

Jake Magri

Problem 1

Find all department names (no duplicates) that have ever sold any item, excluding items of type N, R, or E. Make sure the output uses "Department Name" as the header for the output column.

Department Name
Books
Clothes
Equipment

```
SELECT
    DISTINCT DEP.DEPTNAME AS [Department Name]
FROM
    DEPARTMENT AS DEP, SALE AS S, ITEM AS I
WHERE
    DEP.DEPTNO = S.DEPTNO
AND
    S.ITEMNO = I.ITEMNO
AND
    I.ITEMTYPE NOT IN ('N', 'R', 'E')
;
```

Department Name
Books
Clothes
Equipment

Record: 1 of 3

Problem 2

Find all employees whose boss is either 'Alice' or 'Ned'. List their names, salaries, bosses' names and salaries, and the difference between the salaries (for each row). Use the column headers shown below.

Employee	Employee Salary	Boss	Boss Salary	Difference in salary
Ned	\$45,000.00	Alice	\$75,000.00	\$30,000.00
Andrew	\$25,000.00	Ned	\$45,000.00	\$20,000.00
Clare	\$22,000.00	Ned	\$45,000.00	\$23,000.00
Todd	\$38,000.00	Alice	\$75,000.00	\$37,000.00
Brier	\$43,000.00	Alice	\$75,000.00	\$32,000.00
Sophie	\$35,000.00	Alice	\$75,000.00	\$40,000.00

```

SELECT
    E.EMPNAME AS Employee, E.EMPSALARY AS [Employee Salary],
    B.EMPNAME AS Boss, B.EMPSALARY AS [Boss Salary],
    B.EMPSALARY - E.EMPSALARY AS [Difference in salary]
FROM
    EMPLOYEE AS E, EMPLOYEE AS B
WHERE
    E.BOSSNO = B.EMPNO AND B.EMPNAME IN ('Alice', 'Ned')
;

```

Employee	Employee Salary	Boss	Boss Salary	Difference in salary
Ned	\$45,000.00	Alice	\$75,000.00	\$30,000.00
Andrew	\$25,000.00	Ned	\$45,000.00	\$20,000.00
Clare	\$22,000.00	Ned	\$45,000.00	\$23,000.00
Todd	\$38,000.00	Alice	\$75,000.00	\$37,000.00
Brier	\$43,000.00	Alice	\$75,000.00	\$32,000.00
Sophie	\$35,000.00	Alice	\$75,000.00	\$40,000.00

Problem 3

With a single query, create a new table named NEWEMPLOYEE containing the Number, Name, and Salary of every employee in the EMPLOYEE table whose name ends with "e". (Assume there can be many different names that can end with "e").

EmpNo	EmpName	EmpSalary
1	Alice	\$75,000.00
4	Clare	\$22,000.00
9	Sophie	\$35,000.00
13	Maggie	\$16,000.00
18	Terence	\$32,000.00
*		

```

SELECT
    EMPNO, EMPNAME, EMPSALARY
INTO
    NEWEMPLOYEE
FROM
    EMPLOYEE
WHERE
    EMPNAME LIKE "*e"
;

```

EmpNo	EmpName	EmpSalary	DeptNo	BossNo
1	Alice	\$75,000.00	6	
10	Sanjay	\$15,000.00	8	3
11	Rita	\$15,000.00	2	4
12	Gigi	\$16,000.00	3	4
13	Maggie	\$16,000.00	3	4

Record: 1 of 18

EMPNO	EMPNAME	EMPSALARY
1	Alice	\$75,000.00
4	Clare	\$22,000.00
9	Sophie	\$35,000.00
13	Maggie	\$16,000.00
18	Terence	\$32,000.00

Record: 1 of 5

Problem 4

Return the aggregate (total) delivered quantity for all departments with total number of received deliveries (not delivered quantity!) greater than 7. Show the department name and the aggregate delivered quantity in your output. Use appropriate names for the columns. Order the results in descending by the number of received deliveries.

Department Name	Number of deliveries	Total quantity delivered
Navigation	16	207
Equipment	11	87
Recreation	10	65
Clothes	9	40

```

SELECT
    D.DEPTNAME AS [Department Name], COUNT(DEL.DELNO) AS [Number of
    deliveries], SUM(DEL.DELQTY) AS [Total quantity delivered]
FROM
    DEPARTMENT AS D, DELIVERY AS DEL
WHERE
    D.DEPTNO = DEL.DEPTNO
GROUP BY
    D.DEPTNAME
HAVING
    COUNT(DEL.DELNO) > 7
ORDER BY
    COUNT(DEL.DELNO) DESC
;

```

Query1

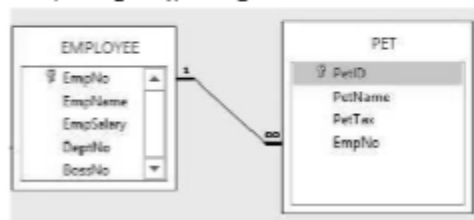
Department Name	Number of deliveries	Total quantity delivered
Navigation	16	207
Equipment	11	87
Recreation	10	65
Clothes	9	40

Problem 5

Create a new table named PET with the following structure :

PETID	AutoNumber	Must have a unique value; Primary Key
PETNAME	Text	Size 15; Must have a value
PETTAX	Number	Must allow decimal places
EMPNO	Text	Must have a value; Foreign key, linking to EMPLOYEE table

Field Name	Data Type
PetID	AutoNumber
PetName	Short Text
PetTax	Number
EmpNo	Short Text



```

CREATE TABLE PET (
    PETID AUTOINCREMENT PRIMARY KEY,
    PETNAME VARCHAR(15) NOT NULL,
    PETTAX DOUBLE,
    EMPNO VARCHAR(30) NOT NULL UNIQUE,
    FOREIGN KEY (EMPNO) REFERENCES EMPLOYEE(EMPNO)
);

```

PETID	PETNA	PETTA	EMPNO
1	bob	324.32	4
2	eric	234.34	3
3	fish	32.1	2

(New)

Records: 4 of 4

Query1

```

CREATE TABLE PET (
    PETID AUTOINCREMENT PRIMARY KEY,
    PETNAME VARCHAR(15) NOT NULL,
    PETTAX DOUBLE,
    EMPNO VARCHAR(30) NOT NULL UNIQUE,
    FOREIGN KEY (EMPNO) REFERENCES EMPLOYEE(EMPNO)
);

```

Field Name	Data Type
PETID	AutoNumber
PETNAME	Short Text
PETTA	Number
EMPNO	Short Text

