**Name: Aaron Lim**

SQL Programming – Level 1 Programming Project 03

# Comparison operators (LIKE) | Functions | Literals

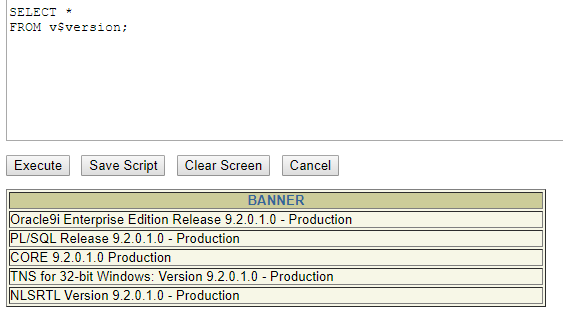
***Reminder: read the Project Guidelines document for instructions on how to format and submit your assignments.***

## Part 1 – use the Oracle 9i server for the following problems.

Demonstrate that you are using the Oracle 9i server by issuing:

SELECT \*

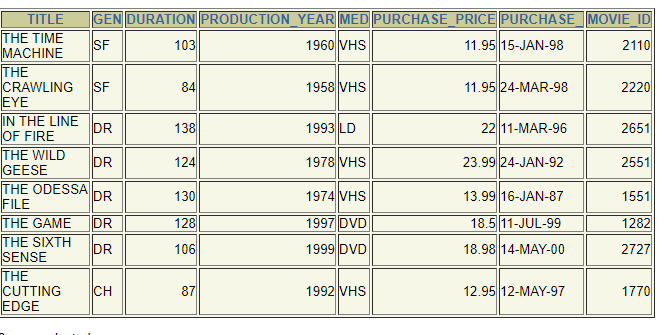
FROM v$version;



1. Show the title of all movies whose title is a single word (use LIKE). This problem is a little bit trickier than I intended. If you get stuck, think about character sets, and ask yourself, what does a two-word title have that a single word title does not?
2. Show the title of all movies whose title includes the word *the.* This is not a trick question – when I say the word ‘*the’* I mean the word ‘*the’*, not the letters: *t h e*. The following title does NOT contain the word *the* THERE IS A HOUSE ON A HILL. To simplify matters, you may presume that no movie is simply titled *The,* nor is *the* the last word in any title (use LIKE).

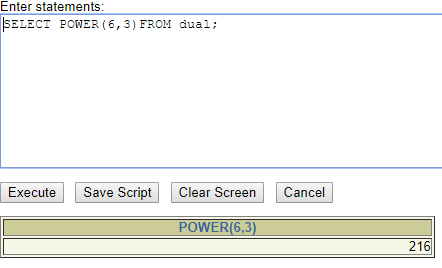
SELECT \* FROM movie

WHERE TITLE LIKE '%TH%E'



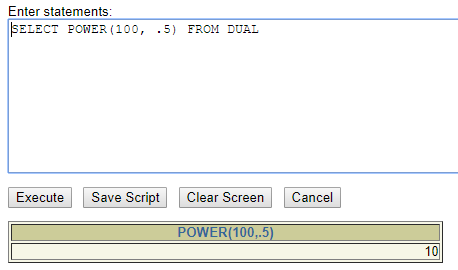
1. Calculate and show the value of 6 raised to the third power (ie. 63).

SELECT POWER(6,3)FROM dual;



1. Calculate and show the square root of 100 (ie. Raised to the ½ power)

SELECT POWER(100, .5) FROM DUAL

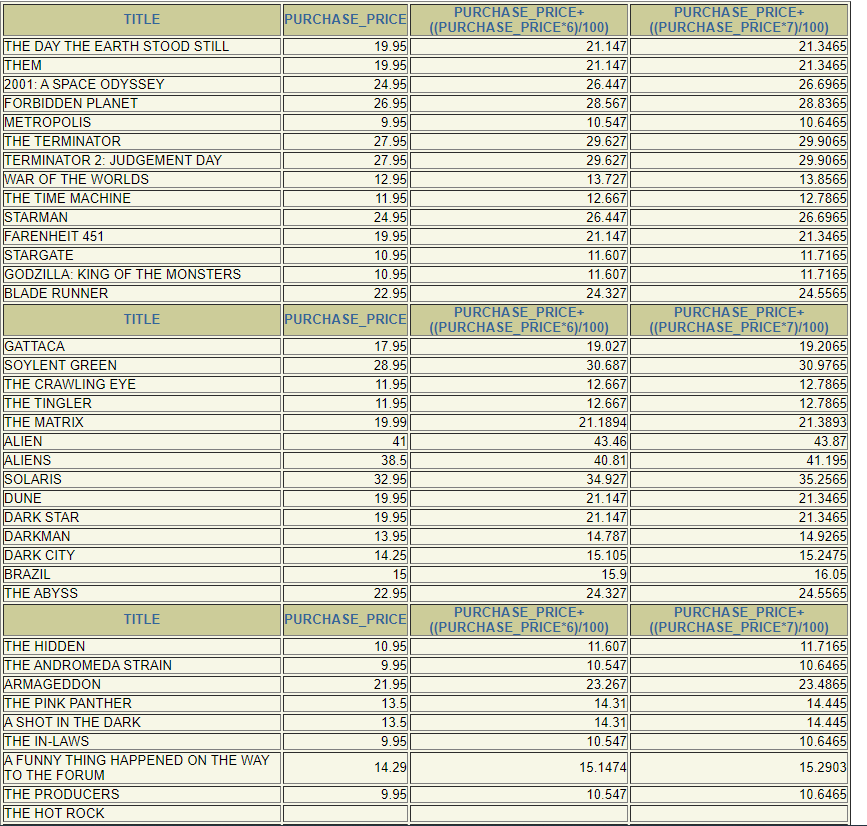


1. For all of the Shakespeare titles in the collection, show the title, cost, and cost + 6% and cost + 7%. Use an appropriate title for the calculated columns. [4 cols]

SELECT TITLE, PURCHASE\_PRICE,PURCHASE\_PRICE + ((PURCHASE\_PRICE \* 6) / 100),PURCHASE\_PRICE + ((PURCHASE\_PRICE \* 7) / 100)

FROM movie;

WHERE gen = 'SF';



1. For all of the Shakespeare titles in the collection, show the title, cost, and cost + 6% and cost + 7%. Use an appropriate title for the calculated columns. Round all calculations to two decimal places. [4 cols]

SELECT TITLE, PURCHASE\_PRICE,ROUND(PURCHASE\_PRICE + ((PURCHASE\_PRICE \* 6) / 100),2),ROUND(PURCHASE\_PRICE + ((PURCHASE\_PRICE \* 7) / 100),2)

FROM movie;

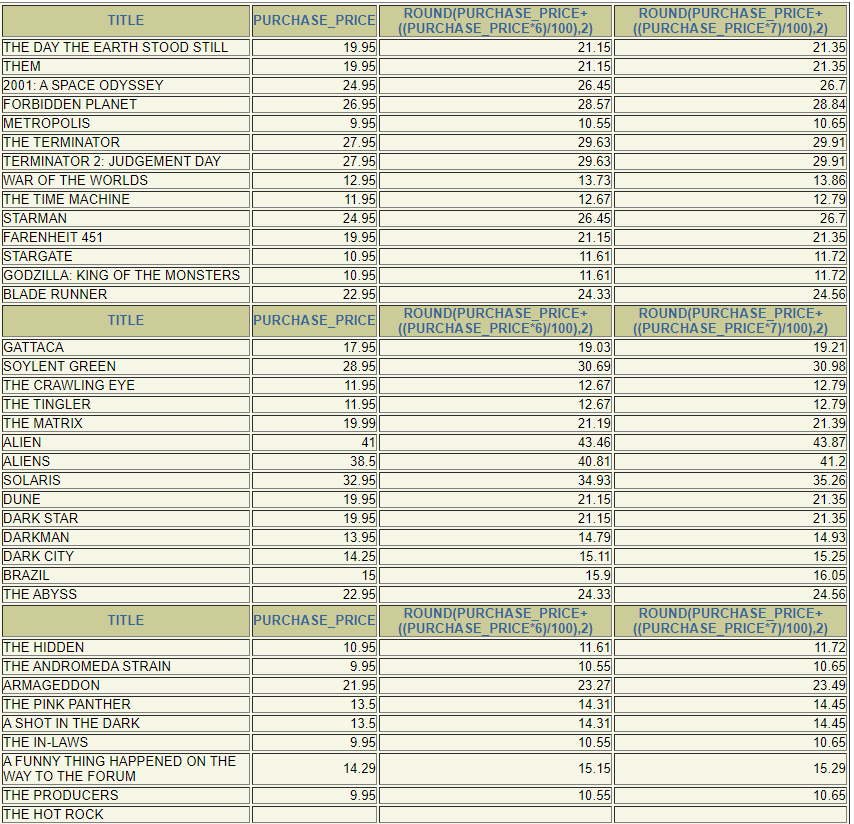
WHERE gen = 'SF'

1. For all of the Shakespeare titles in the collection, show the title, cost, and cost + 6% and cost + 7%. Use an appropriate title for the calculated columns. Truncate all calculations to two decimal places. [4 cols]

SELECT TITLE, PURCHASE\_PRICE,ROUND(PURCHASE\_PRICE + ((PURCHASE\_PRICE \* 6) / 100),2),ROUND(PURCHASE\_PRICE + ((PURCHASE\_PRICE \* 7) / 100),2)

FROM movie;

WHERE gen = 'SF';



1. For each of the CH films (genre=CH), show the title in all lower case characters.

SELECT LOWER(TITLE)

FROM movie

WHERE GEN = 'CH';

1. For each of the CH films (genre=CH), display the title in such a fashion that only the first letter in each word is capitalized.

SELECT UPPER(title)

FROM movies

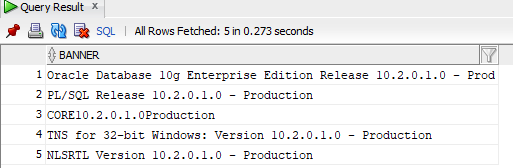
WHERE genre = 'CH'

## Part 2 – use the Oracle 10g server for the following problems.

Demonstrate that you are using the Oracle 10g server by issuing:

SELECT \*

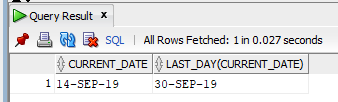
FROM v$version;



1. Show today’s date, as well as the last day (date) of the current month. [2 cols]

SELECT CURRENT\_DATE, LAST\_DAY(CURRENT\_DATE)

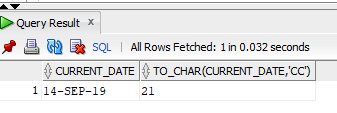
FROM DUAL



1. Display today’s date and the two digits of the century for today’s date. (For example, April 13, 2004 occurs in the 21st century) [2 cols]

SELECT CURRENT\_DATE, TO\_CHAR(CURRENT\_DATE, 'CC')

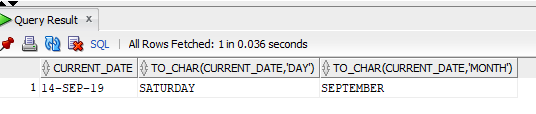
FROM DUAL



1. Display today’s date and (the name of) the day of the week it is, and the name of the month in which it occurs. [3 cols]

SELECT CURRENT\_DATE, TO\_CHAR(CURRENT\_DATE, 'DAY'), TO\_CHAR(CURRENT\_DATE, 'MONTH')

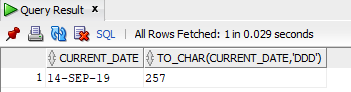
FROM DUAL



1. Display today’s date and the day of the year (sometimes referred to as the Julian Day, or Julian Date) for today’s date. [2 cols]

SELECT CURRENT\_DATE, TO\_CHAR(CURRENT\_DATE, 'DDD')

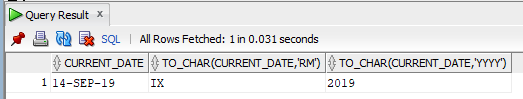
FROM DUAL



1. Display today’s date and the Roman Numerals for both the month and year values of today’s date. [3 cols]

SELECT CURRENT\_DATE, TO\_CHAR(CURRENT\_DATE, 'RM'),TO\_CHAR(CURRENT\_DATE, 'YYYY')

FROM DUAL



1. Display today’s date and the name of the month, in lower case, for the month 3 month’s hence (ie today’s date + 3 months). [2 cols]

SELECT CURRENT\_DATE, TO\_CHAR(ADD\_MONTHS(CURRENT\_DATE, 3),LOWER('MONTH'))

FROM DUAL

