Michael Longstreth
Final Project
December 22, 2017

1.

```
SQL> SELECT last_name, first_name, salary, hire_date, department_id, manager_id
  2  FROM employees
  3  WHERE hire_date < '01-JAN-00' AND department_id IN (80, 100, 110)
  4  ORDER BY department_id, manager_id, last_name;

LAST_NAME                 FIRST_NAME               SALARY HIRE_DATE DEPARTMENT_ID MANAGER_ID
------------------------- -------------------- ---------- --------- ------------- ----------
Cambrault                 Gerald                    11000 15-OCT-99            80        100
Errazuriz                 Alberto                   12000 10-MAR-97            80        100
Partners                  Karen                     13500 05-JAN-97            80        100
Russell                   John                      14000 01-OCT-96            80        100
Bernstein                 David                      9500 24-MAR-97            80        145
Cambrault                 Nanette                    7500 09-DEC-98            80        145
Hall                      Peter                      9000 20-AUG-97            80        145
Olsen                     Christopher                8000 30-MAR-98            80        145
Tucker                    Peter                     10000 30-JAN-97            80        145
Tuvault                   Oliver                     7000 23-NOV-99            80        145
Doran                     Louise                     7500 15-DEC-97            80        146
King                      Janette                   10000 30-JAN-96            80        146
McEwen                    Allan                      9000 01-AUG-96            80        146
Sewall                    Sarath                     7000 03-NOV-98            80        146
Smith                     Lindsey                    8000 10-MAR-97            80        146
Sully                     Patrick                    9500 04-MAR-96            80        146
Greene                    Danielle                   9500 19-MAR-99            80        147
Vishney                   Clara                     10500 11-NOV-97            80        147
Bates                     Elizabeth                  7300 24-MAR-99            80        148
Bloom                     Harrison                  10000 23-MAR-98            80        148
Fox                       Tayler                     9600 24-JAN-98            80        148
Ozer                      Lisa                      11500 11-MAR-97            80        148
Smith                     William                    7400 23-FEB-99            80        148
Abel                      Ellen                     11000 11-MAY-96            80        149
Hutton                    Alyssa                     8800 19-MAR-97            80        149
Livingston                Jack                       8400 23-APR-98            80        149
Taylor                    Jonathon                   8600 24-MAR-98            80        149
Greenberg                 Nancy                     12000 17-AUG-94           100        101
Chen                      John                       8200 28-SEP-97           100        108
Faviet                    Daniel                     9000 16-AUG-94           100        108
Popp                      Luis                       6900 07-DEC-99           100        108
Sciarra                   Ismael                     7700 30-SEP-97           100        108
Urman                     Jose Manuel                7800 07-MAR-98           100        108
Higgins                   Shelley                   12000 07-JUN-94           110        101
Gietz                     William                    8300 07-JUN-94           110        205

35 rows selected.
```

2.

```
SQL> SELECT first_name, last_name, NVL(TRIM(SUBSTR(last_name,3,1)),0) AS "3.ch"
  2  FROM employees
  3  ORDER BY "3.ch";

FIRST_NAME           LAST_NAME                 3.ch
-------------------- ------------------------- ----
Lex                  De Haan                   0
Samuel               McCain                    C
Allan                McEwen                    E
Douglas              Grant                     a
Kimberely            Grant                     a
Jennifer             Whalen                    a
Sigal                Tobias                    b
Anthony              Cabrio                    b
Neena                Kochhar                   c
Peter                Tucker                    c
Sundar               Ande                      d
Renske               Ladwig                    d
Ellen                Abel                      e
Hermann              Baer                      e
Kelly                Chung                     u

FIRST_NAME           LAST_NAME                 3.ch
-------------------- ------------------------- ----
Payam                Kaufling                  u
Kevin                Mourgos                   u
Curtis               Davies                    v
Daniel               Faviet                    v
Jack                 Livingston                v
Susan                Mavris                    v
Oliver               Tuvault                   v
Sarath               Sewall                    w
Tayler               Fox                       x
Pat                  Fay                       y
Julia                Nayer                     y
Jonathon             Taylor                    y
Winston              Taylor                    y

107 rows selected.
```

The questions did not ask to have the character data in alphabetical order (ignoring precedence of case sensitivity), but if you wanted it to be in alphabetical order, you could change the ORDER BY clause to the following: ORDER BY LOWER(“3.ch”);

3.

```
SQL> SELECT first_name, last_name, salary
  2  FROM employees
  3  WHERE LENGTH(last_name) = 5 AND (SUBSTR(last_name,2,1) = 'r' OR SUBSTR(last_name,4,1) = 'e')
  4  ORDER BY last_name;

FIRST_NAME           LAST_NAME                     SALARY
-------------------- ------------------------- ----------
Elizabeth            Bates                           7300
Bruce                Ernst                           6000
Adam                 Fripp                           8200
Timothy              Gates                           2900
Douglas              Grant                           2600
Kimberely            Grant                           7000
Vance                Jones                           2800
Julia                Nayer                           3200
Christopher          Olsen                           8000
Joshua               Patel                           2500
Jose Manuel          Urman                           7800

11 rows selected.
```

4.

```
SQL> SELECT LENGTH(last_name), LENGTH(first_name), COUNT(employee_id)
  2  FROM employees
  3  GROUP BY LENGTH(last_name), LENGTH(first_name)
  4  ORDER BY LENGTH(last_name) DESC, LENGTH(first_name) ASC;

LENGTH(LAST_NAME) LENGTH(FIRST_NAME) COUNT(EMPLOYEE_ID)
----------------- ------------------ ------------------
               11                  5                  1
               10                  4                  1
               10                  5                  2
                9                  5                  4
                9                  6                  1
                9                  7                  3
                8                  3                  1
                8                  5                  3
                8                  6                  2
                8                  7                  1
                7                  3                  1
                7                  4                  1
                7                  5                  5
                7                  6                  4
                7                  7                  4
                6                  3                  1
                6                  4                  1
                6                  5                 11
                6                  6                  7
                6                  7                  4
                6                  8                  3
                6                  9                  1
                5                  2                  1
                5                  4                  2
                5                  5                  5
                5                  6                  4
                5                  7                  9
                5                  8                  2
                5                  9                  2
                5                 11                  2
                4                  4                  3
                4                  5                  3
                4                  6                  4
                4                  7                  2
                4                  9                  1
                3                  2                  1
                3                  3                  1
                3                  4                  1
                3                  5                  1
                3                  6                  1

40 rows selected.
```

I included the two additional columns ‘LENGTH(last_name)’ and ‘LENGTH(first_name) above to check my results. If I wrote the statement as the question is asked the SELECT clause would only include, COUNT(employee_id)

5.

```
SQL> SELECT last_name, department_id, commission_pct AS comm_pct, (salary * NVL(commission_pct,0)) AS comm_amt
  2  FROM employees;

LAST_NAME                 DEPARTMENT_ID   COMM_PCT   COMM_AMT
------------------------- ------------- ---------- ----------
King                                 90                     0
Kochhar                              90                     0
De Haan                              90                     0
Hunold                               60                     0
Ernst                                60                     0
Austin                               60                     0
Pataballa                            60                     0
Lorentz                              60                     0
Greenberg                           100                     0
Faviet                              100                     0
Chen                                100                     0
Sciarra                             100                     0
Urman                               100                     0
Popp                                100                     0
Bates                                80        .15       1095
Kumar                                80         .1        610
Abel                                 80         .3       3300
Hutton                               80        .25       2200
Taylor                               80         .2       1720
Livingston                           80         .2       1680
Grant                                          .15       1050
Johnson                              80         .1        620
Taylor                               50                     0
Fleaur                               50                     0
Sullivan                             50                     0
Geoni                                50                     0
Sarchand                             50                     0
Bull                                 50                     0
Dellinger                            50                     0
Cabrio                               50                     0
Chung                                50                     0
Dilly                                50                     0
Gates                                50                     0
Perkins                              50                     0
Bell                                 50                     0
Everett                              50                     0

LAST_NAME                 DEPARTMENT_ID   COMM_PCT   COMM_AMT
------------------------- ------------- ---------- ----------
McCain                               50                     0
Jones                                50                     0
Walsh                                50                     0
Feeney                               50                     0
OConnell                             50                     0
Grant                                50                     0
Whalen                               10                     0
Hartstein                            20                     0
Fay                                  20                     0
Mavris                               40                     0
Baer                                 70                     0
Higgins                             110                     0
Gietz                               110                     0

107 rows selected.
```

```
SQL> SELECT last_name, salary, comm_pct, comm_amt, (salary + comm_amt) AS current_ful_sal,
  2  CASE
  3     WHEN comm_pct > 0 THEN (salary * 1.1) + comm_amt
  4     WHEN comm_pct IS NULL THEN (salary * 1.15) + comm_amt
  5  END AS full_sal_raise
  6  FROM
  7  (
  8     SELECT last_name, salary, department_id, commission_pct AS comm_pct, (salary * NVL(commission_pct,0)) AS comm_amt
  9     FROM employees
 10  )
 11  WHERE department_id = 100
 12  ORDER BY current_ful_sal;

LAST_NAME                    SALARY   COMM_PCT   COMM_AMT CURRENT_FUL_SAL FULL_SAL_RAISE
------------------------- ---------- ---------- ---------- --------------- --------------
Popp                            6900                    0            6900           7935
Sciarra                         7700                    0            7700           8855
Urman                           7800                    0            7800           8970
Chen                            8200                    0            8200           9430
Faviet                          9000                    0            9000          10350
Greenberg                      12000                    0           12000          13800

6 rows selected.
```

In the first statement for question five, which we are to use as a subquery in the FROM clause, ‘salary’ is not one of the column values included in the statement. If we have to use the exact statement as the subquery in the FROM clause, and cannot add ‘salary’ to the subquery, as I have done above, you would have to run the next statement. The next statement is not ideal, as ‘employee_id’ is not in the subquery and joining the tables only gives us the correct results because the six last names for employees in department_id = 100, are unique.

```
SQL> SELECT a.last_name, a.salary, b.comm_pct, b.comm_amt, (a.salary + b.comm_amt) AS current_ful_sal,
  2  CASE
  3     WHEN b.comm_pct > 0 THEN (a.salary * 1.1) + b.comm_amt
  4     WHEN b.comm_pct IS NULL THEN (a.salary * 1.15) + b.comm_amt
  5  END AS full_sal_raise
  6  FROM employees a INNER JOIN
  7  (
  8     SELECT last_name, department_id, commission_pct AS comm_pct, (salary * NVL(commission_pct,0)) AS comm_amt
  9     FROM employees
 10  ) b
 11  ON a.last_name = b.last_name
 12  WHERE b.department_id = 100
 13  ORDER BY current_ful_sal;

LAST_NAME                    SALARY   COMM_PCT   COMM_AMT CURRENT_FUL_SAL FULL_SAL_RAISE
------------------------- ---------- ---------- ---------- --------------- --------------
Popp                            6900                    0            6900           7935
Sciarra                         7700                    0            7700           8855
Urman                           7800                    0            7800           8970
Chen                            8200                    0            8200           9430
Faviet                          9000                    0            9000          10350
Greenberg                      12000                    0           12000          13800

6 rows selected.
```

6.

```
SQL> SELECT department_name, SUM(salary + comm_amt) AS sum_current_sal, SUM(ful_sal_raise) AS sum_sal_raise
  2  FROM
  3  (
  4     SELECT a.last_name, a.department_name, a.salary, a.comm_pct, a.comm_amt, (a.salary + a.comm_amt) AS current_ful_sal,
  5     CASE
  6             WHEN a.comm_pct > 0 THEN (a.salary * 1.1) + a.comm_amt
  7             WHEN a.comm_pct IS NULL THEN (a.salary * 1.15) + a.comm_amt
  8     END AS ful_sal_raise
  9     FROM
 10             (
 11             SELECT e.last_name, e.salary, e.department_id, e.commission_PCT AS comm_pct, (e.salary * NVL(e.commission_pct,0)) AS comm_amt, d.department_name
 12             FROM employees e INNER JOIN departments d ON e.department_id = d.department_id
 13             ) a
 14  )
 15  GROUP BY department_name
 16  ORDER BY department_name;

DEPARTMENT_NAME              SUM_CURRENT_SAL SUM_SAL_RAISE
---------------------------- --------------- ------------
Accounting                             20300        23345
Administration                          4400         5060
Executive                              58000        66700
Finance                                51600        59340
Human Resources                         6500         7475
IT                                     28800        33120
Marketing                              19000        21850
Public Relations                       10000        11500
Purchasing                             24900        28635
Sales                                 377140       407590
Shipping                              156400       179860

11 rows selected.
```

7.

i.

```
SQL> SELECT employee_id, hire_date, job_id, department_id
  2  FROM employees
  3  WHERE employee_id = 110;

EMPLOYEE_ID HIRE_DATE JOB_ID     DEPARTMENT_ID
----------- --------- ---------- -------------
        110 28-SEP-97 FI_ACCOUNT           100

SQL> INSERT INTO job_history
  2  VALUES (110, '28-SEP-97', sysdate, 'FI_ACCOUNT', 100)
  3  ;

1 row created.

SQL> SELECT * FROM job_history;

EMPLOYEE_ID START_DAT END_DATE  JOB_ID     DEPARTMENT_ID
----------- --------- --------- ---------- -------------
        102 13-JAN-93 24-JUL-98 IT_PROG               60
        101 21-SEP-89 27-OCT-93 AC_ACCOUNT           110
        101 28-OCT-93 15-MAR-97 AC_MGR               110
        201 17-FEB-96 19-DEC-99 MK_REP                20
        114 24-MAR-98 31-DEC-99 ST_CLERK              50
        122 01-JAN-99 31-DEC-99 ST_CLERK              50
        200 17-SEP-87 17-JUN-93 AD_ASST               90
        176 24-MAR-98 31-DEC-98 SA_REP                80
        176 01-JAN-99 31-DEC-99 SA_MAN                80
        200 01-JUL-94 31-DEC-98 AC_ACCOUNT            90
        110 28-SEP-97 21-DEC-17 FI_ACCOUNT           100

11 rows selected.
```

ii.

```
SQL> UPDATE employees
  2  SET job_id = (SELECT job_id FROM jobs WHERE job_title = 'Accounting Manager'), salary = (SELECT (min_salary + max_salary)/2 FROM jobs WHERE job_title = 'Accou
nting Manager')
  3  WHERE employee_id = 110;

1 row updated.
```

iii.

```
SQL> SELECT employee_id, hire_date, salary, job_id, department_id
  2  FROM employees
  3  WHERE employee_id = 110;

EMPLOYEE_ID HIRE_DATE     SALARY JOB_ID     DEPARTMENT_ID
----------- --------- ---------- ---------- -------------
        110 28-SEP-97      12100 AC_MGR               100

SQL> SELECT *
  2  FROM job_history
  3  WHERE employee_id = 110;

EMPLOYEE_ID START_DAT END_DATE  JOB_ID     DEPARTMENT_ID
----------- --------- --------- ---------- -------------
        110 28-SEP-97 21-DEC-17 FI_ACCOUNT           100
```

While the problem does not ask us to update the hire_date column for employee_id = 110, from the above, it would make sense to update the hire_date in the employees table to reflect the system date used as the end_date in the previous statement, by adding “, hire_date = sysdate” to the end of the SET clause for question 7 part two in order to reflect the date employee_id 110 started the new position as an Accounting Manager.

8.

```
SQL> COLUMN job_titles FORMAT a100
SQL> SELECT DISTINCT(job_cat),
  2  LISTAGG(job_title, ', ') WITHIN GROUP (ORDER BY job_title) OVER (PARTITION BY job_cat) AS job_titles
  3  FROM
  4  (
  5     SELECT SUBSTR(job_id,1,2) AS job_cat, job_title
  6     FROM jobs
  7  )
  8  ORDER BY job_cat;

JOB_CAT  JOB_TITLES
-------- ----------------------------------------------------------------------------------------------------
AC       Accounting Manager, Public Accountant
AD       Administration Assistant, Administration Vice President, President
FI       Accountant, Finance Manager
HR       Human Resources Representative
IT       Programmer
MK       Marketing Manager, Marketing Representative
PR       Public Relations Representative
PU       Purchasing Clerk, Purchasing Manager
SA       Sales Manager, Sales Representative
SH       Shipping Clerk
ST       Stock Clerk, Stock Manager

11 rows selected.
```

I first did not include the DISTINCT function and tried to GROUP BY job_cat, but received the following error.

```
SQL> SELECT job_cat,
  2  LISTAGG(job_title, ', ') WITHIN GROUP (ORDER BY job_title) OVER (PARTITION BY job_cat) AS job_titles
  3  FROM
  4  (
  5     SELECT SUBSTR(job_id,1,2) AS job_cat, job_title
  6     FROM jobs
  7  )
  8  GROUP BY job_cat
  9  ORDER BY job_cat;
LISTAGG(job_title, ', ') WITHIN GROUP (ORDER BY job_title) OVER (PARTITION BY job_cat) AS job_titles
        *
ERROR at line 2:
ORA-00979: not a GROUP BY expression
```