**Day 1:**

**1) WAQ to find Accounting department with valid managers (Use departments table)**

SELECT \* from departments where MANAGER\_ID is not null and department\_name='Accounting';

**2) WAQ to find employees who have steve in their first name or have a commission\_Pct**.

select first\_name,commission\_pct from employees where first\_name = '%STEVE%' or commission\_pct is not null;

**3) WAQ to find Unique Salary.**

select distinct salary from employees;

**Day 2:**

char funcs:

**1.Get ful name of employee and print the ful name in upper case**

select initcap(concat(concat(FIRST\_NAME,' '), Last\_name)) as fullname from EMPLOYEES;

select first\_name, last\_name, upper(concat(first\_name, last\_name)) full\_name from employees;   (--> With no space between first\_name and last\_name)

select first\_name, last\_name, upper(concat(rpad(first\_name, length(first\_name)+1, ' '), last\_name)) full\_name from employees;    (--> With space in between first\_name and last\_name)

select upper(first\_name) || ' ' || upper(last\_name) from employees;    (--> Simple query with space in between first and last names)

**2.Waq to print sysdate in differnent date formats and that should print current time also**

select sysdate, to\_char(sysdate, 'dd mon yy hh:mm:ss AM') date\_format\_1, to\_char(sysdate, 'dy dd mon yyyy hh24:mm:ss') date\_format\_2, to\_char(sysdate, 'day dd month year hh:mm:ss AM') date\_format\_3 from employees;

to\_char, to\_date

**Day 3:**

Group by:

1. **Calculate minimum salary under each department**

**of employees table.**

  select min(salary), department\_id from employees group by department\_id

1. **Calculate sum of salaries for each type of job**

**under my employees.**

 select sum(salary), job\_id from employees group by job\_id

1. **Calculate average salary of particular job**

**type in each department.**

  select department\_id, job\_id,avg(salary) from employees group by department\_id, job\_id

1. **Get emp id, emp first name, department name using employees and departments.**

 select e.employee\_id, e.first\_name, d.department\_name from employees e join departments d using (department\_id)

**Useful links for date, number formats and functions:**

<https://docs.oracle.com/cd/B28359_01/server.111/b28286/sql_elements004.htm>

<https://docs.oracle.com/cd/B28359_01/server.111/b28286/functions242.htm>

Inner Join : It will give you only matching rows b/n two tables based on condition you are querying for..

Outer Join : It will give you matching and non matching rows from 2 tables.

Left Outer Join : It gives matching and unmatching rows from left table and only the matching rows from right table.

Right Outer Join : It gives matching and unmatching rows from right table and only the matching rows from left table.

Any – checks for or condition

All – checks for and condition

Hands on -Joins, sub queries, DMl

Employees from each Department whose minimum salary is less than average salary.

Union – all distinct rows from two queries

Union all- all rows including duplicates

Intersection – common rows

Minus – subtract one query resule from other

DML - Data manipulation language- update, insert, delete, select

DDL— Data definition language- create, alter, drop, truncate

DCL – Data control language - grant, revoke

TCL- transaction control language- commit, rollback, savepoint

PK – uniquely identify the row of the table using given column, pk is unique and not null

Composite primary key – combination of more than one columns is unique

Max of 16 columns can be used as composite PK.

FK- to establish some relation b/n 2 tables, field or column in second table which is a primary key in first table. fk can be null, can be non-unique or duplicated

Composite primary key – combination of more than one columns is unique

Deptid- dept table(pk)

Deptid-emp(fk)

Unique – which should be distinct or no duplications of values

Not null

Check – check fro any given condition and restrict data accordingly

Joins

Subqueries

Dml select

DDl

Dcl,tcl

constraints