Sam Barrionuevo ITI-2180 Foundations of Data and Database Management

**Problem A**

Input:

SELECT

title,

retail

FROM

books

WHERE

retail < (

SELECT

AVG(retail)

FROM

books

);

Result:

TITLE RETAIL

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

BODYBUILD IN 10 MINUTES A DAY 30.95

REVENGE OF MICKEY 22

COOKING WITH MUSHROOMS 19.95

HANDCRANKED COMPUTERS 25

THE WOK WAY TO COOK 28.75

BIG BEAR AND LITTLE DOVE 8.95

HOW TO GET FASTER PIZZA 29.95

HOW TO MANAGE THE MANAGER 31.95

SHORTEST POEMS 39.95

9 rows selected.

**Problem B**

Input:

SELECT

b.title,

b.cost,

c.category

FROM

books b,

(

SELECT

category,

AVG(cost) ca

FROM

books

GROUP BY

category

) c

WHERE

b.category = c.category

AND b.cost < c.ca;

Result:

TITLE COST CATEGORY

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

REVENGE OF MICKEY 14.2 FAMILY LIFE

DATABASE IMPLEMENTATION 31.4 COMPUTER

COOKING WITH MUSHROOMS 12.5 COOKING

HANDCRANKED COMPUTERS 21.8 COMPUTER

BIG BEAR AND LITTLE DOVE 5.32 CHILDREN

**Problem C**

Input:

SELECT

order#

FROM

orders

WHERE

shipstate = (

SELECT

shipstate

FROM

orders

WHERE

order# = 1014

);

Result:

ORDER#

----------

1007

1014

**Problem D**

Input:

SELECT

order#,

SUM(quantity \* paideach) as "Total"

FROM

orders

JOIN orderitems USING ( order# )

GROUP BY

order#

HAVING

SUM(quantity \* paideach) > (

SELECT

SUM(quantity \* paideach)

FROM

orderitems

WHERE

order# = 1008

)

ORDER BY

order#;

Result:

ORDER# Total

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

1001 117.4

1002 111.9

1003 106.85

1004 170.9

1005 39.95

1006 54.5

1007 335.85

1009 41.95

1010 55.95

1011 85.45

1012 166.4

ORDER# Total

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

1013 55.95

1014 44

1016 85.45

1018 75.9

15 rows selected.

**Problem E**

Input:

SELECT UNIQUE

lname,

fname

FROM

author

JOIN bookauthor USING ( authorid )

JOIN books USING ( isbn )

JOIN orderitems USING ( isbn )

WHERE

isbn IN (

SELECT

isbn

FROM

author

JOIN bookauthor USING ( authorid )

JOIN books USING ( isbn )

JOIN orderitems USING ( isbn )

GROUP BY

isbn

HAVING

COUNT(isbn) = (

SELECT

MAX(COUNT(\*))

FROM

orderitems

GROUP BY

isbn

)

);

Result:

LNAME FNAME

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

BAKER JACK

**Problem F**

Input:

SELECT

title,

retail

FROM

books

JOIN publisher USING ( pubid )

WHERE

retail = (

SELECT

MAX(retail)

FROM

books

WHERE

pubid = 3

)

Result:

TITLE RETAIL

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

HOLY GRAIL OF ORACLE 75.95

**Problem G**

Input:

CREATE VIEW reorderinfo AS

SELECT

isbn,

title,

contact,

phone

FROM

books

JOIN publisher USING ( pubid )

Result:

View REORDERINFO created.

**Problem H**

Input:

UPDATE reorderinfo

SET

isbn = 11111111

WHERE

title LIKE 'REVENGE%';

Result:

Error starting at line : 1 in command -

update reorderinfo

set isbn=11111111 where title like 'REVENGE%'

Error report -

ORA-02292: integrity constraint (SEBARRIONUEVO.ORDERITEMS\_ISBN\_FK) violated - child record found

Response: Yes there was an error message, it was a foreign key on the orderitems table because the original isbn record was from the books table. It was being referenced so it could not be changed.

**Problem I**

Input:

CREATE VIEW full\_orders AS

SELECT

\*

FROM

customers

JOIN orders USING ( customer# )

JOIN orderitems USING ( order# )

WITH READ ONLY;

Result:

View FULL\_ORDERS created.

**Problem J**

Input:

SELECT

firstname,

lastname,

order#,

nvl(shipcost, 0) AS shipping,

tax.rate AS tax,

round((SUM(quantity \* paideach) + nvl(shipcost, 0)) +(SUM(quantity \* paideach) + nvl(shipcost, 0)) \*(rate / 100),

2) AS total

FROM

full\_orders

JOIN tax ON shipstate = tax.state

GROUP BY

firstname,

lastname,

order#,

shipcost,

rate

ORDER BY

total DESC;

Result:

FIRSTNAME LASTNAME ORDER# SHIPPING TAX TOTAL

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

TAMMY GIANA 1007 7 6.25 364.28

KENNETH FALAH 1004 0 7 182.86

BECCA NELSON 1012 6 6 182.74

JAKE LUCAS 1001 3 4 125.22

REESE MCGOVERN 1002 3 6.25 122.08

BONITA MORALES 1003 4 6 117.5

LEILA SMITH 1016 2 6 92.7

JAKE LUCAS 1011 2 4 90.95

BONITA MORALES 1018 0 6 80.45

JENNIFER SMITH 1010 2 7 62.01

JASMINE LEE 1013 2 4 60.27

FIRSTNAME LASTNAME ORDER# SHIPPING TAX TOTAL

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

LEILA SMITH 1006 2 6 59.89

TAMMY GIANA 1014 3 6.25 49.94

THOMAS PIERSON 1008 3 6 45.47

CINDY GIRARD 1009 0 6.5 44.68

GREG MONTIASA 1005 2 4 43.63

GREG MONTIASA 1019 2 4 24.96

KENNETH FALAH 1015 2 7 23.49

CINDY GIRARD 1000 2 6.5 23.38

KENNETH JONES 1020 2 4 22.83

STEVE SCHELL 1017 3 6 22.15

21 rows selected.