List the last name and department number of all male employees.
List the last name and department number of all male employees.
<table>
<thead>
<tr>
<th>Domain Relational Calculus (59)</th>
<th>SQL – underspecified (74)</th>
</tr>
</thead>
<tbody>
<tr>
<td>${&lt;l,n&gt;</td>
<td>\exists &lt;l,f.s,x,d&gt; \in \text{Employee} (\exists &lt;n,i,m&gt; \in \text{Department} (\land x = \textquote{M} \land d=i))}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tuple Relational Calculus (89)</th>
<th>SQL - aliased (82)</th>
</tr>
</thead>
<tbody>
<tr>
<td>${t</td>
<td>\exists e \in \text{Employee} (\exists d \in \text{Department} (t.lname = e.lname \land t.dname = d.dname \land e.sex = 'M' \land e.dno = d.dnumber))}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Relational Algebra [in RC/SQL style] (71)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$e \leftarrow \text{Employee}$</td>
<td></td>
</tr>
<tr>
<td>$d \leftarrow \text{Department}$</td>
<td></td>
</tr>
<tr>
<td>$t \leftarrow \Pi \text{lname, dname}$</td>
<td></td>
</tr>
<tr>
<td>$(\sigma e.sex='M' \land e.dno = d.dnumber (e \times d))$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Relational Algebra [in “native” (and underspecified) style] (64)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Pi \text{lname, dname} (\sigma \text{sex='M'} (\text{Employee} \bowtie dno = dnumber \text{Department}))$</td>
<td></td>
</tr>
</tbody>
</table>
List the last name and department number of all male employees without dependents.

SELECT e.lname, d.dname
FROM Employee e, Department d
WHERE e.sex = 'M' 
and e.dno = d.dnumber 
and e.ssn not in (SELECT essn
FROM Dependent)

Relational Algebra [in RC/SQL style] (109)

\[
e \leftarrow \text{Employee} \\
d \leftarrow \text{Department} \\
c \leftarrow \text{Dependent}
\]
\[
A \leftarrow \Pi \text{Iname, dname, ssn} (\sigma \text{sex='M'} \land \text{e.dno} = \text{d.dnumber} \ (e \times d)) \\
W \leftarrow \Pi \text{Iname, dname, ssn} A \bowtie a.\text{ssn} = c.\text{essn} C \\
T \leftarrow \Pi \text{Iname, dname} (A - W)
\]

Relational Algebra [in “native” (and underspecified) style] (102)

\[
A \leftarrow \Pi \text{Iname, dname} (\sigma \text{sex='M'} (\text{Employee} \bowtie \text{dno} = \text{dnumber} \text{Department}))
\]
List the last name and department number of all male employees without dependents.

Domain Relational Calculus (84)

\{<l,n>| \exists <l,f.s,x,d> \in Employee (\exists <n,i,m> \in Department (\forall x = 'M' \land d=i \\
\land \neg \exists <p,a,g> \in Dependent(s=p))))\}

Tuple Relational Calculus (119)

\{t| \exists e \in Employee (\exists d \in Department (t.lname = e.lname \\
\land t.dname = d.dname \\
\land e.sex = 'M' \land e.dno = d.dnumber \\
\land \neg \exists c \in Dependent(e.ssn=c.essn))))\}

Relational Algebra [in RC/SQL style] (109)

\begin{align*}
e & \leftarrow \text{Employee} \\
d & \leftarrow \text{Department} \\
c & \leftarrow \text{Dependent} \\
A & \leftarrow \Pi \text{Name}, \text{Name}, \text{ssn} (\sigma e.\text{sex}='M' \\
\land e.\text{dno} = d.\text{dnumber} (e \times d)) \\
W & \leftarrow \Pi \text{Name}, \text{Name}, \text{ssn} A \bowtie \\
\quad a.\text{ssn} = c.\text{essn} C \\
T & \leftarrow \Pi \text{Name}, \text{Name} (A - W)
\end{align*}

English (82)

SQL - aliased (127)

```
SELECT e.iname, d.dname
FROM Employee e, Department d
WHERE e.sex = 'M'
\land e.dno = d.dnumber
\land e.ssn not in (SELECT essn
FROM Dependent)
```

QBE (34)

```
EMPL | ssn | iname | sex | dno
--- | --- | --- | --- | ---
_ | _ | _l | M | _d

DEPT | dnumber | dname | mgrssn
--- | --- | --- | ---
_ | _n | _d

DEPN | essn | depname
--- | --- | ---
_ | _s

RESULT | iname | dname
--- | ---
P._l | P._n
```