbol
```



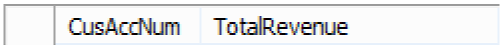
StockSymbol	TotalRevenue

```
CREATE VIEW STOCKTYPEREVENUE (StockType, TotalRevenue) AS
        SELECT S.StockType, SUM(T.TransFee)
        FROM Stock S, Transact T, Order O
        WHERE O.StockSymbol = S.StockSymbol AND O.Recorded = 1
        AND O.OrderID=T.OrderId
        GROUP BY S.StockType
```



StockType	TotalRevenue

```
CREATE VIEW CUSTOMERREVENUE(CusAccNum, TotalRevenue) AS
    SELECT O.CusAccNum, SUM(T.TransFee) AS TotalRevenue
    FROM Order O, Transact T
    WHERE O.Recorded = 1 AND O.OrderID = T.OrderId
    GROUP BY O.CusAccNum
```



CusAccNum	TotalRevenue

- Determine which customer representative generated most total revenue

```
CREATE VIEW EMPLOYEEREVENUE(EmpId, TotalRevenue) AS
    SELECT O.EmpId, SUM(T.TransFee) AS TotalRevenue
    FROM ORDER O, Transact T
    WHERE O.OrderID=T.OrderId AND O.Recorded = 1 AND O.EmpID <> 0
    GROUP BY O.EmpId
```

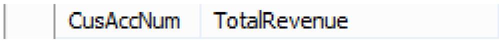
EmpId	TotalRevenue

```
SELECT E.EmpId, E.TotalRevenue
    FROM EMPLOYEEREVENUE E
    WHERE  E.TotalRevenue  =
        (SELECT MAX(E.TotalRevenue)
        FROM EMPLOYEEREVENUE E)
```

EmpId	TotalRevenue

- Determine which customer generated most total revenue

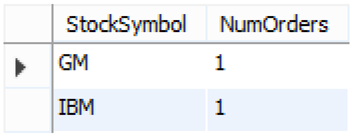
```
SELECT C.CusAccNum, C.TotalRevenue
    FROM CUSTOMERREVENUE C
    WHERE C.TotalRevenue =
    (SELECT MAX(C.TotalRevenue)
    FROM CUSTOMEREVENUE C)
```



CusAccNum	TotalRevenue

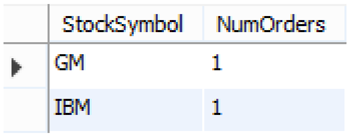
- Produce a list of most actively traded stocks

```
CREATE VIEW MostTraded(StockSymbol, NumOrders) AS
    SELECT O.StockSymbol, COUNT(*) AS NumOrders
    FROM Order O
    GROUP BY O.StockSymbol
```



StockSymbol	NumOrders
GM	1
IBM	1

```
SELECT M.StockSymbol, M.NumOrders
        FROM MostTraded M
        ORDER BY ASC NumOrders
        LIMIT 25
```

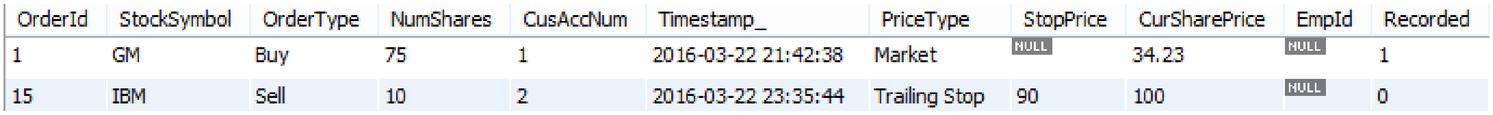


StockSymbol	NumOrders
GM	1
IBM	1

Customer-Representative-Level Transactions

- Record an order

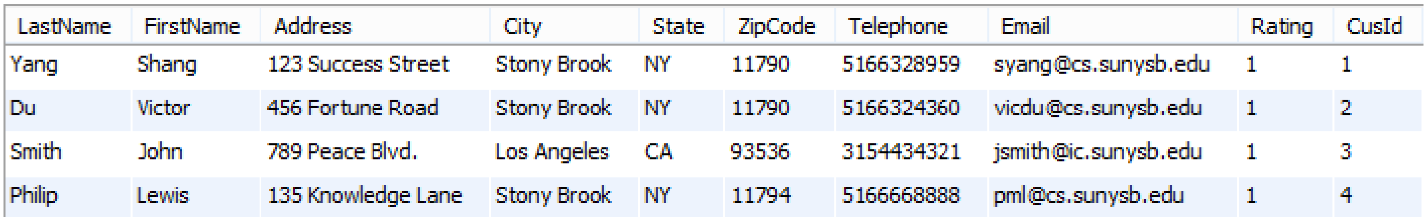
```
UPDATE ORDER O
SET Recorded = 1
WHERE O.OrderId = :order_id

UPDATE ORDER O
SET Recorded = 1
WHERE O.OrderId = 1
```



OrderId	StockSymbol	OrderType	NumShares	CusAccNum	Timestamp_	PriceType	StopPrice	CurSharePrice	EmpId	Recorded
1	GM	Buy	75	1	2016-03-22 21:42:38	Market	NULL	34.23	NULL	1
15	IBM	Sell	10	2	2016-03-22 23:35:44	Trailing Stop	90	100	NULL	0

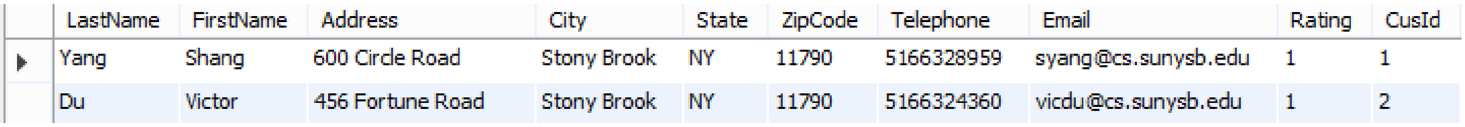
- Add, Edit and Delete information for a customer

```
INSERT INTO CUSTOMER(
LastName, FirstName, Address, City, State, ZipCode, Telephone, Email, Rating, CusId)
VALUES (
:last_name, :first_name, :address, :city, :state, :zipcode, :telephone, :email, :rating, :cus_id)

INSERT INTO Customer(
LastName, FirstName, Address, City, State, ZipCode, Telephone, Email, Rating)
VALUES (
'Yang', 'Shang', '123 Success Street', 'Stony Brook', 'NY', '11790', '5166328959', 'syang@cs.sunysb.edu',
1);
```



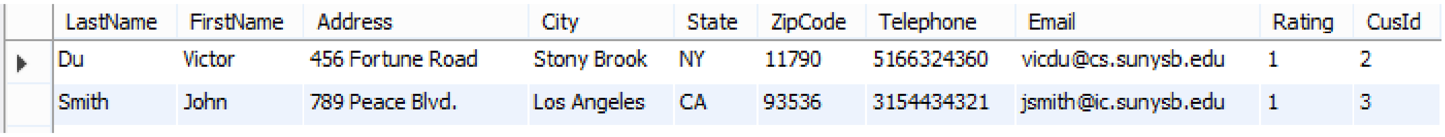
LastName	FirstName	Address	City	State	ZipCode	Telephone	Email	Rating	CusId
Yang	Shang	123 Success Street	Stony Brook	NY	11790	5166328959	syang@cs.sunysb.edu	1	1
Du	Victor	456 Fortune Road	Stony Brook	NY	11790	5166324360	vicdu@cs.sunysb.edu	1	2
Smith	John	789 Peace Blvd.	Los Angeles	CA	93536	3154434321	jsmith@ic.sunysb.edu	1	3
Philip	Lewis	135 Knowledge Lane	Stony Brook	NY	11794	5166668888	pml@cs.sunysb.edu	1	4

```
UPDATE CUSTOMER C
SET field = :value
WHERE CusId = :cus_id

UPDATE CUSTOMER C
SET C.Address = '600 Circle Road'
WHERE c.CusId = 1
```



LastName	FirstName	Address	City	State	ZipCode	Telephone	Email	Rating	CusId
Yang	Shang	600 Circle Road	Stony Brook	NY	11790	5166328959	syang@cs.sunysb.edu	1	1
Du	Victor	456 Fortune Road	Stony Brook	NY	11790	5166324360	vicdu@cs.sunysb.edu	1	2

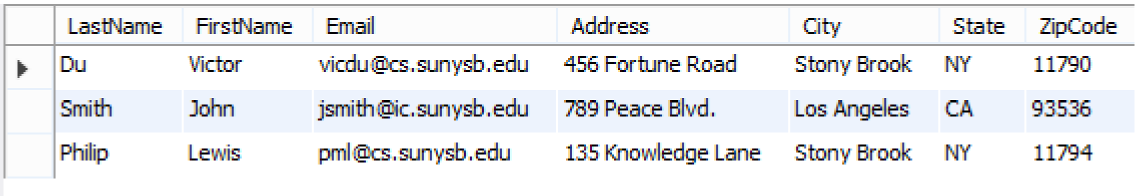
```
DELETE FROM CUSTOMER
WHERE CusId = :cus_id

DELETE FROM CUSTOMER
WHERE CusId = 1
```



LastName	FirstName	Address	City	State	ZipCode	Telephone	Email	Rating	CusId
Du	Victor	456 Fortune Road	Stony Brook	NY	11790	5166324360	vicdu@cs.sunysb.edu	1	2
Smith	John	789 Peace Blvd.	Los Angeles	CA	93536	3154434321	jsmith@ic.sunysb.edu	1	3

- Produce customer mailing lists

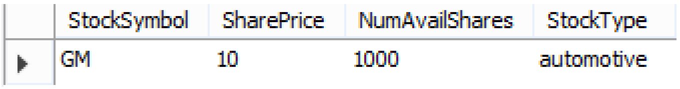
```
SELECT LastName, FirstName, Email, Address, City, State, ZipCode
FROM CUSTOMER
```



LastName	FirstName	Email	Address	City	State	ZipCode
Du	Victor	vicdu@cs.sunysb.edu	456 Fortune Road	Stony Brook	NY	11790
Smith	John	jsmith@ic.sunysb.edu	789 Peace Blvd.	Los Angeles	CA	93536
Philip	Lewis	pml@cs.sunysb.edu	135 Knowledge Lane	Stony Brook	NY	11794

- Produce a list of stock suggestions for a given customer (based on that customer's past orders)

```
CREATE VIEW Suggest(StockSymbol, SharePrice, NumAvailShares, StockType) AS
        SELECT S.StockSymbol, S.SharePrice, S.NumAvailShares, S.StockType
        FROM Stock S, Order_ O
        WHERE (O.CusAccNum = :_cusid AND O.StockSymbol = S.StockSymbol)

CREATE VIEW Suggest(StockSymbol, SharePrice, NumAvailShares, StockType) AS
        SELECT S.StockSymbol, S.SharePrice, S.NumAvailShares, S.StockType
        FROM Stock S, Order_ O
        WHERE (O.CusAccNum = 1 AND O.StockSymbol = S.StockSymbol)
```

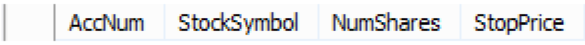


StockSymbol	SharePrice	NumAvailShares	StockType
GM	10	1000	automotive

Customer-Level Transactions

- A customer's current stock holdings

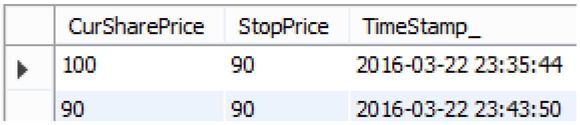
```
SELECT P.AccNum, P.StockSymbol, P.NumShares, P.Stop, P.StopPrice
FROM Portfolio P, Account A, Customer C
WHERE P.AccNum = A.AccNum
          AND A.CusId = C.CusId
          AND C.CusId = :customer_id

SELECT P.AccNum, P.StockSymbol, P.NumShares, P.Stop, P.StopPrice
FROM Portfolio P, Account A, Customer C
WHERE P.AccNum = A.AccNum
          AND A.CusId = C.CusId
          AND C.CusId = 3
```



AccNum	StockSymbol	NumShares	StopPrice

- The share-price and trailing-stop history for a given conditional order

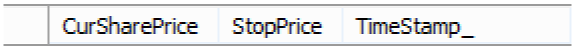
```
SELECT P.CurSharePrice, P.StopPrice, P.TimeStamp
FROM CONDITIONALPRICEHISTORY P
WHERE P.OrderId = :_orderid AND P.PriceType = ‘Trailing Stop’ AND P.StopPrice > 0

SELECT P.CurSharePrice, P.StopPrice, P.TimeStamp
FROM CONDITIONALPRICEHISTORY
WHERE P.OrderId = 15 AND P.PriceType = ‘Trailing Stop’ AND P.StopPrice > 0
```



CurSharePrice	StopPrice	TimeStamp_
100	90	2016-03-22 23:35:44
90	90	2016-03-22 23:43:50

- The share-price and hidden-stop history for a given conditional order

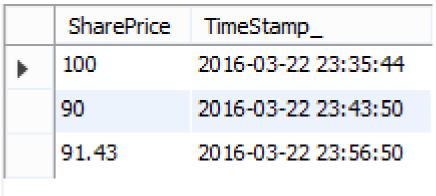
```
SELECT P.CurSharePrice, P.StopPrice, P.TimeStamp
FROM CONDITIONALPRICEHISTORY P, ORDER O
WHERE P.OrderId = :_orderid  AND P.PriceType = “Hidden Stop’ AND P.StopPrice > 0

SELECT P.CurSharePrice, P.StopPrice, P.TimeStamp
FROM CONDITIONALPRICEHISTORY P, ORDER O
WHERE P.OrderId = 15 P.PriceType = “Hidden Stop’ AND P.StopPrice > 0
```



CurSharePrice	StopPrice	TimeStamp_

- The share-price history of a given stock over a certain period of time (e.g., past six months)

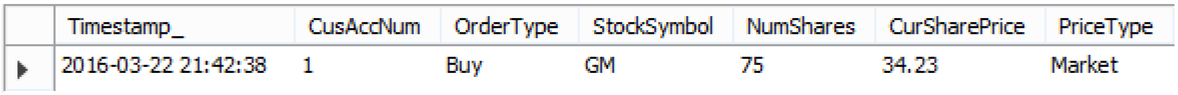
```
SELECT P.SharePrice, P.TimeStamp
FROM STOCKPRICEHISTORY P
WHERE P.StockSymbol = :stock_symbol
        AND MONTH(TIMEDIFF(NOW(), P.TimeStamp)) < :_month_past

SELECT P.SharePrice, P.TimeStamp
FROM STOCKPRICEHISTORY P
WHERE P.StockSymbol = ‘IBM’
        AND MONTH(TIMEDIFF(NOW(), P.TimeStamp)) < 6
```



	SharePrice	TimeStamp_
▶	100	2016-03-22 23:35:44
	90	2016-03-22 23:43:50
	91.43	2016-03-22 23:56:50

- A history of all current and past orders a customer has placed.

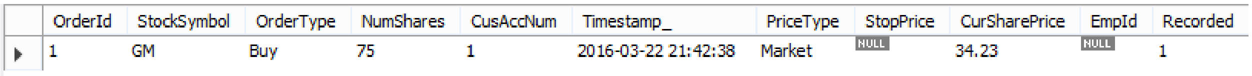
```
SELECT O.Timestamp, O.CusAccNum, O.OrderType, O.StockSymbol, O.NumShares, O.CurSharePrice,
        O.PriceType
FROM Order_ O, Customer C, Account A
WHERE O.CusAccNum = A.AccNum
        AND A.CusId = C.CusId AND C.CusId = :customer_id

SELECT O.Timestamp, O.CusAccNum, O.OrderType, O.StockSymbol, O.NumShares, O.CurSharePrice,
        O.PriceType
FROM Order_ O, Customer C, Account A
WHERE O.CusAccNum = A.AccNum
        AND A.CusId = C.CusId AND C.CusId = 4
```



	Timestamp_	CusAccNum	OrderType	StockSymbol	NumShares	CurSharePrice	PriceType
▶	2016-03-22 21:42:38	1	Buy	GM	75	34.23	Market

- Stocks available of a particular type and most-recent order info

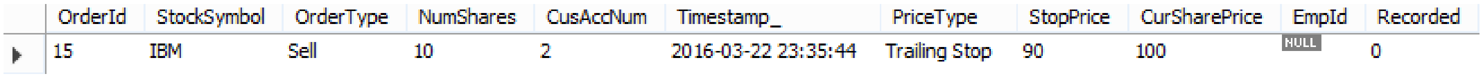
```
SELECT O.*
FROM STOCK S, Order_ O
WHERE S.StockType LIKE '%:_stocktype%' AND O.StockSymbol = S.StockSymbol

SELECT O.*
FROM STOCK S, Order_ O
WHERE S.StockType LIKE '%auto%' AND O.StockSymbol = S.StockSymbol
```



	OrderId	StockSymbol	OrderType	NumShares	CusAccNum	Timestamp_	PriceType	StopPrice	CurSharePrice	EmpId	Recorded
▶	1	GM	Buy	75	1	2016-03-22 21:42:38	Market	NULL	34.23	NULL	1

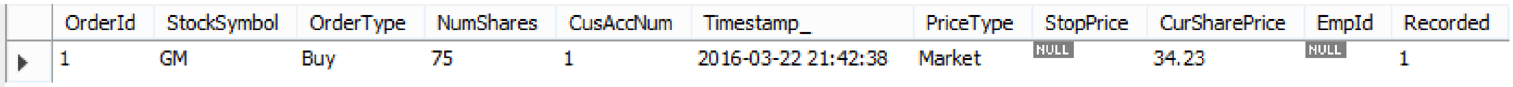
- Stocks available with a particular keyword or set of keywords in the stock name, and most-recent order info.

```
SELECT O.*
FROM STOCK S, Order_ O
WHERE S.StockSymbol LIKE '%:_stocksymbol%' AND O.StockSymbol = S.StockSymbol

SELECT O.*
FROM STOCK S, Order_ O
WHERE S.StockSymbol LIKE '%I%' AND O.StockSymbol = S.StockSymbol
```



OrderId	StockSymbol	OrderType	NumShares	CusAccNum	Timestamp_	PriceType	StopPrice	CurSharePrice	EmpId	Recorded
15	IBM	Sell	10	2	2016-03-22 23:35:44	Trailing Stop	90	100	NULL	0

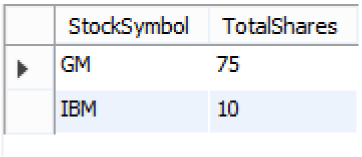
```
SELECT O.*
FROM ORDER_ O, CUSTOMER C, ACCOUNT_ A
WHERE O.CusAccNum = A.AccNum AND A.CusId = C.CusId AND C.CusId = 4
AND MONTH(TIMEDIFF(NOW(), O.Timestamp)) <= 3
```



OrderId	StockSymbol	OrderType	NumShares	CusAccNum	Timestamp_	PriceType	StopPrice	CurSharePrice	EmpId	Recorded
1	GM	Buy	75	1	2016-03-22 21:42:38	Market	NULL	34.23	NULL	1

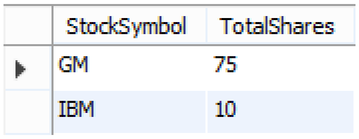
- Best-Seller list of stocks generated based on what stocks have been sold the most.

```
CREATE VIEW BestSellers(StockSymbol, TotalShares) AS
        SELECT O.StockSymbol, SUM(O.NumShares) AS TotalShares
        FROM Order O
        GROUP BY O.StockSymbol
```



StockSymbol	TotalShares
GM	75
IBM	10

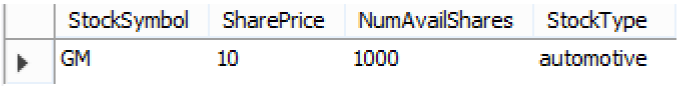
```
SELECT B.StockSymbol, B.TotalShares
        FROM BestSellers B
        ORDER BY DESC TotalShares
        LIMIT 25
```



StockSymbol	TotalShares
GM	75
IBM	10

- Personalized stock suggestion list. This finds stocks that are of the same type that the customer currently owns and stocks that the customer once owned, but no longer does.

```
CREATE VIEW Suggest(StockSymbol, SharePrice, NumAvailShares, StockType) AS
        SELECT S.StockSymbol, S.SharePrice, S.NumAvailShares, S.StockType
        FROM Stock S, Order_ O
        WHERE (O.CusAccNum = 1 AND O.StockSymbol = S.StockSymbol)
```



	StockSymbol	SharePrice	NumAvailShares	StockType
▶	GM	10	1000	automotive