

The One-to-Many Relationship

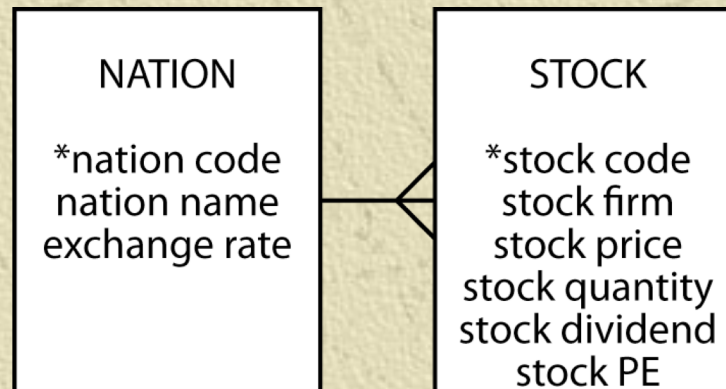
*Cow of many-well milked and
badly fed*

Spanish proverb

The one-to-many relationship

✦ Entities are related to other entities

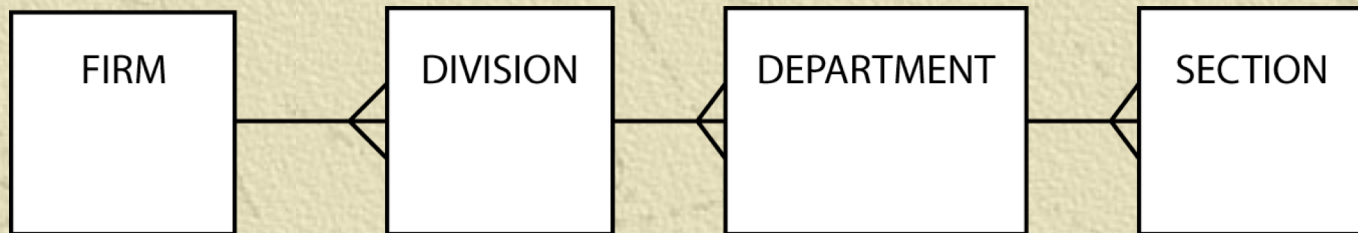
✦ A 1:m relationship



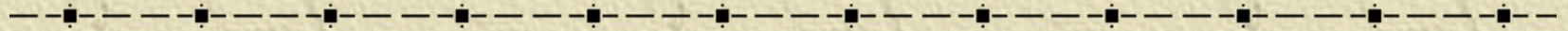
Hierarchical relationships

✦ Occur frequently

✦ Multiple 1:m relationships



STOCK with additional columns



stock							
stkcode	stkfirm	stkprice	stkqty	stkdiv	stkpe	natname	exchrate
FC	Freedonia Copper	27.50	10529	1.84	16	United Kingdom	1.00
PT	Patagonian Tea	55.25	12635	2.50	10	United Kingdom	1.00
AR	Abyssinian Ruby	31.82	22010	1.32	13	United Kingdom	1.00
SLG	Sri Lankan Gold	50.37	32868	2.68	16	United Kingdom	1.00
ILZ	Indian Lead &Zinc	37.75	6390	3.00	12	United Kingdom	1.00
BE	Burmese Elephant	.07	154713	0.01	3	United Kingdom	1.00
BS	Bolivian Sheep	12.75	231678	1.78	11	United Kingdom	1.00
NG	Nigerian Geese	35.00	12323	1.68	10	United Kingdom	1.00
CS	Canadian Sugar	52.78	4716	2.50	15	United Kingdom	1.00
ROF	Royal Ostrich Farms	33.75	1234923	3.00	6	United Kingdom	1.00
MG	Minnesota Gold	53.87	816122	1.00	25	USA	0.67
GP	Georgia Peach	2.35	387333	.20	5	USA	0.67
NE	Naremben Emu	12.34	45619	1.00	8	Australia	0.46
QD	Queensland Diamond	6.73	89251	.50	7	Australia	0.46
IR	Indooroopilly Ruby	15.92	56147	.50	20	Australia	0.46
BD	Bombay Duck	25.55	167382	1.00	12	India	0.0228

Create another entity to avoid update anomalies

- ✦ Insert
- ✦ Delete
- ✦ Update

Mapping to a relational database

- ✦ Each entity becomes a table
- ✦ The entity name becomes the table name
- ✦ Each attribute becomes a column
- ✦ Add a column to the table at the many end of a 1:m relationship
- ✦ Put the identifier of the one end in the added column

NATION and STOCK

nation		
<u>natcode</u>	natname	exchrates
UK	United Kingdom	1.00
USA	United States	0.67
AUS	Australia	0.46
IND	India	0.0228

stock						
<u>stkcode</u>	stkgfirm	stkprice	stkqty	stkdiv	stkpe	natcode
FC	Freedonia Copper	27.50	10529	1.84	16	UK
PT	Patagonian Tea	55.25	12635	2.50	10	UK
AR	Abyssinian Ruby	31.82	22010	1.32	13	UK
SLG	Sri Lankan Gold	50.37	32868	2.68	16	UK
ILZ	Indian Lead & Zinc	37.75	6390	3.00	12	UK
BE	Burmese Elephant	.07	154713	0.01	3	UK
BS	Bolivian Sheep	12.75	231678	1.78	11	UK
NG	Nigerian Geese	35.00	12323	1.68	10	UK
CS	Canadian Sugar	52.78	4716	2.50	15	UK
ROF	Royal Ostrich Farms	33.75	1234923	3.00	6	UK
MG	Minnesota Gold	53.87	816122	1.00	25	USA
GP	Georgia Peach	2.35	387333	.20	5	USA
NE	Naremben Emu	12.34	45619	1.00	8	AUS
QD	Queensland Diamond	6.73	89251	.50	7	AUS
IR	Indooroopilly Ruby	15.92	56147	.50	20	AUS
BD	Bombay Duck	25.55	167382	1.00	12	IND

Foreign keys

✦ A foreign key is a column that is a primary key of another table

◆ `natcode` in `stock` is a foreign key because `natcode` is the primary key of `nation`

✦ Records a 1:m relationship

Referential integrity constraint

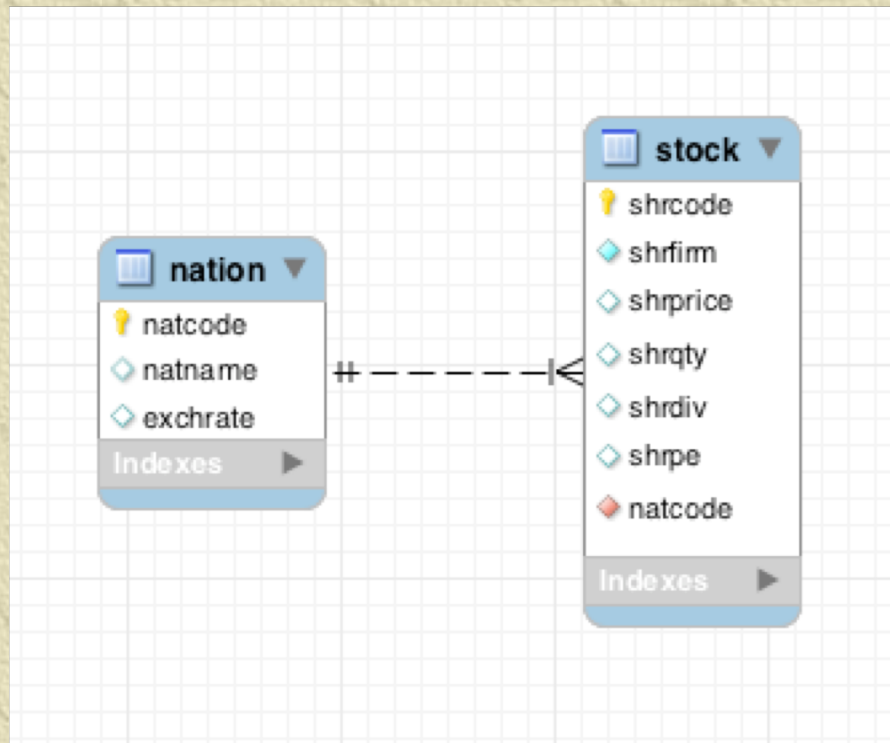
- ✦ For every value of a foreign key there is a primary key with that value
- ✦ For every value of `natcode` in `stock` there is a value of `natcode` in `nation`
- ✦ A primary key must exist before the foreign key can be defined
 - ◆ Must create the nation before its stocks

Creating the tables

```
CREATE TABLE nation (  
    natcode      CHAR(3),  
    natname      VARCHAR(20),  
    exchrates    DECIMAL(9,5),  
    PRIMARY KEY (natcode));
```

```
CREATE TABLE stock (  
    stkcode      CHAR(3),  
    stkfirm      VARCHAR(20),  
    stkprice     DECIMAL(6,2),  
    stkqty       DECIMAL(8),  
    stkdiv       DECIMAL(5,2),  
    stkpe        DECIMAL(5),  
    natcode      CHAR(3),  
    PRIMARY KEY (stkcode),  
    CONSTRAINT fk_has_nation FOREIGN KEY(natcode)  
        REFERENCES nation(natcode));
```

Representing a 1:m relationship in MySQL Workbench



A non-identifying relationship in MySQL Workbench

Exercise

-
- ✦ Develop a data model to keep track of a distance runner's times over various lengths
 - ✦ Create the database and add 3 rows for each of 2 athletes

Join



Create a new table from two existing tables by matching on a common column

```
SELECT * FROM stock, nation
WHERE stock.natcode = nation.natcode;
```

stkcode	stkfirm	stkprice	stkqty	stkdiv	stkpe	natcode	natcode	natname	exchrte
NE	Naremben Emu	12.34	45619	1.00	8	AUS	AUS	Australia	0.46000
IR	Indooroopilly Ruby	15.92	56147	0.50	20	AUS	AUS	Australia	0.46000
QD	Queensland Diamond	6.73	89251	0.50	7	AUS	AUS	Australia	0.46000
BD	Bombay Duck	25.55	167382	1.00	12	IND	IND	India	0.02280
ROF	Royal Ostrich Farms	33.75	1234923	3.00	6	UK	UK	United Kingdom	1.00000
CS	Canadian Sugar	52.78	4716	2.50	15	UK	UK	United Kingdom	1.00000
FC	Freedonia Copper	27.50	10529	1.84	16	UK	UK	United Kingdom	1.00000
BS	Bolivian Sheep	12.75	231678	1.78	11	UK	UK	United Kingdom	1.00000
BE	Burmese Elephant	0.07	154713	0.01	3	UK	UK	United Kingdom	1.00000
ILZ	Indian Lead & Zinc	37.75	6390	3.00	12	UK	UK	United Kingdom	1.00000
SLG	Sri Lankan Gold	50.37	32868	2.68	16	UK	UK	United Kingdom	1.00000
AR	Abyssinian Ruby	31.82	22010	1.32	13	UK	UK	United Kingdom	1.00000
PT	Patagonian Tea	55.25	12635	2.50	10	UK	UK	United Kingdom	1.00000
NG	Nigerian Geese	35.00	12323	1.68	10	UK	UK	United Kingdom	1.00000
MG	Minnesota Gold	53.87	816122	1.00	25	US	US	United States	0.67000
GP	Georgia Peach	2.35	387333	0.20	5	US	US	United States	0.67000

Join

Report the value of each stock holding in UK pounds. Sort the report by nation and firm.

```
SELECT natname, stkfirm, stkprice, stkqty, exchrates,
       stkprice*stkqty*exchrates AS stkvalue
FROM stock,nation
WHERE stock.natcode = nation.natcode
ORDER BY natname, stkfirm;
```

natname	stkfirm	stkprice	stkqty	exchrates	stkvalue
Australia	Indooroopilly Ruby	15.92	56147	0.46000	411175.71
Australia	Narembeen Emu	12.34	45619	0.46000	258951.69
Australia	Queensland Diamond	6.73	89251	0.46000	276303.25
India	Bombay Duck	25.55	167382	0.02280	97506.71
United Kingdom	Abyssinian Ruby	31.82	22010	1.00000	700358.20
United Kingdom	Bolivian Sheep	12.75	231678	1.00000	2953894.50
United Kingdom	Burmese Elephant	0.07	154713	1.00000	10829.91
United Kingdom	Canadian Sugar	52.78	4716	1.00000	248910.48
United Kingdom	Freedonia Copper	27.50	10529	1.00000	289547.50
United Kingdom	Indian Lead & Zinc	37.75	6390	1.00000	241222.50
United Kingdom	Nigerian Geese	35.00	12323	1.00000	431305.00
United Kingdom	Patagonian Tea	55.25	12635	1.00000	698083.75
United Kingdom	Royal Ostrich Farms	33.75	1234923	1.00000	41678651.25
United Kingdom	Sri Lankan Gold	50.37	32868	1.00000	1655561.16
United States	Georgia Peach	2.35	387333	0.67000	609855.81
United States	Minnesota Gold	53.87	816122	0.67000	29456209.73

GROUP BY - reporting by groups

Report by nation the total value of stockholdings.

```
SELECT natname, SUM(stkprice*stkqty*exchrates) AS stkvalue
FROM stock, nation WHERE stock.natcode = nation.natcode
GROUP BY natname;
```

natname	stkvalue
Australia	946430.65
India	97506.71
United Kingdom	48908364.25
United States	30066065.54

HAVING - the WHERE clause of groups

Report the total value of stocks for nations with two or more listed stocks.

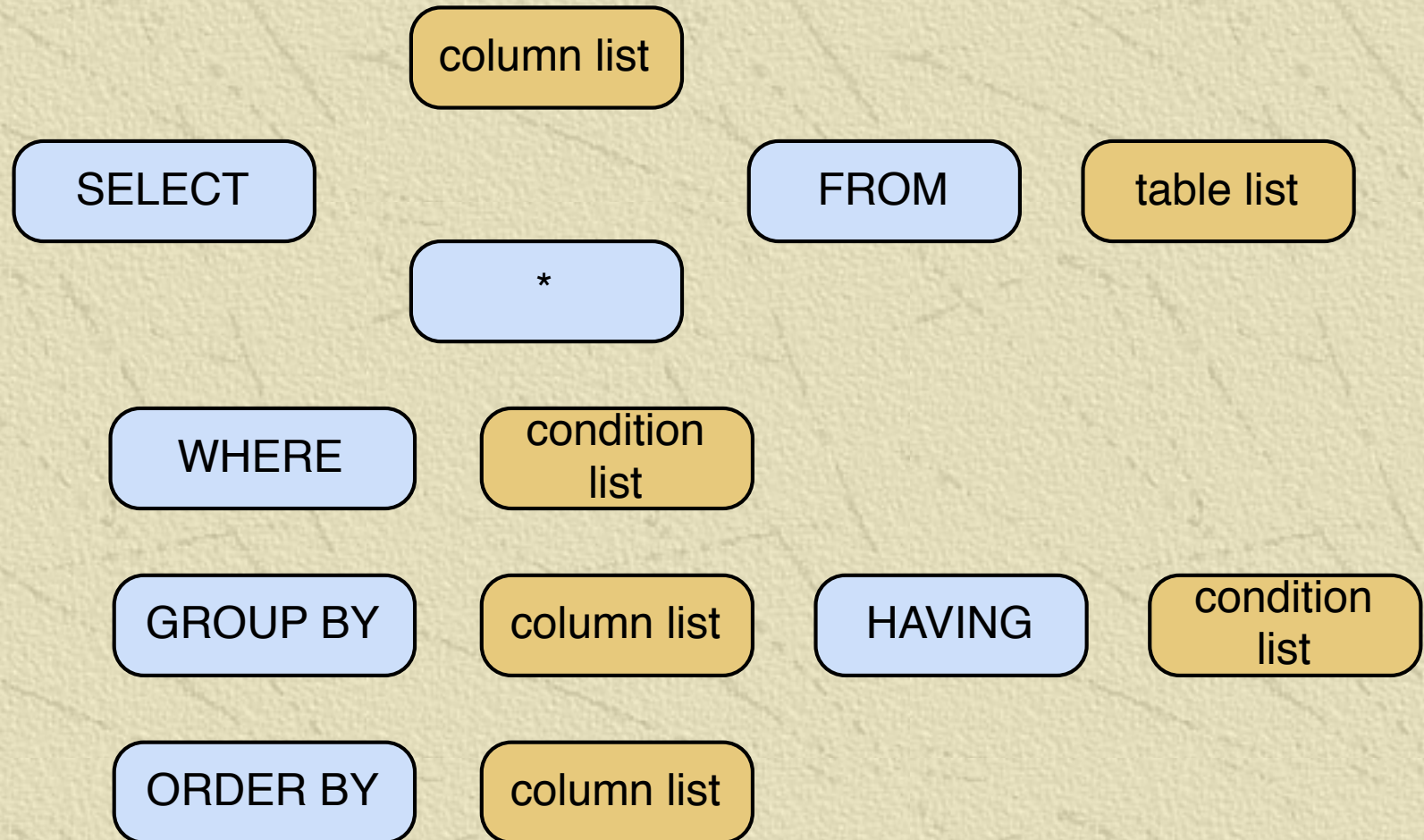
```
SELECT natname, SUM(stkprice*stkqty*exchrates) AS stkvalue
FROM stock, nation WHERE stock.natcode = nation.natcode
GROUP BY natname
HAVING COUNT (*) >= 2;
```

natname	stkvalue
Australia	946430.65
United Kingdom	48908364.25
United States	30066065.54

Exercise

-
- ✦ Report the total dividend payment for each country that has three or more stocks in the portfolio

Structure of SQL statements



Regular expression

✦ Search for a string not containing specified characters

✦ [^a-f] means any character not in the set containing a, b, c, d, e, or f

✦ *List the names of nations with non-alphabetic characters in their names*

```
SELECT * FROM nation
WHERE natname REGEXP '[^a-z|A-Z]'
```

Regular expression

- ✦ Search for a string containing a repetition
- ✦ {n} means repeat the pattern n times
- ✦ *List the names of firms with a double 'e'.*

```
SELECT * FROM stock
WHERE stkfirm REGEXP '[e]{2}'
```

Regular expression

- ✦ Search for a string containing several different specified strings
- ✦ | means alternation (or)
- ✦ *List the names of firms with a double 's' or a double 'n'.*

```
SELECT * FROM stock
WHERE stkfirm REGEXP '[s]{2}|[n]{2}'
```

Regular expression

- ✦ Search for multiple versions of a string
- ✦ [ea] means any character from the set containing e and a
 - ✦ It will match for 'e' or 'a'
- ✦ *List the names of firms with names that include 'inia' or 'onia'.*

```
SELECT * FROM stock  
WHERE stkfirm REGEXP '[io]nia'
```

Regular expression

Find firms with 't' as the third letter of their name.

```
SELECT shrfirm FROM share
WHERE shrfirm REGEXP '^(.){2}t';
```

Find firms not containing an 's' in their name.

```
SELECT shrfirm FROM share
WHERE shrfirm NOT REGEXP 's|S';
```

regexlib.com

-
- ✦ A library of regular expressions
 - ✦ Cheat sheet for creating expressions
 - ✦ Regex Tester

Exercise

✦ Report the names of nations starting with 'United'

Subqueries



A query nested within another query

Report the names of all Australian stocks.

```
SELECT stkfirm FROM stock
WHERE natcode IN
  (SELECT natcode FROM nation
   WHERE natname = 'Australia');
```

stkfirm
Narembeen Emu
Queensland Diamond
Indooroopilly Ruby

Correlated subquery

✦ Solves the inner query many times

Find those stocks where the quantity is greater than the average for that country.

```
SELECT natname, stkfirm, stkqty FROM stock, nation
WHERE stock.natcode = nation.natcode
AND stkqty >
    (SELECT AVG(stkqty) FROM stock
     WHERE stock.natcode = nation.natcode);
```

natname	stkfirm	stkqty
Australia	Queensland Diamond	89251
United Kingdom	Bolivian Sheep	231678
United Kingdom	Royal Ostrich Farms	1234923
United States	Minnesota Gold	816122

*Correlated
subqueries can be
resource intensive*

Correlated subquery

nation		
natcode	natname	exchrates
UK	United Kingdom	1.00
USA	United States	0.67
AUS	Australia	0.46
IND	India	0.0228

```
SELECT natname, stkfirm, stkqty FROM
stock, nation
WHERE stock.natcode = nation.natcode
AND stkqty >
(SELECT AVG(stkqty) FROM stock
WHERE stock.natcode = nation.natcode);
```

stock						
stkcode	stkfirm	stkprice	stkqty	stkdiv	stkpe	natcode
FC	Freedonia Copper	27.50	10529	1.84	16	UK
PT	Patagonian Tea	55.25	12635	2.50	10	UK
AR	Abyssinian Ruby	31.82	22010	1.32	13	UK
SLG	Sri Lankan Gold	50.37	32868	2.68	16	UK
ILZ	Indian Lead &Zinc	37.75	6390	3.00	12	UK
BE	Burmese Elephant	.07	154713	0.01	3	UK
BS	Bolivian Sheep	12.75	231678	1.78	11	UK
NG	Nigerian Geese	35.00	12323	1.68	10	UK
CS	Canadian Sugar	52.78	4716	2.50	15	UK
ROF	Royal Ostrich Farms	33.75	1234923	3.00	6	UK
MG	Minnesota Gold	53.87	816122	1.00	25	USA
GP	Georgia Peach	2.35	387333	.20	5	USA
NE	Naremben Emu	12.34	45619	1.00	8	AUS
QD	Queensland Diamond	6.73	89251	.50	7	AUS
IR	Indooroopilly Ruby	15.92	56147	.50	20	AUS
BD	Bombay Duck	25.55	167382	1.00	12	IND

Correlated subquery

nation	natcode	natname	exchrates
UK	United Kingdom	1.00	
USA	United States	0.67	
AUS	Australia	0.46	
IND	India	0.0228	

```
SELECT natname, stkfirm, stkqty FROM
stock, nation
WHERE stock.natcode = nation.natcode
AND stkqty > 172275.5;
```

stock	stkcode	stkfirm	stkprice	stkqty	stkdiv	stkpe	natcode
FC	Freedonia Copper	27.50	10529	1.84	16	UK	
PT	Patagonian Tea	55.25	12635	2.50	10	UK	
AR	Abyssinian Ruby	31.82	22010	1.32	13	UK	
SLG	Sri Lankan Gold	50.37	32868	2.68	16	UK	
ILZ	Indian Lead &Zinc	37.75	6390	3.00	12	UK	
BE	Burmese Elephant	.07	154713	0.01	3	UK	
BS	Bolivian Sheep	12.75	231678	1.78	11	UK	
NG	Nigerian Geese	35.00	12323	1.68	10	UK	
CS	Canadian Sugar	52.78	4716	2.50	15	UK	
ROF	Royal Ostrich Farms	33.75	1234923	3.00	6	UK	
MG	Minnesota Gold	53.87	816122	1.00	25	USA	
GP	Georgia Peach	2.35	387333	.20	5	USA	
NE	Naremben Emu	12.34	45619	1.00	8	AUS	
QD	Queensland Diamond	6.73	89251	.50	7	AUS	
IR	Indooroopilly Ruby	15.92	56147	.50	20	AUS	
BD	Bombay Duck	25.55	167382	1.00	12	IND	

Exercise

-
- ✦ Report the country, firm, and stock holding for the maximum quantity of stock held for each country

Views - virtual tables

- ✦ An imaginary table constructed by the DBMS when required
- ✦ Only the definition of the view is stored, not the result

```
CREATE VIEW stkvalue
(nation, firm, price, qty, exchrates, value)
AS SELECT natname, stkfirm, stkprice, stkqty, exchrates,
       stkprice*stkqty*exchrates
       FROM stock, nation
       WHERE stock.natcode = nation.natcode;
```

Views - querying

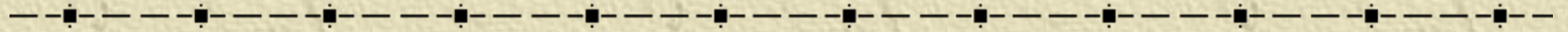


Query exactly as if a table

```
SELECT nation, firm, value  
FROM stkvalue WHERE value > 100000;
```


nation	firm	value
United Kingdom	Freedonia Copper	289547.50
United Kingdom	Patagonian Tea	698083.75
United Kingdom	Abyssinian Ruby	700358.20
United Kingdom	Sri Lankan Gold	1655561.16
United Kingdom	Indian Lead & Zinc	241222.50
United Kingdom	Bolivian Sheep	2953894.50
United Kingdom	Nigerian Geese	431305.00
United Kingdom	Canadian Sugar	248910.48
United Kingdom	Royal Ostrich Farms	41678651.25
United States	Minnesota Gold	29456209.73
United States	Georgia Peach	609855.80
Australia	Narembeen Emu	258951.69
Australia	Queensland Diamond	276303.24
Australia	Indooroopilly Ruby	411175.71

Why create a view?



- ✦ Simplify query writing
 - ◆ Calculated columns
- ✦ Restrict access to parts of a table

Exercise

 Create a view for dividend payment

Summary

New topics

- ◆ 1:m relationship
- ◆ Foreign key
- ◆ Correlated subquery
- ◆ GROUP BY
- ◆ HAVING clause
- ◆ View