Oracle DB – Cheat Sheet on Null and Conditional Functions
(Not an Exhaustive List)

1. Calculations that use a null value result in null
2. **NVL**(value that may contain null, value to replace null)
    a. Same data type in null and replacement cols
    b. Examples
        i. NVL(comment_col,'no comment')
        ii. NVL(expense_amt,0)
        iii. NVL(hire_date,'01-JAN-02')
    c. List employee commission amounts
        i. SELECT last_name, nvl(commission_pct,0)
           FROM employees;
        ii. SELECT last_name,
              nvl(commission_pct,0) * salary as "Commission Amount'
           FROM employees;
    d. Correct for null in expense_amt before calculating tax:
       SELECT item,expense_amt,NVL(expense_amt,0)*.07 AS 'Tax'
       FROM item_table;
3. **NVL2**(expr1 that may be NULL, expr2 to return if NOT NULL, expr3 to return if NULL)
    a. expr1 can be any data type
    b. expr2 & expr3 can be any type except LONG
    c. If expr2 is character data then a VARCHAR2 is returned; max size is 2GB
    d. Data type for expr2 and expr3 must be same type
    e. Example
        i. SELECT last_name, first_name,
              NVL2(comm_pct,salary*(1+comm_pct),salary) AS "Gross Pay"
           FROM employees
        ii. SELECT last_name,
              NVL2(commission_pct,'Commission','No Commission')
           FROM employees
4. **NULLIF**(expr1, expr2)
    a. Returns NULL if expr1=expr2
    b. Returns expr1 if not equal
    c. Example
       SELECT last_name, nullif(commission_pct,.2)
       FROM employees
       WHERE NOT commission_pct IS NULL;
       (can also use IS NOT)
5. **COALESCE**(expr1, expr2, expr3, … exprn)
    a. If expr1 is NULL, use expr2
    b. If expr2 is NULL, use expr3, etc. until a not NULL expr is found
    c. can use in place of NLV2
    d. Example
       SELECT last_name, commission_pct, salary,
          COALESCE(commission_pct,0) *salary + salary AS "Pay"
       FROM employees;

6. **CASE** expr
   WHEN comparison1 THEN return_expr1
   WHEN comparison2 THEN return_expr2
   …
   WHEN comparison THEN return_exprn
   ELSE else_expr
   END
   a. Return a value when a match is found
   b. Can work with comparisons
   c. Example
      i. SELECT last_name, commission_pct,
         (CASE commission_pct
           WHEN 0.1 THEN 'Low'
           WHEN 0.15 THEN 'Average'
           WHEN 0.2 THEN 'High'
           ELSE 'N/A'
         END) Commission
        FROM employees
        ORDER BY last_name
      ii. SELECT last_name, job_id, salary,
         (CASE
           WHEN job_id LIKE 'SA_MAN' AND salary < 12000 THEN '10%'
           WHEN job_id LIKE 'SA_MAN' AND salary >= 12000 THEN '15%'
           WHEN job_id LIKE 'IT_PROG' AND salary < 9000 THEN '8%'
           WHEN job_id LIKE 'IT_PROG' AND salary >= 9000 THEN '12%'
           ELSE 'NOT APPLICABLE'
         END) Raise
        FROM employees;
7. **DECODE** (col/expr, search1, result1, search2, result2, …,
   searchn, resultn, default)
   a. Oracle proprietary
   b. exact matches only
   c. if default is omitted NULL is returned if no match is found in search values
   d. Example
      SELECT last_name, commission_pct,
      DECODE (commission_pct,
        0.1,'Low', 0.15, 'Average', 0.2, 'High', 'N/A')
      FROM employees
      ORDER BY last_name;