Nested Queries

It is possible to nest sub-queries in queries. This can be done in the WHERE, HAVING, and FROM clauses. Let's look at one in the WHERE clause:

```sql
mysql> SELECT * FROM Students;
+-----+------------------+
| SID | SName            | NumCredits |
+-----+------------------+
| 1   | Elmer Fudd       | 48         |
| 2   | Roger Ramjet     | 20         |
| 3   | Alice Wonderland | 72         |
+-----+------------------+
3 rows in set (0.00 sec)

mysql> SELECT * from Students S
-> WHERE S.SID NOT IN
  -> (SELECT SID FROM Students WHERE SName LIKE 'Roger%');
+-----+------------------+
| SID | SName            | NumCredits |
+-----+------------------+
| 1   | Elmer Fudd       | 48         |
| 3   | Alice Wonderland | 72         |
+-----+------------------+
2 rows in set (0.02 sec)
```

In this example, we have used the subquery in the where clause to create a second relation that is used for set difference. You can also put one in the FROM clause.

```sql
mysql> SELECT * FROM
  -> (SELECT SName, E.ClassID, Grade
  -> FROM Students S, Enrolled E
  -> WHERE S.SID = E.SID) as ST;
+------------------+
| SName        | ClassID | Grade |
+------------------+
| Elmer Fudd    | CS314   | NULL  |
| Elmer Fudd    | CS575   | NULL  |
| Roger Ramjet  | CS575   | NULL  |
+------------------+
3 rows in set (0.00 sec)
```
And finally, you can put one in the HAVING clause.

```sql
mysql> SELECT Count(*) FROM Students S, Enrolled E
    -> WHERE (S.SID = E.SID)
    -> GROUP BY ClassID
    -> HAVING ClassID in
    -> (SELECT DISTINCT E1.ClassID FROM Enrolled E1
    ->     WHERE E1.ClassID LIKE 'CS5%');

+----------+
| Count(*) |
+----------+
|    2     |
+----------+
1 row in set (0.00 sec)
```