Exercises for the lecture on October 24

This week’s exercises are theoretical, and consist of problems concerning normalization.

1. **Problem 2 from exam in Introduction to Databases, June 2006**

Consider a relation with the schema \texttt{Sales(reseller,manufacturer,product,amount)}.

The following is a valid instance of \texttt{Sales}:

<table>
<thead>
<tr>
<th>reseller</th>
<th>manufacturer</th>
<th>product</th>
<th>amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Silman</td>
<td>SoftFloor AG</td>
<td>Velour</td>
<td>101000</td>
</tr>
<tr>
<td>Bjarnes Tæpper</td>
<td>Bogetæpper</td>
<td>Berber</td>
<td>207000</td>
</tr>
<tr>
<td>Top Tæpper</td>
<td>Bogetæpper</td>
<td>Kashmir</td>
<td>77000</td>
</tr>
<tr>
<td>Silman</td>
<td>SoftFloor AG</td>
<td>Berber</td>
<td>72000</td>
</tr>
<tr>
<td>Bjarnes Tæpper</td>
<td>Bogetæpper</td>
<td>Valnød</td>
<td>17000</td>
</tr>
</tbody>
</table>

\textbf{a)} Which of the following potential FDs do \textit{not} hold, based on the instance above?

1. \texttt{amount → product}
2. \texttt{amount → product reseller}
3. \texttt{product → manufacturer}
4. \texttt{manufacturer → product}
5. \texttt{reseller product → amount}

\textbf{b)} State a functional dependency (FD), that ensures that \texttt{Sales} can be split as in the above example without loss of information. The FD should in other words ensure that the SQL statement:

\begin{align*}
(\text{SELECT reseller, manufacturer FROM Sales}) \text{ NATURAL JOIN} \\
(\text{SELECT reseller, product, amount FROM Sales})
\end{align*}

always returns a relation that is identical to \texttt{Sales}. Further, express in words what the FD says.

\textbf{c)} Give an instance of \texttt{Sales}, where the splitting fails, i.e., where the SQL statement of question \textbf{b)} does \textit{not} return the same instance.
3. Problem 2 from exam in Databasesystemer, June 2005

We consider the relation Doctor, containing information on medical doctors and the diseases they treat:

Doctor(id, address, postcode, city, speciality, disease)

Each doctor has a unique id, and furthermore precisely one address. For every post code there is exactly one city. A doctor can have more than one speciality, but in every city there is at most one doctor with a given speciality. A disease belongs to exactly one speciality, but a speciality may cover many diseases.

(a) State all keys of Doctor. Explain any assumptions you make on the data.
(b) Perform normalization to 3NF, stating the resulting relation schema, and what functional dependencies have been used.

4. Consider the following relation schemas, aimed at registering data on resellers of products in different cities:

Reseller(id, address, city, product_id)

A reseller can have several products, but address and city are functionally dependent on id. That is, id and product_id constitute a candidate key. We assume that each product has a reseller with exclusive rights in each city, i.e.,

city, product_id → id.

In particular, this means that city and product_id also form a candidate key. Perform normalization to 3NF and BCNF, respectively. What are the advantages and disadvantages of the two normal forms in this case?