**IST370 Practice Worksheet (under construction)**

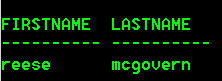
**Theme: Single-Row Functions**

**Write corresponding queries to produce the query results shown as below:**

**CHARACTER STRING FUNCTIONS**

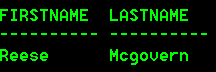
SELECT \_\_\_\_(firstname) firstname, \_\_\_\_(lastname) lastname FROM customers

WHERE state = 'IL';



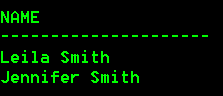
SELECT \_\_\_\_\_(firstname) firstname, \_\_\_\_\_(lastname) lastname FROM customers

WHERE state = 'IL';

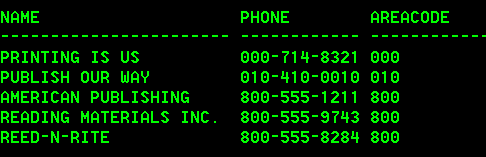


SELECT \_\_\_\_\_(firstname) \_\_\_\_\_ \_\_\_\_(lastname) name FROM customers

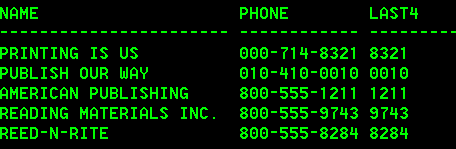
WHERE \_\_\_\_\_(lastname) = 'smith';



SELECT name, phone, \_\_\_\_\_ areacode FROM publishers;



SELECT name, phone, \_\_\_\_\_ last4 FROM publishers;

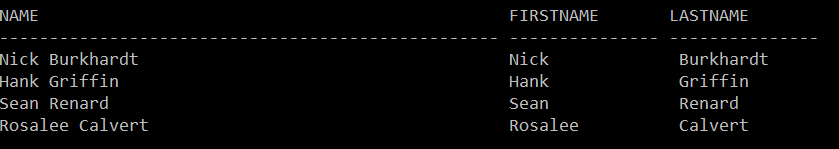


SELECT name,

\_\_\_\_(name, 1, \_\_\_\_ (name, ' ') - 1) firstname,

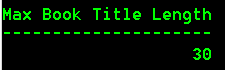
\_\_\_\_(name, \_\_\_\_\_(name, ' ') + 1) lastname

FROM instructors;



SELECT **\_\_\_\_(\_\_\_\_**(title)) "Max Book Title Length"

FROM books;



**Concatenating Character Strings**

SELECT firstname, lastname, 'Customer Number: ' || customer# "Customer Number"

FROM customers;

or

SELECT firstname, lastname, CONCAT('Customer Number: ' CONCAT customer#) "Customer Number"

FROM customers;



**Does the following query work? If not, why?**

SELECT **CONCAT(firstname,|| ' '||, lastname)** "Customer Name", **CONCAT ('Customer Number: ', customer#)** "Customer Number"

FROM customers;

**Question: How many arguments can CONCAT take?**

**Then how do you modify the code to make it work?**

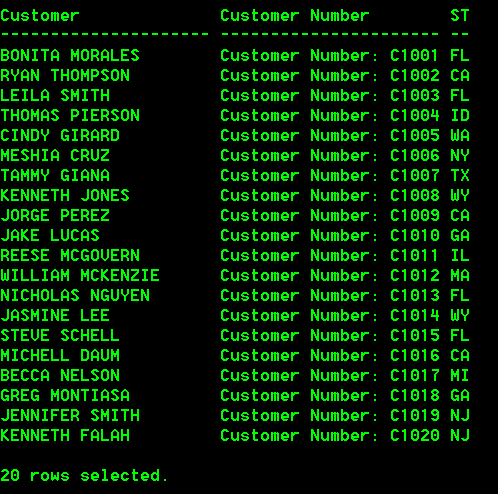
SELECT firstname ||’’|| lastname "Customer", CONCAT ('Customer Number: ', customer#) "Customer Number", state

FROM customers;

or

SELECT CONCAT(firstname, CONCAT(' ', lastname)) "Customer", CONCAT ('Customer Number: ', customer#) "Customer Number", state

FROM customers;



**NUMBER FUNCTIONS**

**Run the following queries for rounding the number and see what result you obtain:**

SELECT

ROUND(15.7945,3) nearest\_thousandth,

ROUND(15.7945,2) nearest\_hundredth,

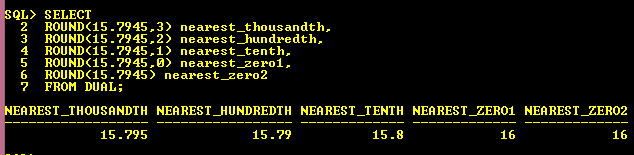
ROUND(15.7945,1) nearest\_tenth,

ROUND(15.7945,0) nearest\_zero1,

ROUND(15.7945) nearest\_zero2

FROM DUAL;

Result:



**Run the following queries for truncating the number and see what result you obtain:**

SELECT

TRUNC(15.7945,3) nearest\_thousandth,

TRUNC(15.7945,2) nearest\_hundredth,

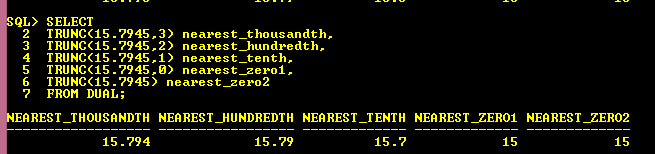
TRUNC(15.7945,1) nearest\_tenth,

TRUNC(15.7945,0) nearest\_zero1,

TRUNC(15.7945) nearest\_zero2

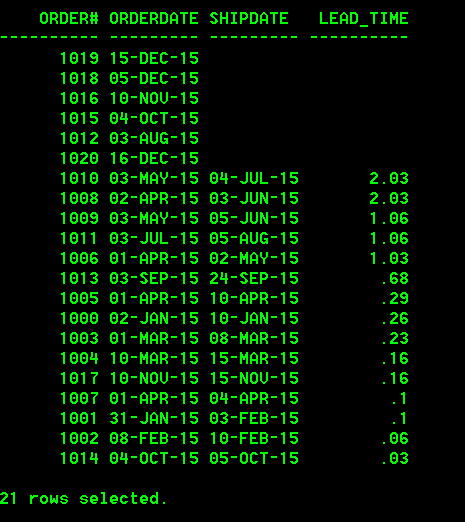
FROM DUAL;

Result:



**DATE/TIME FUNCTIONS**

**What is the lead time (in months) between the order date and ship date of each order placed in the past?**



SELECT order#, orderdate, shipdate,

ROUND(MONTHS\_Between(shipdate,orderdate)**,** 2) lead\_time

FROM orders

ORDER BY lead\_time DESC;

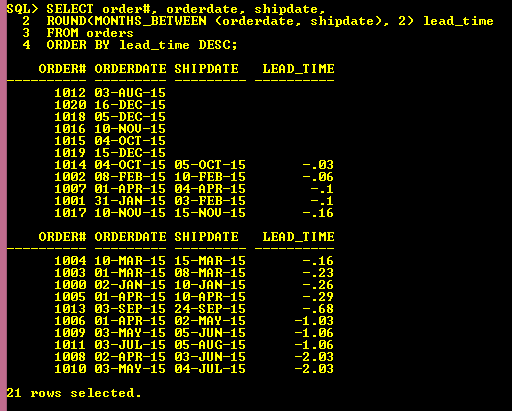
**Question: What do you get from running the following query?**

SELECT order#, orderdate, shipdate,

ROUND(MONTHS\_BETWEEN (orderdate, shipdate), 2) lead\_time

FROM orders

ORDER BY lead\_time DESC;



**What is the longest lead time (in months) of delivery?**

SELECT MAX(MONTHS\_BETWEEN(shipdate,orderdate) max\_lead\_time

FROM orders;

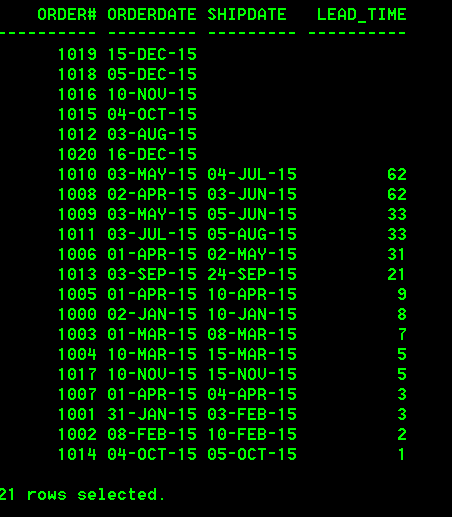
**What is the lead time (in DAYS) between the order date and ship date of each order placed in the past?**

SELECT order#, orderdate, shipdate,

lead\_time

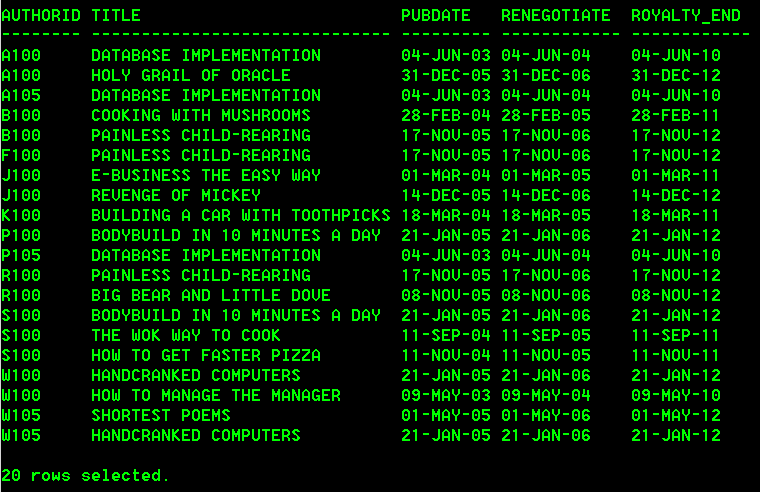
FROM orders

ORDER BY lead\_time DESC;



Each book needs to be evaluated after a year for renegotiation of royalty payments after it was published. The royalty payments will be drop in the 7th year after the book was published.

Figure out the **renegotiation date** and **royalty drop date for each book.**



COLUMN authorid FORMAT A8

COLUMN renegotiate FORMAT A12

COLUMN royalty\_end FORMAT A12

SELECT authorid, title, pubdate,

**ADD\_MONTHS**(pubdate, 12) renegotiate,

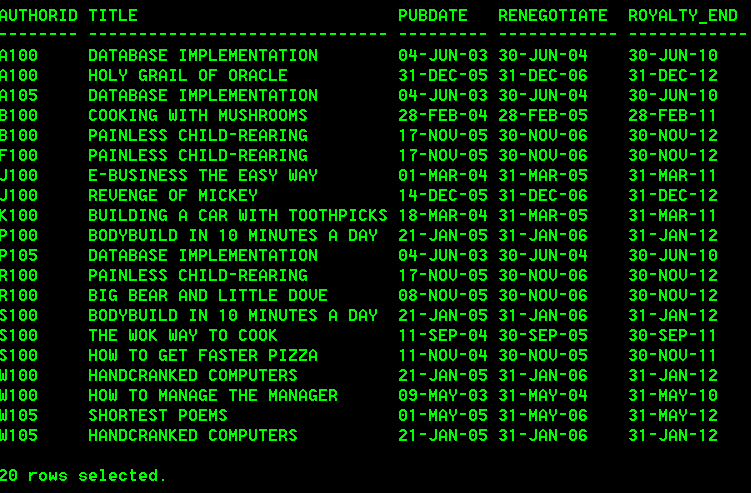
**ADD\_MONTHS** (pubdate, 84) royalty\_end

FROM books JOIN bookauthor USING(isbn)

JOIN authors USING(authorid)

ORDER BY 1;

(Continued from previous question) **The renegotiation and royalty drop dates should always take place on the last day of the specified month. F**igure out the renegotiation date and royalty drop date.



SELECT authorid, title, pubdate,

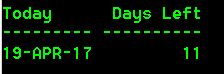
**LAST\_DAY**(ADD\_MONTHS(pubdate), 12) renegotiate,

**LAST\_DAY** (ADD\_MONTHS (pubdate), 84) royalty\_end

FROM books JOIN bookauthor USING(isbn) JOIN authors USING(authorid)

ORDER BY 1;

How do you figure out today's date and how many days left in this month?



SELECT SYSDATE "Today",

**LAST\_DAY(**SYSDATE**) –** SYSDATE "Days Left"

FROM DUAL;

One of the store customer service staff's duties is to contact the customers by email for the survey on customer satisfaction on **the** **following Friday** after the date of shipment for each order. Create a query to produce the following report:



set linesize 100

COLUMN make\_contact FORMAT A20

COLUMN email FORMAT A17

SELECT customer#, lastname, email, order#, shipdate,

**NEXT\_DAY(**shipdate, **‘FRIDAY’)** make\_contact

FROM customers JOIN orders USING(customer#)

WHERE shipdate IS NOT NULL

ORDER BY shipdate;

(Continued from previous question) How do you display the contact date (including the day of the week) with a special data format shown as below?



set linesize 120

COLUMN make\_contact FORMAT A30

SELECT customer#, lastname, email, order#, shipdate,

TO\_CHAR(NEXT\_DAY(shipdate, 'FRIDAY'), ‘ MONTH DD-DAY-‘) make\_contact

FROM customers JOIN orders USING(customer#)

WHERE shipdate IS NOT NULL

ORDER BY shipdate;

**Create a query lists employees' info, along with their annual pay. Note a $ sign should be placed preceding the value of annual pay.**

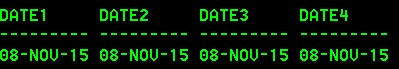


SELECT empno, lastname, TO\_CHAR(monthly\_salary\*12, ‘$999,999.99’) annual\_pay

FROM emp

ORDER BY 3 DESC;

**There is a date field storing the hire date of employees. However, the column was not set with an input format. As a result, the column may be entered in various ways by different individuals. For example, for the date 'November 8, 2015', we might see such entries: 2015/11/08, 110815, 20151108, or 20150811. How do convert these various forms of data entry into an Oracle default date format, 'DD-MON-YY'?**



SELECT TO\_DATE('2015/11/08', ‘yyyy/mm/dd’) date1,

TO\_DATE ('110815', ‘mmddyy’) date2,

TO\_DATE('20151108', ‘yyyymmdd’) date3,

TO\_DATE('20150811', ‘yyyyddmm’) date4

FROM dual;

**Create the following report listing empno, last name, year of hire date, and years of employment.**



SELECT empno, lastname,

\_\_\_\_\_(hiredate, 'YYYY') year\_hire,

\_\_\_\_\_(\_\_\_\_\_(SYSDATE, 'YYYY')) -

\_\_\_\_\_(\_\_\_\_\_(hiredate, 'YYYY')) year\_employment

FROM emp

ORDER BY 4 DESC;

MORE TO COME…