**USEFUL LINKS**

https://www.w3resource.com/mysql/date-and-time-functions/mysql-convert\_tz-function.php

https://dev.mysql.com/doc/refman/5.7/en/date-and-time-functions.html

https://publib.boulder.ibm.com/tividd/td/TWS/SC32-1274-02/en\_US/HTML/SRF\_mst273.htm

**Assignment**

-- update commission for all employees who are manager(2 employees to 15%), manager for

  more than 2 employees give 20% commission)

-- calculate the increment based on commission\_pct

-- Select last\_name, if(commission\_pct> ? ,'good','avg') rating from employees;

Amazon mobile products table --

OS company\_name model cost

Android Samsung galaxy 67

Android Google nexus 67

Android Google motorola 67

Ios apple 6s 70

Ios apple 7s 80

Costliest mobile under apple

Cheapest mobile under samsung

Questions for interview

Joins

Group by, group functions

Sub queries

DDL, DCL, DML

Primary key, foreign key

Set operators - union, union all

Drop, delete, truncate

SQL CLASS 3

Functions- single row & multi row functions

Single row - character functions, number functions, date & time functions

Implicit and explicit conversions

Conditional functions

Group functions

utf8 and utf16

**Functions**

Functions are set of sql statements that accept input parameter, perform actions and return the result.

 There are user defined functions, system functions / in-built functions

Single row functions and multi row functions

100 rows single row when u pass a query it returns 100 rows

Single row functions are those which return a result for each row in the table

Like if there are 5 rows in a table the command will return 5 rows as results.

Mysql has some inbuilt functions

for eg all string functions.

Substr, length

select \* from employees;

select first\_name from employees where employee\_id = 100;

select first\_name, substr(first\_name, 1, 4) from employees where employee\_id = 100;

-- Substr (arg, starting index, number of characters required)

empid first 3 character of last name employee add

length function - will return the number of characters in the string data type

select first\_name, length(first\_name) from employees where employee\_id = 100;

If textbox is 10 space and you want to  right align use lpad

There are other string in built functions in sql

They are also called Character manipulation functions

instr, lpad, rpad, trim, replace

**Case manipulation - lower, upper, initcap**

**examples of case manipulation**

select \* from employees where upper(email)='SKING' ;

select \* from employees where email=upper('sking');

Both will return same output.

select upper('john'); --use for learning the function

  in oracle you need to use dual

select upper('john') from dual;

select upper('john'), lower('john');

**Nested function**

select length(concat('john', 'candy')) ;

select instr('sql', 'q') java a it returns index of first occurance

lpad allow to pad the length of space

select lpad('java', 10, ' ');

same for rpad('java', 10, '\*');

select trim (' java ');

select length (' java '), length( trim (' java ');

select trim('H' from 'HelloworldH');

select replace('jack and jue', 'j','bl');

select first\_name, instr(first\_name, 'a') from employee;

**Number Manipulation functions**

round(50.123456, 5)

trunc(50.123456,5)

mod

All these functions are used after you fetch from database you can format or manipulate and use it in the application.

select round(50.123456, 5), round(50.123456, 2), round(50.123456, 0);

select round(50. 698, 2), round(50.698, 0);

round(44.567, -1) --- -1 will work on tens place -2 hundred place

If the number in tens place is half of ten round to nearest tens

In oracle is uses a table called dual for no table option

Truncate - will not check for remaining digits it will not compare

Select truncate(45.5567, 2), truncate(45.987,0);

Remainder -- Difference between quotient and modulus

Select mod(50,2);

Operations on Date columns

Formatting date, date and time operation, +/- 7 business days

-- yyyy-mm-dd people hired after the particular date

select \* from employees where hire\_date > '1987-06-17';

select sysdate(); date along with time it is an inbuilt function and it can be used in your queries

-- to subtract one month from todays date, you can also add date, can use one day / year

As per business requirement you can display the warranty period

Select subdate(sysdate(), interval 1 month);

Select adddate(sysdate(), interval 1 month);

Select adddate(sysdate(), interval 100 day);

Select subdate(‘2017-01-30’, interval 1 month);

Select subdate(‘2017-01-30’, interval 7 day);

-- difference between two dates

Select datediff(‘2107-02-20’,’2017-01-30’);

--

Select extract(year from sysdate());

-- find current year, month, day

Select extract(year from sysdate()), extract(month from sysdate()), extract(day from sysdate());

Select dayofmonth(sysdate()), dayofweek(‘2017-01-30’), dayofyear(sysdate();

Select curdate();

-- any difference between sysdate and curdate ---sysdate is with timestamp

Select current\_time();

Select dayname(Sysdate()); -- gives alphabetical day

-- versions keep updating the commands so please keep checking before implementing

-- formatting is for displaying in ui

Select date\_format(curdate(),”%d/%m/%y”);

Select date\_format(sysdate(), “%h:%i,%s %p”);

Select date\_format(current\_time(), “%r”);

-- default time format as h m s am/pm its a short cut

Select time\_format(current\_time(), “%r”);

Select time\_format(‘15:00’,”%r”);

-- You can combine like last day of the month

Select last\_day(sysdate());

Nested functions can be done like find first day of next month

Select adddate(last\_day(sysdate()), interval 1 day);

One more we need to go through is the timestamp

Let s go thro time stamp GMT is considered common time zone

Based on the application we might have to convert the timezones

Convert\_tz convert from one tmezone to another

-- SELECT CONVERT\_TZ('2004-01-01 12:00:00','GMT','MET');

--SELECT TIMEDIFF('2000:01:01 00:00:00', '2000:01:01 00:00:00.000001');

Timestamp

SELECT TIMESTAMP('2003-12-31');

SELECT TIMESTAMP('2003-12-31 12:00:00','12:00:00');

Presentation is based on oracle, we are executing in my sql as it is mostly used and open source too.

From pg 114 explain there will difference in the syntax

You round and truncate on date too.

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

Describe employee;

Explain decimal and integer;

Implicit Data type conversion

Varchar /char ---- number / int

 Varchar /char ---- date

number / int ---- varchar

Date ----- varchar

Select \* from employees where salary > 2400;

Select \* from employees where salary > '2400';

Select \* from employees where employee\_id= '100';

select \* from employees where hire\_date > '1987-06-17';

-- for expression evaluation the data type conversion is implicit

-- conversion occurs for comparison operators

SELECT 1 > '6x';

Explicit data type conversion

-- in oracle To\_number, to\_char, to\_date

Number ---> char

Date ----> char

Char ----> number

Char -----> date

To\_char --- takes date and format

In mysql it is date(), dateformat(), cast(), convert()

SELECT 38.8, CAST(38.8 AS CHAR);

--SELECT CONVERT(\_latin1'Müller' USING utf8);

ASCII, UTF8, UTF16 Unicode Text Format

Null functions

Select \* from employees;

-- Commission pct is null based on that we need to do some functions

-- We need to calculate the increment

Select salary, salary + (commission\_pct\*salary) incsalary from employees;

--- The calculation goes null when operation is with null

--- To avoid this we have some functions

Select commission\_pct, ifnull(commission\_pct,0) from employees;

Select salary, salary + (ifnull(commission\_pct,0)\*salary) incsalary from employees;

-- simple if else

Select last\_name, if(commission\_pct>0.3,'good','avg') rating from employees;

-- nullif (p1,p2)

-- if both are equal --null otherwise p1 is returned

Select nullif('hi','hi');

Select nullif('hi','hello');

Select first\_name, length(first\_name), last\_name, length(last\_name),

 nullif(length(first\_name),length(last\_name)) result from employees;

-coalease is similar to nullif- it takes the first not null value

Select last\_name, manager\_id,commission\_pct, coalesce(manager\_id, commission\_pct, -1) comm from employees order by commission\_pct;

Select coalesce('hi', null, 0);

Select coalesce(null, null, 'p3', 'p4',0);

---looks for the first not null value otherwise returns default value

If- else statements

Case column name when ‘value1’ then execute1

  When ‘value2’ then execute2

When ‘value3’ then execute3

  Else execute end

Select last\_name, job\_id, salary,

case job\_id when 'IT\_PROG' then 1.10\*salary

When 'ST\_CLERK' then 1.15\*salary

When 'SA\_REP' then 1.20\*salary

Else salary end 'REVISED\_SALARY'

from employees;

--------

That is all about single row functions now we will go through group functions

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

Group functions and group by

-- single row function we saw multiple rows input and output

Select lower(first\_name) from employees;

---- average - multiple rows input single output

Select avg(salary) from employees;

Select max(salary) from employees;

Select min(salary) from employees;

Select sum(salary) from employees;

--

Select count(\*) from employees ;

--

Select count(\*) from employees where salary > 10000;

-- With group functions you should not add individual columns

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

Group by …..having

When u want to group some data based on requirement and do some operations

Useful when we have to execute group functions after we group data

Like if we have to find the highest salary in each department

Minimum salary in each department

Employees whose salary is less than average in each department

Amazon mobile products table --

OS company\_name model cost

Android Samsung galaxy 67

Android Google nexus 67

Android Google motorola 67

Ios apple 6s 70

Ios apple 7s 80

Costliest mobile under apple

Cheapest mobile under samsung

Select max(cost) from products group by company\_name having company\_name=’apple’

Select min(cost) from products group by company\_name having company\_name=’samsung’

Subgroup

Select max(cost) from products group by OS, company\_name having company\_name=’apple’

-- find all job id in each department

Select job\_id, department from employees group by job\_id;

-- find max salary under each department

Select max(salary), department\_id from employees group by department\_id;

-- only for one dept filtering

Select max(salary), department\_id from employees group by department\_id having department\_id=40;

-- sub group

-- find all department for each job

Select department\_id, job\_id from employees group by job\_id, department\_id;

Select max(salary), department\_id, job\_id from employees group by department\_id, job\_id;

-- non grouped column name cannot be used with the query