Q1. Consider the following database schema:

Neighbors(name1,name2,duration)
Colleagues(name1,name2,duration)

Write a Relational Algebra Plan for the SQL query below.

SELECT DISTINCT C1.name1, C2.name2
FROM Colleagues C1, Neighbors N, Colleagues C2
WHERE
C1.name2 = N.name1
AND N.name2 = C2.name1
AND C1.duration < 10
AND C2.duration < 10
AND N.duration > 100

-----------------------------------------------------------------------

Q2. Consider the following relational schema:

R(A,B)
S(C,D,E)
T(F, H, G)

Write a Relational Algebra Plan for the SQL query below.

SELECT R.B, S.E, sum(T.G)
FROM R, S, T
WHERE R.A = S.C
AND S.D = T.F
AND T.H > 55
GROUP BY R.B, S.E

-----------------------------------------------------------------------

Q3. Consider the following relational schema:

Users(uid, name)
Comment(uid, pid, score, txt)
Picture(pid, author, img)

(1) Write a Relational Algebra expression that is equivalent to the SQL query below:

SELECT distinct u.uid
FROM Users u, Picture x, Comment y
WHERE u.uid = x.author and x.pid = y.pid and y.score > 8
GROUP BY u.uid, x.pid
HAVING count(*) > 10
(2) Write a Relational Algebra expression that is equivalent to the SQL query below:

```
SELECT  x.pid  
FROM    picture x  
WHERE NOT EXISTS  
(SELECT *  
FROM comment y  
WHERE  x.pid = y.pid and y.score <5)
```

(3) Consider the Relational Algebra expression below:

```
\Pi_{uid}  
\sigma_{s > 20}  
\forall_{x.uid, sum(c) \rightarrow s}  
\forall_{x.uid=y.uid}  
\forall_{y.uid, count(\ast) \rightarrow c}  
\sigma_{y.score < 3}
```

Write an equivalent SQL query without using any subqueries.