/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* \*\*

\*\* CHAPTER 3. Basic Data Manipulation Practical 1. \*\*

\*\* Limiting columns, calculations, aliases, sorting and duplicates \*\*

\*\* \*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--p1) Write a SELECT statement to display the entire content of the company table.

-- (Hint: you can use \* here to return complete rows)

-- When you run the statement, you should see results like this:

Use database0

SELECT \* FROM company;

company\_no name tel county post\_code

----------- -------------------- --------------- --------------- ----------

1000 Happy Heaters PLC (01306)345672 London SE3 89L

2000 Icicle Igloos Inc 0207-987-1265 London N1 4LH

3000 Judo Jeans PLC 0207-478-2990 London N9 2FG

4000 Kipper Kickers Inc 01254-987766 Devon PL4 9RT

--p2) Write a SELECT statement to return the department number (dept\_no), manager

-- and department\_name (dept\_name) for all departments, i.e. the dept table.

-- Expected results:

SELECT dept\_no, manager, dept\_name FROM dept;

dept\_no manager dept\_name

----------- -------------------- --------------------

1 Adam Apricot Animal Products

2 Barbara Banana Business Systems

3 Paul Peach Credit Control

4 Diver Dan Desktop Systems

5 Xavier Xylophone Electrical Repairs

--p3) Write a SELECT statement to return the order number (order\_no) and value (order\_value) of each sale.

SELECT order\_no, order\_value from sale;

-- Expected results:

order\_no order\_value

----------- -----------

100 7

200 6

300 12

400 5

500 2

600 27

700 3

800 3

--p4) Assuming there's a ten percent shipping charge, modify the query to calculate the overall value of each sale.

-- (Hint: Use a column alias.)

SELECT order\_no, order\_value, order\_value + order\_value \* 0.1 as value\_with\_shipping from sale;

-- Expected results:

order\_no value\_with\_shipping

----------- ---------------------------------------

100 7.7

200 6.6

300 13.2

400 5.5

500 2.2

600 29.7

700 3.3

800 3.3

--p5) We would like you to list all the salespeople, but...

-- Some INFORMATION for you - the sales\_targets of the salespeople are ANNUAL targets.

-- We would like you to display 2 columns only.

-- These are the surnames (lname) and the QUARTERLY target of each salesperson.

-- We would also like the rows to be sorted by these quarterly targets,

-- but with the largest value appearing first.

-- (Hint: It's another calculation.)

select lname, sales\_target/4 as quaterly\_target from salesperson order by sales\_target/4 desc

-- Expected results:

lname Quarterly Target

--------------- ------------------

Custard 3.500000

Goalie 3.250000

Flipper 3.000000

Ernst 2.750000

Brick 2.250000

Digger 1.750000

--p6) Display 3 columns, each salesperson's employee number, dept number and full name, displayed as 'fullname'.

-- (Hint: the salesperson's name is stored in the fname and lname columns and can be concatenated.)

select emp\_no, dept\_no, concat(fname,' ',lname) as fullname from salesperson

emp\_no dept\_no fullname

----------- ----------- -------------------------------

10 1 Alan Brick

20 2 Billy Custard

30 2 Chris Digger

40 3 Dick Ernst

50 3 Ernest Flipper

60 3 Fred Goalie

--p7) Modify the query you've just written to display the salespserson's name as initials.

-- They should be displayed as one value separated by a fullstop, eg. Alan Brick should be ‘A.B’.

-- The answer set should have a column heading saying 'Initials'.

-- REMINDER, the function 'SUBSTRING' has EXACTLY 3 MANDATORY (comma-separated) parameters.

select emp\_no, dept\_no, concat(SUBSTRING(fname,1,1),'.',SUBSTRING(lname,1,1)) as fullname

from salesperson

emp\_no dept\_no Initials

----------- ----------- --------

10 1 A.B

20 2 B.C

30 2 C.D

40 3 D.E

50 3 E.F

60 3 F.G

--p8) List the employee numbers of those salespeople who have sold.

-- Remove duplicates!! - (3 row result set)

SELECT DISTINCT emp\_no FROM sale

emp\_no

-----------

10

50

60

-- IF YOU HAVE TIME

--p9) From the salesperson table list the dept\_no/county combinations. Again,

-- no duplicate entries please. - (4 rows)

select DISTINCT dept\_no, county from salesperson

dept\_no county

----------- ---------------

1 Surrey

2 Hampshire

3 London

3 Surrey

--p10) List (sorted) just the month numbers of the 8 sales.

-- Use the DatePart function

SELECT MONTH(order\_date) month FROM sale order by MONTH(order\_date)

month

-----------

1

5

5

6

7

7

8

11

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* \*\*

\*\* END OF CHAPTER 3. Practical 1 \*\*

\*\* \*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* \*\*

\*\* CHAPTER 3. Basic Data Manipulation Practical 2. \*\*

\*\* Using a 'WHERE' clause to restrict rows \*\*

\*\* \*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- Just in case you need them:

SELECT \* FROM dept

SELECT \* FROM sale

SELECT \* FROM contact

SELECT \* FROM company

--p1) Display the names of companies based in London.

-- (Hint: Check the county column).

name

--------------------

Happy Heaters PLC

Icicle Igloos Inc

Judo Jeans PLC

--p2) Display a list of sales which have a value greater than 6.

-- (SELECT \* will be fine here.)

order\_no emp\_no company\_no contact\_code order\_value order\_date description

----------- ----------- ----------- ------------ ----------- ----------------------- --------------------------------------------------

100 60 1000 MM 7 2006-06-24 00:00:00.000 Toshiba 6700 Pro

300 60 2000 OO 12 2006-07-14 00:00:00.000 ScanPRO 4800 Scanner

600 50 3000 PP 27 2006-05-23 00:00:00.000 Complete Desktop Publishing System

--p3) Display a list of sales made to company 3000 by the salesperson with an employee number of 60.

-- For each sale, display the value and description.

order\_value description

----------- --------------------------------------------------

6 MS Office Professional \* 30

3 Printer cartridges

--p4) Display only those sales whose order value is in the range 10 to 30 inclusive

-- (2 row result set). Note which employee(s) made these 2 sales.

-- (SELECT \* will be fine here too.)

order\_no emp\_no company\_no contact\_code order\_value order\_date description

----------- ----------- ----------- ------------ ----------- ------------------------------------------------------ --------------------------------------------------

300 60 2000 OO 12 2006-07-14 00:00:00.000 ScanPRO 4800 Scanner

600 50 3000 PP 27 2006-05-23 00:00:00.000 Complete Desktop Publishing System

--p5) Now modify the statement you have just written:

-- Copy/Paste it underneath here and amend the WHERE clause so that we further

-- restrict these sales to just those made by either employee 10 or employee 50

-- (1 row result set).

order\_no emp\_no company\_no contact\_code order\_value order\_date description

----------- ----------- ----------- ------------ ----------- ------------------------------------------------------ --------------------------------------------------

600 50 3000 PP 27 2006-05-23 00:00:00.000 Complete Desktop Publishing System

-- Did you just use the word 'OR' in the last bit of code?

-- If you did, and used no parentheses i.e. '(' and ')', then you asked the wrong question

-- although you might appear to have the correct answer!!

-- Hopefully you used the 'IN' word which avoided using 'OR'. If you didn’t, ensure you now do.

-- This SELECT statement is INCORRECT, as it would show all sales by emp\_no 50 regardless of value:

SELECT \*

FROM sale

WHERE order\_value BETWEEN 10 AND 30

AND emp\_no = 10 OR emp\_no = 50

--p6) Produce firstly a list of departments whose name contains 'SYS'.

-- Get this working first!! (depts 2 & 4 should appear).

dept\_no dept\_name manager sales\_target

----------- -------------------- -------------------- ---------------------------------------

2 Business Systems Barbara Banana 15.00

4 Desktop Systems Diver Dan 5.00

--p7) Now amend the query to ALSO include any depts whose sales\_target is less than 10.

-- Get that working now (depts 1, 2 & 4 should appear).

dept\_no dept\_name manager sales\_target

----------- -------------------- -------------------- ---------------------------------------

1 Animal Products Adam Apricot 5.00

2 Business Systems Barbara Banana 15.00

4 Desktop Systems Diver Dan 5.00

--p8) But we DO NOT want to see under ANY circumstances

-- a row in the answer set that has 'Barbara Banana' as manager.

-- So, amend the query further to reflect this requirement (depts 1 & 4 ONLY should now appear).

-- If you have the wrong answer, OR if you have used no parentheses, then revisit your code!!

dept\_no dept\_name manager sales\_target

----------- -------------------- -------------------- --------------

1 Animal Products Adam Apricot 5.00

4 Desktop Systems Diver Dan 5.00

-- ONLY IF YOU HAVE TIME

--p9) Display contact names and tel numbers in Inner London.

-- Inner London is '0207'.

-- Note: tel numbers will never be stored as numeric columns because you don't

-- ever want to do arithmetic on them!

-- (Correct answer is 3 rows, not 2 and not 4)

-- You need to work around some very 'iffy'/'dodgy'/'bad' data.

name tel

-------------------- -------------------------

Munching Mike (0207)223-9887

Ollie Octopus 0207-341-566670 ext 10

Ricky Rambo 0207-988-0777

-- If you are not happy with your answer, then don't worry, we will REVIEW after practical!

--p10) List sales made in May and July (any year).

-- Use the MS SQL Server date function called 'DATEPART'

-- (4 rows)

order\_no order\_date

----------- ------------------------------------------------------

200 2006-05-01 00:00:00.000

300 2006-07-14 00:00:00.000

500 2006-07-23 00:00:00.000

600 2006-05-23 00:00:00.000

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* \*\*

\*\* END OF CHAPTER 3. Practical 2 \*\*

\*\* \*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* \*\*

\*\* CHAPTER 3. Basic Data Manipulation Practical 3. \*\*

\*\* Nulls, Nullability, and 3 way logic \*\*

\*\* \*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- Just in case you need it:

SELECT \* FROM salesperson

--p1) Produce a list of people who have a real (non null) post code value

-- (4 row result set).

emp\_no post\_code

----------- ----------

10 RT8 8LP

20 RF3 9UD

30 W45 TY3

50 CR1 2GH

--p2) Now display emp number and post code of the sales people, but if the post code is 'unknown'

-- then the string 'Post Code Unknown' should be displayed instead.

-- Make use of the function 'COALESCE', ensuring the 2nd column (calculated) has a col heading.

-- (Hint: There's no WHERE clause for this one.)

emp\_no Postcode

----------- -----------------

10 RT8 8LP

20 RF3 9UD

30 W45 TY3

40 Post Code Unknown

50 CR1 2GH

60 Post Code Unknown

--p3) Run only the SELECT/FROM of this precoded query and then estimate how many rows will be

-- returned if you then run it with the WHERE clause added.

-- Now uncomment the 'WHERE' and run the query again. Did you estimate correctly?

SELECT emp\_no, post\_code

FROM salesperson

--WHERE post\_code <> 'RT8 8LP'

-- Can you see how you could get 5 rows to appear?

SELECT emp\_no, post\_code

FROM salesperson

--WHERE post\_code <> 'RT8 8LP' OR ????????????????

emp\_no post\_code

----------- ----------

20 RF3 9UD

30 W45 TY3

40 NULL

50 CR1 2GH

60 NULL

-- IF YOU HAVE TIME

-- Consider these 2 code samples, feel free to run them

CREATE VIEW PostCodeList

AS

SELECT emp\_no, COALESCE(post\_code, 'Post Code Unknown') AS Postcode

FROM salesperson

The command(s) completed successfully.

SELECT \*

FROM PostCodeList -- treating the view as if it was a table

emp\_no Postcode

----------- -----------------

10 RT8 8LP

20 RF3 9UD

30 W45 TY3

40 Post Code Unknown

50 CR1 2GH

60 Post Code Unknown

-- Can you see now how COALESCE can become useful?

-- We do a chapter on VIEWs later, but this should whet your appetite.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* \*\*

\*\* END OF CHAPTER 3. Practical 3. \*\*

\*\* \*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/