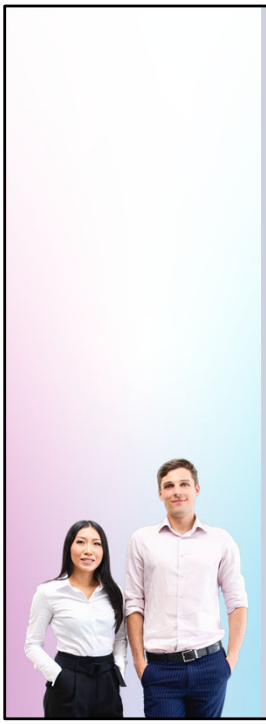


Mthree Alumni Training

SQL with Financial Examples

- Introductions to the finance data
- Mathematical calculations in SQL

SQL and Finance

Objectives

We will now use some of the things we have learnt to go through some finance examples with SQL as well as getting you familiar with some of the mathematical calculations MySQL allows you to do.

Trainer Notes

Here we will take some examples of SQL and practice on some financial data.

Flat Files

Fix.csv:

tag ,name
35 ,MsgType
39 ,OrdStatus
40 ,OrdType
54 ,Side

fix.csv:

tag ,name ,value ,descr
35 ,MsgType ,U ,A "U" as the first character(i.e. U1,U2...
39 ,OrdStatus ,0 ,New
40 ,OrdType ,1 ,Market
54 ,Side ,1,Buy

instrument.csv:

instrument ,ric ,isin ,sedol ,cusip ,bbid ,mic
III ,III.L ,GB00B1YW4409 ,B1YW440 , ,III:LN ,XLON
AAL ,AAL.L,GB00B1XZS820 ,B1XZS820 , ,AAL:LN ,XLON
,BP.N , , , , ,XNYS
, ,XS0248392812, , , ,

Trainer/Trainee Notes

Here is an example of our finance data stored in csv files. What do we think of this data structure?

flat=2 dimensional
csv=comma separated values
txt=tab separated values
fixed width is another flat file format

CSV and Excel

CSV files do have some advantages in that they translate easily into Excel, and Excel has an option that allows you to turn a CSV file into a spreadsheet.

This is extremely useful when pulling data out of a database or a log file that you then need to give to a business user.

Data stored in csv files allows you to parse the data easily with most programming languages.

As a data warehouse, it forms a simple flat schema.

Trainer/Trainee Notes

Often we would pull information out of a database into a csv which can be loaded into Excel by a business user.

Data beyond spreadsheets

Excel can operate in 2 modes:

- Local files such as .csv, ,txt, .xlsx
- ODBC/OLE DB

Local files became unmanageable when data becomes large (65k+) – the application becomes clunky and really slow.

A spreadsheet basically has a database underneath. Real databases can have multiple viewers including:

- Excel
- Web
- SQL tools (such as SQL Developer or mySQL)

A database allows one centralised copy with multiple user access.

Trainer/Trainee Notes

mthree

Kdb+

Developed and marketed by Kx systems, kdb+ is a column based relational time series database.

Typically it is used for bigdata, where the number of rows per day exceeds 10 million. Some typical use cases:

Orders/Fills | Trades/Quotes | Reference Data

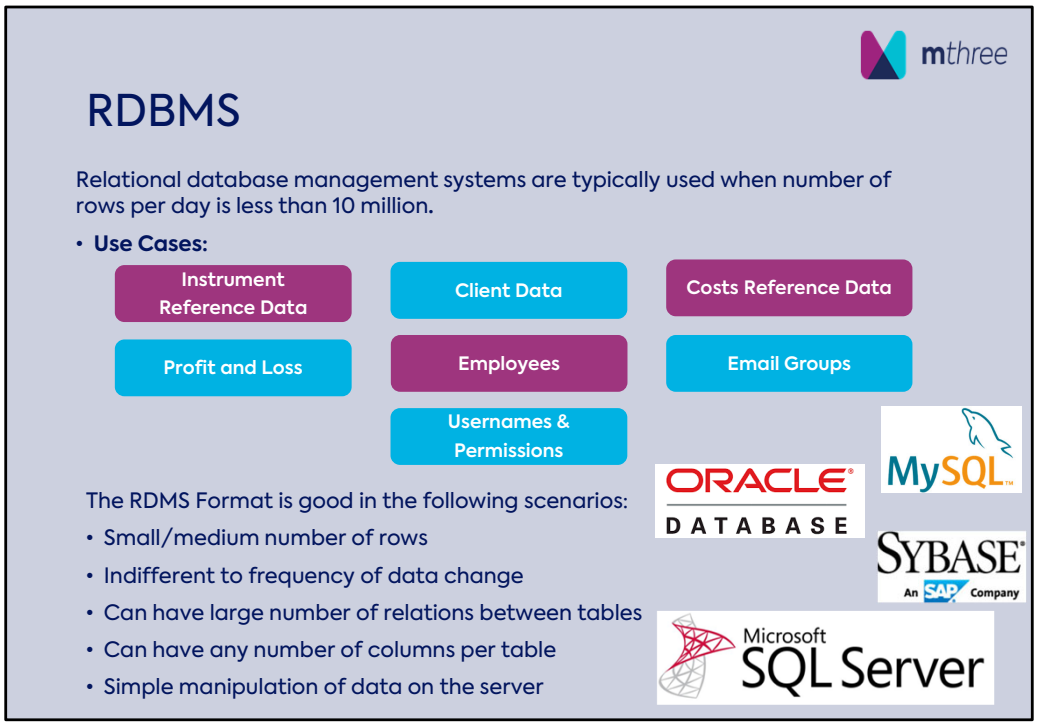
KDB is useful in the following scenarios:

- Large numbers of rows
- Historical data doesn't change often
- Few relations between tables
- Normal number of columns (e.g. 1-200)
- Powerful programming language allows for complex manipulation of data on server
- Not normally used for document storage

Trainer/Trainee Notes

As the database is high performance, it is highly useful in low latency trading situations as well as perfect for dealing with high volumes of market data. You will often see KDB databases used in conjunction with equity trading systems to hold and manage market data.

The database has also been used for other time-sensitive data applications including commodity markets such as energy trading, telecommunications, sensor data, log data, and machine and computer network usage monitoring.



Trainer/Trainee Notes

These will always be administered by a central DBA group within a big company, but you will have access to run queries on the data and potentially do some updates. Manipulation of production data is definitely something to discuss – and not something you should be doing unless in an outage for some reason as it is your golden source of data. What if it was payment information and you changed the account details?

Microsoft Access

Microsoft Access is also a RDBMS (relational database management system) from Microsoft.

- It combines the relational Microsoft Jet Database Engine with a GUI and software development tools, e.g. VBA
- It is bundled with Microsoft Office
- Not used as much as the rest of the office tools such as Word, Excel and PowerPoint
- Good for personal use but also can scale for enterprise or web solutions, using MS SQL Server

Trainer/Trainee Notes

For more details on Access you can find info on Wikipedia:
https://en.wikipedia.org/wiki/Microsoft_Access

Further SQL Concepts

As well as using our database management system to look at the makeup of our database, SQL provides us with a handy command to look at our tables in more depth.

describe instrument;

Field	Type	Null	Key	Default	Extra
ID	int	NO	PRI	NULL	
INSTRUMENT	varchar(30)	YES		NULL	
RIC	varchar(14)	YES		NULL	
ISIN	varchar(12)	YES		NULL	
SEDOL	varchar(9)	YES		NULL	
CUSIP	varchar(10)	YES		NULL	
BBID	varchar(12)	YES		NULL	
MIC	char(4)	YES		NULL	
NAME	varchar(50)	YES		NULL	
CIK	char(10)	YES		NULL	

Trainer/Trainee Notes

Here we get a view of the columns that are setup in the database table instrument, as well as the value type, whether it can be null and what is the primary key

The Orders Table

The orders table in the finance data describes a standard equity trading system

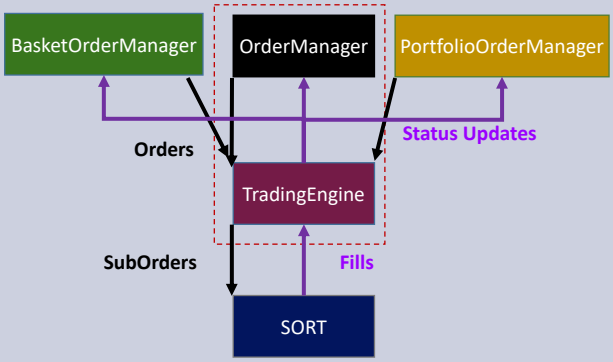
describe orders;

Start Page | ORDERS | Index

Columns | Data | Model | Constraints | Grants | Statistics

Actions...

	COLUMN_NAME	DATA_TYPE
1	ID	NUMBER(38,0)
2	DATETIME	TIMESTAMP(6)
3	CLIENTID	VARCHAR2(20 BYTE)
4	ROOTORDID	VARCHAR2(10 BYTE)
5	PARENTORDID	VARCHAR2(10 BYTE)
6	MSGTYPE	CHAR(2 BYTE)
7	ORDERID	VARCHAR2(10 BYTE)
8	SYSTEM	VARCHAR2(20 BYTE)
9	INSTRUMENT	NUMBER(38,0)
10	ORDSTATUS	CHAR(1 BYTE)



Trainer/Trainee Notes

This is a refresher of the typical trading setup – to help understand the data in the database. Remember the relationship between parent and child orders in an algo trading engine.

Returning the Same Table Twice

It may be necessary in your SQL commands to return the output of a table twice into your results.

MySQL provides a handy option to do this when you are choosing what columns you want to return.

```
select column1, column2, table.* from table;
```

This will return column1, column2 and then all the columns from the table you have selected. Effectively, you will get columns 1 and 2 twice in the result, with the full table displayed as column3 onwards.

Trainer/Trainee Notes

Here we are simply introducing the concept of table.* and that it can be used in the select statement to return all the columns of the table.

Column Arithmetic

For columns that contain numeric data, it is possible to do arithmetic on those values and return the result:

Command	Comments
select column1, column2+column3 from table;	Here we will return two columns in the result, column1 and the sum of column2 and column3
select column1, column2-column3 from table;	Here we will return two columns in the result, column1 and the result of subtracting column3 from column2
select column1, column2*column3 from table;	Here we will return two columns in the result, column1 and the result of multiplying column2 and column3
select column1, column2/column3 from table;	Here we will return two columns in the result, column1 and column2 divided by column3

Trainer/Trainee Notes

This is useful if you want to do some calculations on mathematical values within your query. As usual you should rename the columns using aliases, so it makes sense to the person reading the result.

The Dual Table

The dual table is a special 1 row, 1 column table that can be used in queries where you do not need to include any data from tables in the results.

This allows you to effectively use the table to do maths!

SELECT 1234.6 FROM DUAL;	This will return the number 1234.6
SELECT 2,3,2*3 FROM DUAL;	This will return the values, 2,3,6

Be aware there are slight differences between MySQL and Oracle DB around syntax and how you can use the table.

Trainer/Trainee Notes

Try out some queries on dual and see what it returns for you. You can also run these against live tables as well, but that is not necessary when you do not require data from those tables

SQL Functions

There is a wealth of extremely useful functions built into MySQL.

Function	Comments
SELECT ABS(number);	Select the absolute value of the number provided
SELECT AVG(column) FROM table;	Return the average of the entire column
SELECT CEIL(number);	CEIL function will return the smallest integer value that is greater than or equal to the number
SELECT 10 DIV 2;	This will return the result of 10 divided by 2
SELECT FLOOR(number);	FLOOR function will return the largest integer value less than or equal to the number provided
SELECT GREATEST(1, 4, 6, 4, 12);	Select the greatest number in a list
SELECT LEAST(4, 3, 6, 1);	Select the least number in a list
SELECT MAX(column) FROM table;	Select the max number from a column
SELECT Min(column) FROM table;	Select the min number from a column
SELECT MOD(number1, number2);	Select the remainder of number1/number2

Trainer/Trainee Notes

Here are some functions that can be useful in calculating output values

SQL Functions (continued)

Function	Comments
SELECT RAND();	This method will generate a random number between 0 and 1
SELECT RAND() * (20-7) + 7;	This is an extension of the previous method to get you to return a number between two specific numbers – in this case a number between 7 and 20
SELECT ROUND(number, decimalPlaces);	The round function will return the number rounded to the number of decimal places provided as the second argument
SELECT SUM(column) FROM table;	This will return the sum of all the values in the column provided

Trainer/Trainee Notes

Here are some functions that can be useful in calculating output values