Consider again the fact table 
\texttt{sales(city, product, date, amt)}

and the same dimension tables

Consider a transaction that inserts each time one tuple in sales providing its cityID, which already exists in City, its productID, which already exists in Product its date and amt.

Assume that the table \texttt{pd(product, date, sumamt)} is precomputed

Describe the INSERT/DELETE/UPDATE commands that this transaction must issue in order to keep \texttt{pd} up-to-date

The following lattice of candidates assumes fact table is \texttt{sales(city, product, date, amt)}

How many nodes are there in a lattice of candidates for \texttt{sales(city, product, date, paymentType, amt)}

Recall the midterm problem: The same jsp accomplished with different queries

\texttt{users = userStatement.executeQuery("SELECT ID, name FROM users") ;}

\texttt{while (users.next()) { …}

\texttt{purchases = purchasesStatement.executeQuery("SELECT COUNT(*) AS cnt FROM purchases WHERE user = " + users.getInt("ID") ;}

\texttt{…} }

\texttt{Version 1 (many queries) }

\texttt{users_purchases = Statement.executeQuery("SELECT name, COUNT(*) as cnt FROM users LEFT OUTER JOIN purchases ON (users.id = purchases.user) GROUP BY name, user.id;") ;}

\texttt{Version 2 (single query) }

\texttt{Assume each user has purchases.}

Guess which indices will work for Version 1? Why? For Version 2? Why?