--Assignment 2

--CST203 Nathan Balaniuk

--Q1

--Join

SELECT DISTINCT fName AS "First Name", lName AS "Last Name", phone AS "Phone Number"

FROM A3.EMPLOYEE E

JOIN A3.Department D ON E.EMPNO = D.MGREMPNO ORDER BY E.LNAME, E.fName;

--Subquery

SELECT fName, lName, Phone FROM A3.EMPLOYEE WHERE empNo IN (SELECT mgrEmpNo FROM A3.DEPARTMENT)

ORDER BY lName, fName;

--Q2

--SELECT D.deptNo AS "Department Number", D.deptName AS "Department", COUNT(\*) AS "Number of employees"

--FROM A3.DEPARTMENT D RIGHT JOIN A3.Employee E ON D.DEPTNO = D.DEPTNO GROUP BY D.deptNo, D.deptName;

SELECT NVL(to\_char(E.deptNo),'No Dept') AS "Department Number", NVL(D.deptName, '<not assigned>') AS "Department", COUNT(\*) AS "Number of employees"

FROM A3.EMPLOYEE E LEFT JOIN A3.DEPARTMENT D ON E.DEPTNO = D.DEPTNO GROUP BY E.DEPTNO, D.deptName;

SELECT DeptName, deptNo FROM A3.DEPARTMENT;

SELECT \* FROM A3.Employee WHERE DEPTNO =1004;

--Union

--NVL(D.deptName, '<not assigned>') AS "Department",

-- NVL(D.deptName, '<not assigned>')

--FIX THIS

--could use a subquery in each union, or a join in each

--not really efficient though

--or just subquery the whole thing / make this into a table and take from there

--just fo NVL for a

--No Join

SELECT NVL(to\_char(deptNo),'No Dept') AS "Department Number", NULL AS "Department", COUNT(\*) AS "Number of employees"

FROM A3.EMPLOYEE E

GROUP BY DEPTNO

UNION

SELECT to\_char(deptNo), MAX(NVL(deptName, '<not assigned>')), MAX(0) FROM A3.Department D

GROUP BY DeptNo;

--With join in a union

--this one actually works

SELECT NVL(to\_char(E.deptNo),'No Dept') AS "Department Number", NVL(D.deptName, '<not assigned>') AS "Department", COUNT(\*) AS "Number of employees"

FROM A3.EMPLOYEE E LEFT JOIN A3.DEPARTMENT D ON E.DEPTNO = D.DEPTNO GROUP BY E.DEPTNO, D.deptName

UNION

SELECT NVL(to\_char(deptNo),'No Dept') AS "Department Number", NVL(deptName, '<not assigned>') AS "Department", COUNT(\*) FROM A3.Department D

WHERE deptNo IS NULL

GROUP BY DeptNo, Deptname;

--Q3 This question was a bit ambiguous, so I did each person not assigned to a department first day for each project, and their first day on their first project

SELECT \* FROM A3.Employee WHERE DeptNo IS NULL;

--Earlierst work for each person per project

SELECT DISTINCT E.empNo AS "Employee Number", E.lName AS "Last Name",

E.fName AS "First Name", P.PROJNO AS "Project Number", P.DeptNo AS "Department Number"--,

--MIN(W.dateWorked) AS "First Time Worked on Project"

FROM A3.Employee E

JOIN A3.WorksOn W ON E.EmpNo = W.empNo

JOIN A3.Project P ON P.PROJNO = W.ProjNo

WHERE E.DeptNo IS NULL

GROUP BY E.EmpNo, E.fName, E.lName, P.PROJNO, P.DeptNo;

--Earliest work per person total - the date they started on their first project

SELECT DISTINCT E.empNo AS "Employee Number", E.lName AS "Last Name",

E.fName AS "First Name", P.PROJNO AS "Project Number", P.DeptNo AS "Department Number" --,

--MIN(W.dateWorked) AS "First Time Worked on Project"

FROM A3.Employee E

JOIN A3.WorksOn W ON E.EmpNo = W.empNo

JOIN A3.Project P ON P.PROJNO = W.ProjNo

WHERE (E.DeptNo IS NULL AND W.dateWorked = (SELECT MIN(dateWorked) FROM A3.WorksOn W WHERE W.empNo = E.empNo))

GROUP BY E.EmpNo, E.fName, E.lName,P.PROJNO, P.DeptNo;

SELECT E.empNo, E.deptNo FROM A3.Employee E FULL JOIN A3.Department D ON E.deptNo = D.deptNo WHERE E.deptNo IS NULL;

SELECT empNo, MIN(dateWorked) FROM A3.WorksOn WHERE empNo IN(SELECT E.empNo FROM A3.Employee E FULL JOIN A3.Department D ON E.deptNo = D.deptNo WHERE E.deptNo IS NULL) GROUP BY empNo;

--earliest work day for each employee with no department

SELECT E.empNo, E.lName, E.fName,

--Q4

--Hard code Kenya Greens EmpNo, it should never change

SELECT empNo FROM A3.EMPLOYEE WHERE lName='Green';

--2044

SELECT E.empNO AS "Employee Number",E.fName ||' '|| E.lName AS "Name", E.phone AS "Phone Number" FROM A3.Employee E

JOIN A3.DEPARTMENT D ON E.deptNo = D.deptNO

JOIN A3.WorksOn W ON E.empNo = W.EMPNO

WHERE (D.MGREMPNO = 2044 ) --hard Coded empNo for Kenya Greens

GROUP BY E.empNo, E.fName,E.lname,E.phone

HAVING SUM(W.HOURSWORKED) >= 10;