List major code with student count of at least 400 students with a minimum gpa of 3.5 (list the values in descending student count).

```sql
SELECT major, count(*)
FROM student
WHERE gpa >= 3.5
GROUP BY major
HAVING count(*) >= 400
ORDER BY count(*) DESC
```
List last name of employees who work for 'AQUARIUS' project and born after '1957-12-31'.

```
SELECT Lname
FROM Employee, Works_on, Project
WHERE Pname = 'AQUARIUS' AND
      Pnumber = Pno AND
      ESSN = SSN AND
      BDate > '1957-12-31'
```
Rule 1: Breakdown Conjunctive (AND) selection conditions and move down the tree as far as possible.

\[ \text{Employee} \times \text{Project} \]

\[ \text{Employee} \times \text{Project} \]

Rule 2: Reorder product operations such that the product associated with the most restrictive selection be performed first.

\[ \text{Employee} \times \text{Project} \]
Rule 3: Replace each pair of product and select operations with a join (sort-merge) operation.

Rule 4: Apply project operation in each branch of each join operation to reduce the width of the input tables to the join operation.

Final Optimized Query Tree
SQL Query for the optimized query tree:

```
SELECT Lname
FROM (SELECT SSN, Lname
      FROM Employee
      WHERE BDATE > '1957-12-31') as L
JOIN (SELECT ESSN
      FROM (SELECT ESSN, PNO
             FROM works_on) as RA
      JOIN (SELECT Pnumber
             FROM Project
             WHERE Pname = 'AQUARIUS') as RB
      ON RA.PNO = RB.Pnumber) as R
      ON L.SSN = R.ESSN
```