**ER and Conceptual Modeling**
- isA relationships (independent dimension total/partial and disjunct/non-disjunct)
- weak entities
- (min , max)-notation against 1:n – notation
- ternary relationships

**Questions and Hints:**
- What are the steps in a database lifecycle?
- What are the components of a model?
- What conceptual models do you know?

**Relational Databases [from Exam 03/04]**
Consider the following ER-diagram that shows the conceptual model of an information system for the music industry.

```
artist
  name
  artid
  contact
  email
  telephone

band
  bid
  name
  leader

album
  aid
  title
  year

solo album
  a_records

band album
  b_records

record label
  address
  name
  zip
  city
  street

artist
  part
  a
  n

band
  part
  b
  m

a) Map this diagram to a relational database schema. Use the formal notation presented in the lecture and tutorials including intra- and interrelational dependencies. (5 points)

b) Formulate the following query in tuple relational calculus:
List all bandleaders, who publish a solo album at a record label that is located at Munich. (3 points)

c) Describe the following query in SQL:
List the names of record labels and the names of bands together with the number of albums that the band has published at this record label before 1980. Only those bands should be considered, that are not lead by an artist who has published a solo album. (4 points)

d) Give an expression in Relational Algebra that calculates the result of the following query:
List all names of bands that have published at the record label ‘Funky records’ and consist only of artists who have not published a solo album. (3 points)

**Questions and Hints:**
- What is the definition of a tuple, a relation schema and a relational database schema?
- What are intra-relational dependencies? What are inter-relational dependencies?
- What are the operations of the relational algebra? How can they be translated into SQL?
- What is the relation concerning expressivity between
  - Relational Algebra,
  - Tuple Relational Calculus and
  - Domain Relational Calculus?
- How are tables, views and triggers defined in SQL?
- What are aggregate functions? How are they used in SQL?
**Algorithmic Schema Design**
Consider a relation \( r = (U,F) \) with the attribute set \( U = \{ A, B, C, D, E, F, G, H, I, J \} \) and the set of functional dependencies \( F = \{ AB \rightarrow C, A \rightarrow DE, B \rightarrow F, B \rightarrow G, F \rightarrow G, D \rightarrow HIJ \} \).

a) Check for the following functional dependencies whether they hold or not: \( AF \rightarrow BI \), \( AB \rightarrow DJ \)
b) What are the key(s) of \( r \)?
c) In which normal form is \( r \)?
d) Design a relational database schema using the decomposition algorithm.
e) Design a relational database schema using the synthesis algorithm.

**Questions and Hints:**
- What is a (candidate) key? What is a primary key? What is a (non-)key attribute?
- What are the Armstrong axioms? What are they used for?
- When are two FD sets equivalent?
- What are normal forms? Define them! What is their usage?
- Is there a “best” normal form? Why not?
- How can you check for dependency preservation?
- How can you check for lossless join property?
- Which of the above properties is guaranteed by (a) synthesis and (b) decomposition algorithm? What normal forms are guaranteed?

**Object Oriented Databases**
Consider the following ER-diagram that models an information system around a car dealer and his customers. Car dealers can decide to join their forces in a purchasing association, i.e. a group of dealers that purchase cars from the manufacturer as a group.

Transfer the above ER-diagram into an UML structure diagram.

a) Explain briefly the difference between the following types of object relationships.
   Give for each of these types two classes of your structure diagram, that are in a relationship which could be reasonably modelled by this type.
   - association
   - aggregation
   - composition

b) Do not consider the bank account and customer entities for the following tasks:
   a. Map the ER diagram to ODL classes (including definitions of extents as entry points).
   b. Formulate the following information demands as OQL queries against your database:
      i. All car dealers who sell leased Mercedes cars together with the number of purchasing associations they belong to.
      ii. All car dealers with the average price of their purchased cars and the average leasing rate of the leased cars.
      iii. All purchasing associations with a list of their members that sell BMW cars.

**Questions and Hints:**
- What kind of UML diagrams do you know? When are they used?
- Explain briefly three advantages of object-oriented database systems in comparison with relational database systems
- Be aware of different syntax for
  1. ODMG: ODL/OQL
  2. Object-Relational Databases according to SQL-99/2003
  3. Oracles Implementation
XML and Semistructured Databases
Consider the XML schema presented below.

a) Give an example XML document that is valid according to the XML schema.

b) Transform the XML schema into an equivalent DTD (with library as the root element) as far as possible. Which constraints cannot be expressed in a DTD?

c) Formulate an XQuery against a valid document of this type that lists for each author the books he/she has written and were published after 2002 and the number of journals he/she is an editor of in the following output format:

```
<result>
  <author name="...">
    <book title="..." publicationYear="..."/>
    <book title="..." publicationYear="..."/>
    ...
    <journalEditor noJournals="..">
      <author name="....."> ...
    </author>
    ...
  </author>
  ...
</result>
```

d) Consider the two XSLT scripts presented below. They are intended to construct an XML document containing all books with their title and publication year in the following format:

```
<booklist>
  <book>
    <title>...</title>
    <published>...</published>
  </book>
</booklist>
```

Decide for each XSLT script whether it fulfills these requirements. Identify and correct all mistakes you find.

Questions and Hints:
- What is semi-structured data?
- What are differences between the Relational and the Semi-Structured Data Model? What are the advantages/disadvantages? When is which usually used?
- What are differences between semi-structured data models (such as OEM) and XML?
- What is RDF? What is the purpose of RDF?

XML Schema
```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified" attributeFormDefault="unqualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="library">
    <xs:complexType>
      <xs:sequence>
        <xs:choice maxOccurs="unbounded">
          <xs:element name="book">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="title" type="xs:string"/>
                <xs:element name="author" type="xs:string" maxOccurs="3"/>
                <xs:element name="publicationYear" type="xs:gYear"/>
              </xs:sequence>
              <xs:attribute name="isbn" type="xs:string" use="required"/>
            </xs:complexType>
          </xs:element>
          <xs:element name="journal">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="title" type="xs:string"/>
                <xs:element name="editor" type="xs:string" maxOccurs="5"/>
                <xs:element name="firstIssue" type="xs:positiveInteger"/>
                <xs:element name="lastIssue" type="xs:positiveInteger" minOccurs="0"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:choice>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```
XSLT scripts

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:fo="http://www.w3.org/1999/XSL/Format">
    <xsl:template match="/library">
        <booklist>
            <xsl:apply-templates/>
        </booklist>
    </xsl:template>

    <xsl:template match="/library/book">
        <book>
            <xsl:apply-templates/>
        </book>
    </xsl:template>

    <xsl:template match="title">
        <xsl:value-of select="."/>
    </xsl:template>

    <xsl:template match="publicationYear">
        <published>
            <xsl:value-of select="."/>
        </published>
    </xsl:template>
</xsl:stylesheet>
```