Fachhochschule Kiel
University of Applied Sciences

Data Modeling

Data Modeling

Fachhochschule Kiel
University of Applied Sciences

Data modeling refers to procedures for the formal representation of objects relevant in a defined context by means of their attributes and relationships. The aim is to uniquely define and specify the objects to be managed in an information system. The results are data models that form the basis for implementation in a database system.

- System requirements and data analysis
- Design of an Entity-Relationship Model
- Translation of the Entity-Relationship Model into tables
- Normalization of the tables
- Relational data model

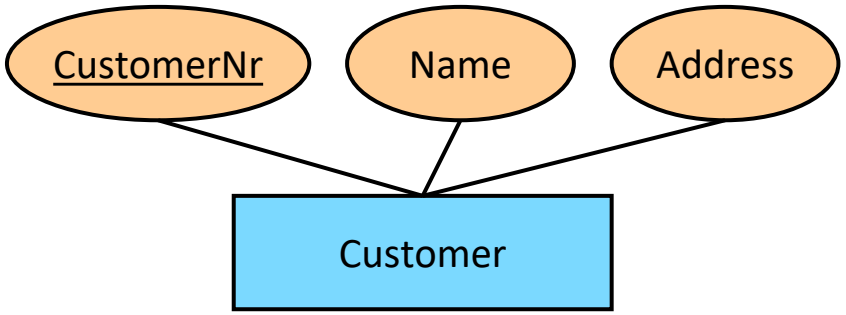
Entity-Relationship Model?

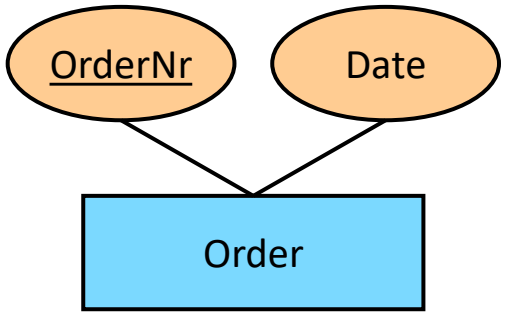
Fachhochschule Kiel
University of Applied Sciences

Entity-Relationship Model (ER-Model)

Fachhochschule Kiel
University of Applied Sciences

- **Entity:** An entity is an individual, uniquely identifiable object that can be characterized by properties.
- **Attribute:** An attribute consists of a name and a data type. Attributes are used to model properties.
- **Entity type:** An entity type is a schema with a unique name. It is used to group entities. All attributes of which the entities have concrete values are grouped together in the schema.

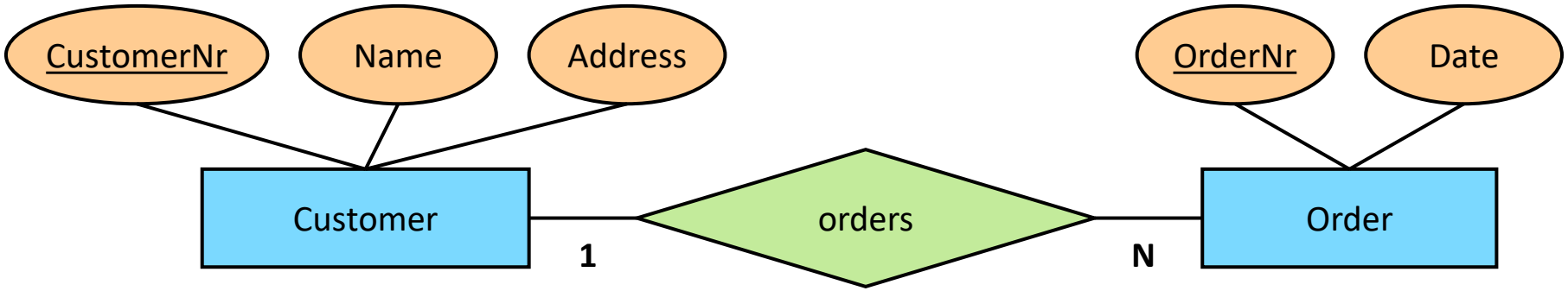




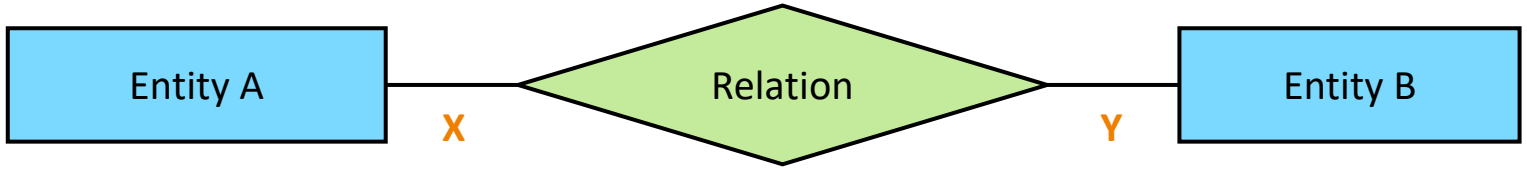
Relations and Cardinalities

Fachhochschule Kiel
University of Applied Sciences

- **Relation:** A relation represents a relationship between two entity types.
- **Cardinality:** A relation is defined by an **X:Y** combination, where X and Y have different cardinalities:
  - **1** a relationship to exactly one entity
  - **C** a relationship to no or one entity
  - **N** a relationship to one or more entities
  - **NC** a relationship to no, one or more entities



Examples of possible pairs of cardinalities



Entity A	Entity B	X	Y	Relation
Husband	Wife	1	1	married
Man	Wife	1	C	married
Mother	Child	1	N	born
Woman	Child	1	NC	born
Employee	Handy	C	C	have
Mentor	Artist	C	N	supports
Lake	River	C	NC	flows
Student	Lecture	M	N	participates
Article	Order	M	NC	appears
Article	Stock	MC	NC	stores

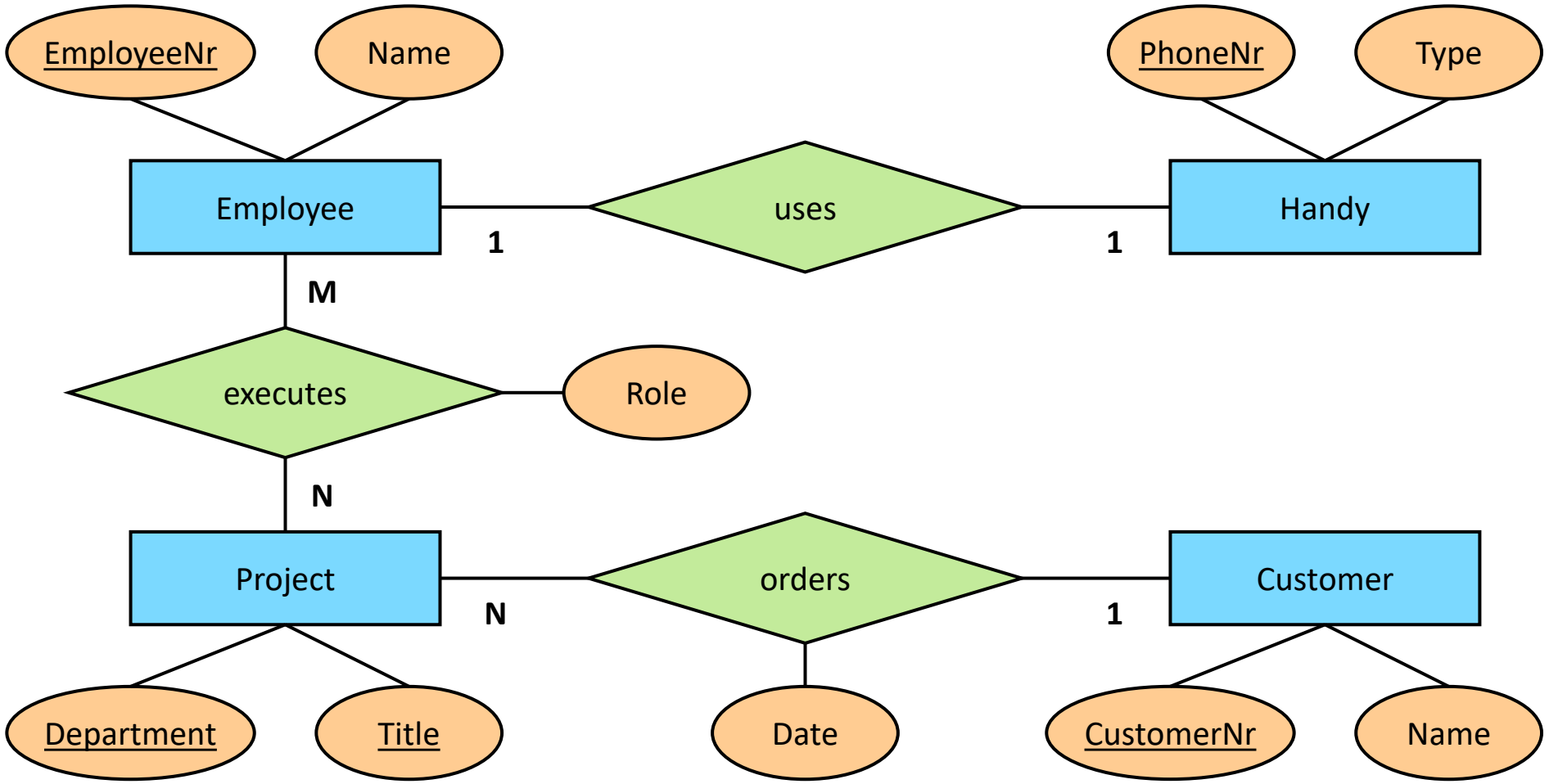
Example: Entity-Relationship Model

Fachhochschule Kiel
University of Applied Sciences

- Customers, projects, employees and their mobile phones are to be managed in a database.
- A customer has a name and is uniquely identified by a customer number.
- A project is uniquely identified by department and project title.
- An employee has a name and is uniquely identified by an employee number.
- A handy has a type and is uniquely identified by the phone number.
- An employee uses exactly one handy. An employee executes one or more projects in different roles.
- A project is ordered by exactly one customer on one date. A project can be executed by one or more employees in different roles.
- A customer can order one or more projects on different dates.
- A handy is used by exactly one employee.

Solution: Entity-Relationship Model

Fachhochschule Kiel
University of Applied Sciences



Translation of ER models into tables?

Fachhochschule Kiel
University of Applied Sciences

Tables

Fachhochschule Kiel
University of Applied Sciences

- **Data type:** A data type is a set of values. Elementary data types are text, number, date, and Boolean.
- **Data types for tables:** In tables, the elementary data types text, number, date, and Boolean can be used, whereby the value NULL is added to each data type.
- **Table:** A table has a name, a set of attributes, each attribute being a subset of a data type for tables, and a content that is a subset of the cross product of the attributes.
- **Primary Key:** A primary key is the set of attributes with which each data record in a table can be uniquely identified.

Customer		
CustomerNr	**Name**	**Address**
1	Michael	C12-2.15
2	Till	C12-2.16

Order	
OrderNr	**Date**
4711	01.03.2020
4712	16.03.2020

Translation of ER models into tables

Fachhochschule Kiel
University of Applied Sciences

- Targets:
  1. When filling tables with data, redundant data should be avoided.
  2. If it does not make sense from a practical point of view, no NULL values should be required to fill the tables.
  3. In consideration of 1. and 2., a minimum number of tables should be created.
- Steps:
  1. Translation of entity types
  2. Translation of 1:1 relations
  3. Translation of 1:N relations
  4. Translation of M:N relations
- The result of the translation is a set of tables that represents a relational data model for the Entity-Relationship model.

Step 1: Translation of entity types

Fachhochschule Kiel
University of Applied Sciences

- Each entity type is translated into a table, where the attributes of the entity type become the attributes of the table.

Employee	
EmployeeNr	**Name**
1	Michael
2	Till
3	Jens

Handy	
PhoneNr	**Type**
0171	Apple
0172	Samsung
0173	Apple

Project	
Department	**Title**
FBIE	DS-DM
FBIE	DS-EM
FBW	DS-EM

Customer	
CustomerNr	**Name**
4711	FH
4712	TUHH

Step 2: Translation of 1:1 relations

Fachhochschule Kiel
University of Applied Sciences

- 1:1 relations describe a close connection between entity types, as there is pairing between the respective entities. This information is combined in one table.

Employee			
EmployeeNr	**Name**	**PhoneNr**	**Type**
1	Michael	0171	Apple
2	Till	0172	Samsung
3	Jens	0173	Apple

Project	
Department	**Title**
FBIE	DS-DM
FBIE	DS-EM
FBW	DS-EM

Customer	
CustomerNr	**Name**
4711	FH
4712	TUHH

Step 3: Translation of 1:N relations

Fachhochschule Kiel
University of Applied Sciences

- For 1:N relations, the identifying attributes of the one or more entities are included in the table as foreign keys, as are the attributes of the relation.

Employee			
EmployeeNr	**Name**	**PhoneNr**	**Type**
1	Michael	0171	Apple
2	Till	0172	Samsung
3	Jens	0173	Apple

Project			
Department	**Title**	**CustomerNr**	**Date**
FBIE	DS-DM	4711	16.03.2020
FBIE	DS-EM	4711	01.03.2020
FBW	DS-EM	4712	01.03.2020

Customer	
CustomerNr	**Name**
4711	FH
4712	TUHH

Step 4: Translation of M:N relations

Fachhochschule Kiel
University of Applied Sciences

- For M:N relations, a new table is created for the identifying attributes of both entities and for the attributes of the relation.

Employee			
EmployeeNr	**Name**	**PhoneNr**	**Type**
1	Michael	0171	Apple
2	Till	0172	Samsung
3	Jens	0173	Apple

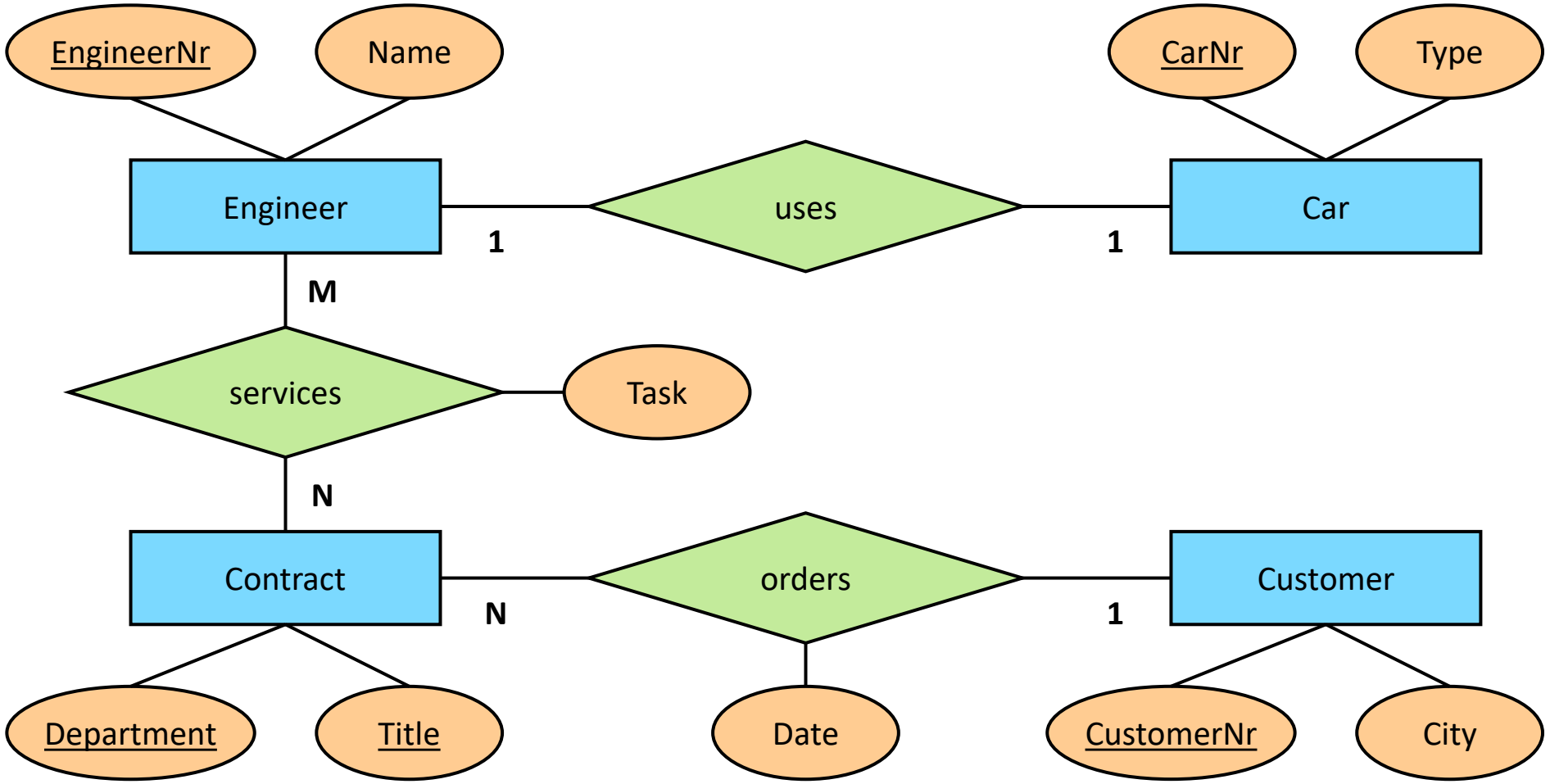
Execution			
EmployeeNr	**Department**	**Title**	**Role**
1	FBIE	DS-DM	Lecturer
1	FBIE	DS-EM	Lecturer
2	FBW	DS-EM	Lecturer
3	FBW	DS-EM	Supervisor

Project			
Department	**Title**	**CustomerNr**	**Date**
FBIE	DS-DM	4711	16.03.2020
FBIE	DS-EM	4711	01.03.2020
FBW	DS-EM	4712	01.03.2020

Customer	
CustomerNr	**Name**
4711	FH
4712	TUHH

Example: Translation of ER-Model into tables

Fachhochschule Kiel
University of Applied Sciences



Solution: Translation of ER-Model into tables

Fachhochschule Kiel
University of Applied Sciences

Engineer			
EngineerNr	**Name**	**CarNr**	**Type**
1	Michael	123	Audi
2	Till	456	BMW
3	Jens	789	Mercedes

Service			
EngineerNr	**Department**	**Title**	**Task**
1	Service	DS-CC	Design
1	Service	DS-BDT	Design
2	Training	DS-BDT	Workshop
3	Training	DS-BDT	Workshop

Contract			
Department	**Title**	**CustomerNr**	**Date**
Service	DS-CC	4711	16.03.2020
Service	DS-BDT	4711	01.03.2020
Training	DS-BDT	4712	01.03.2020

Customer	
CustomerNr	**City**
4711	Kiel
4712	Hamburg

Normalization of tables?

Fachhochschule Kiel
University of Applied Sciences

Normalization of tables

Fachhochschule Kiel
University of Applied Sciences

- **Normalization:** Normalization of a relational data schema is the division of attributes into several tables so that a form is created that no longer contains redundancies.
- **Functional Dependency:** A set of attributes B of a table is functionally dependent on a set of attributes A of the table, if there can only be a maximum of one concrete assignment of the attributes from B for each concrete assignment of the attributes from A.
- **Key Candidate:** If the set M of a table that contains all attributes of the table is fully functionally dependent on a set of attributes A, then A is a key candidate of the table.

Collaboration				
ENr	**Employee**	**FNr**	**Faculty**	**Projects**
1	Michael	1	FBIE	{(1,Cloud),(2,Data)}
2	Till	2	FBW	{(2,Data),(3,Tools)}
3	Jens	1	FBIE	{(1,Cloud),(3,Tools)}

First Normal Form

Fachhochschule Kiel
University of Applied Sciences

- A table is in first normal form if the value ranges of the attributes are atomic, i.e. each attribute only receives values from an unstructured value range. This means that no sets, enumerations, or repeating groups may occur in the individual attributes.

Collaboration					
ENr	**Employee**	**FNr**	**Faculty**	**PNr**	**Project**
1	Michael	1	FBIE	1	Cloud
1	Michael	1	FBIE	2	Data
2	Till	2	FBW	2	Data
2	Till	2	FBW	3	Tools
3	Jens	1	FBIE	1	Cloud
3	Jens	1	FBIE	3	Tools

Second Normal Form

Fachhochschule Kiel
University of Applied Sciences

- A table in first normal form is in second normal form if each non-empty subset of non-key attributes is fully functionally dependent on each key candidate.

Collaboration	
ENr	**PNr**
1	1
1	2
2	2
2	3
3	1
3	3

Employee			
ENr	**Employee**	**FNr**	**Faculty**
1	Michael	1	FBIE
2	Till	2	FBW
3	Jens	1	FBIE

Project	
PNr	**Project**
1	Cloud
2	Data
3	Tools

Third Normal Form

Fachhochschule Kiel
University of Applied Sciences

- A table in second normal form is in third normal form if no non-key attribute is transitively dependent on any key.

Collaboration	
ENr	**PNr**
1	1
1	2
2	2
2	3
3	1
3	3

Employee		
ENr	**Employee**	**FNr**
1	Michael	1
2	Till	2
3	Jens	1

Faculty	
FNr	**Faculty**
1	FBIE
2	FBW

Project	
PNr	**Project**
1	Cloud
2	Data
3	Tools

Example: Normalization of tables

Fachhochschule Kiel
University of Applied Sciences

Assignments				
ENr	**Engineer**	**LNr**	**Level**	**Contracts**
1	Michael	1	Senior	{(1,Microsoft),(2,Google)}
2	Till	2	Junior	{(2,Google),(3,Amazon)}
3	Jens	1	Senior	{(1,Microsoft),(3,Amazon)}

Solution: Normalization of tables

Fachhochschule Kiel
University of Applied Sciences

Collaboration	
ENr	**CNr**
1	1
1	2
2	2
2	3
3	1
3	3

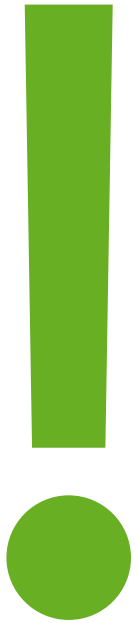
Engineer		
ENr	**Engineer**	**LNr**
1	Michael	1
2	Till	2
3	Jens	1

Level	
LNr	**Level**
1	Senior
2	Junior

Contract	
CNr	**Contract**
1	Microsoft
2	Google
3	Amazon

Exercise!

Fachhochschule Kiel
University of Applied Sciences



Exercise

Fachhochschule Kiel
University of Applied Sciences

- Define the term data modeling and explain the steps from data analysis to creating a relational data model.
- Name and describe the components from which an Entity-Relationship Model is created.
- Name the different cardinalities for relations and give an example for each one.
- What is the aim of translating an ER model into tables? In which steps does the translation take place?

Exercise

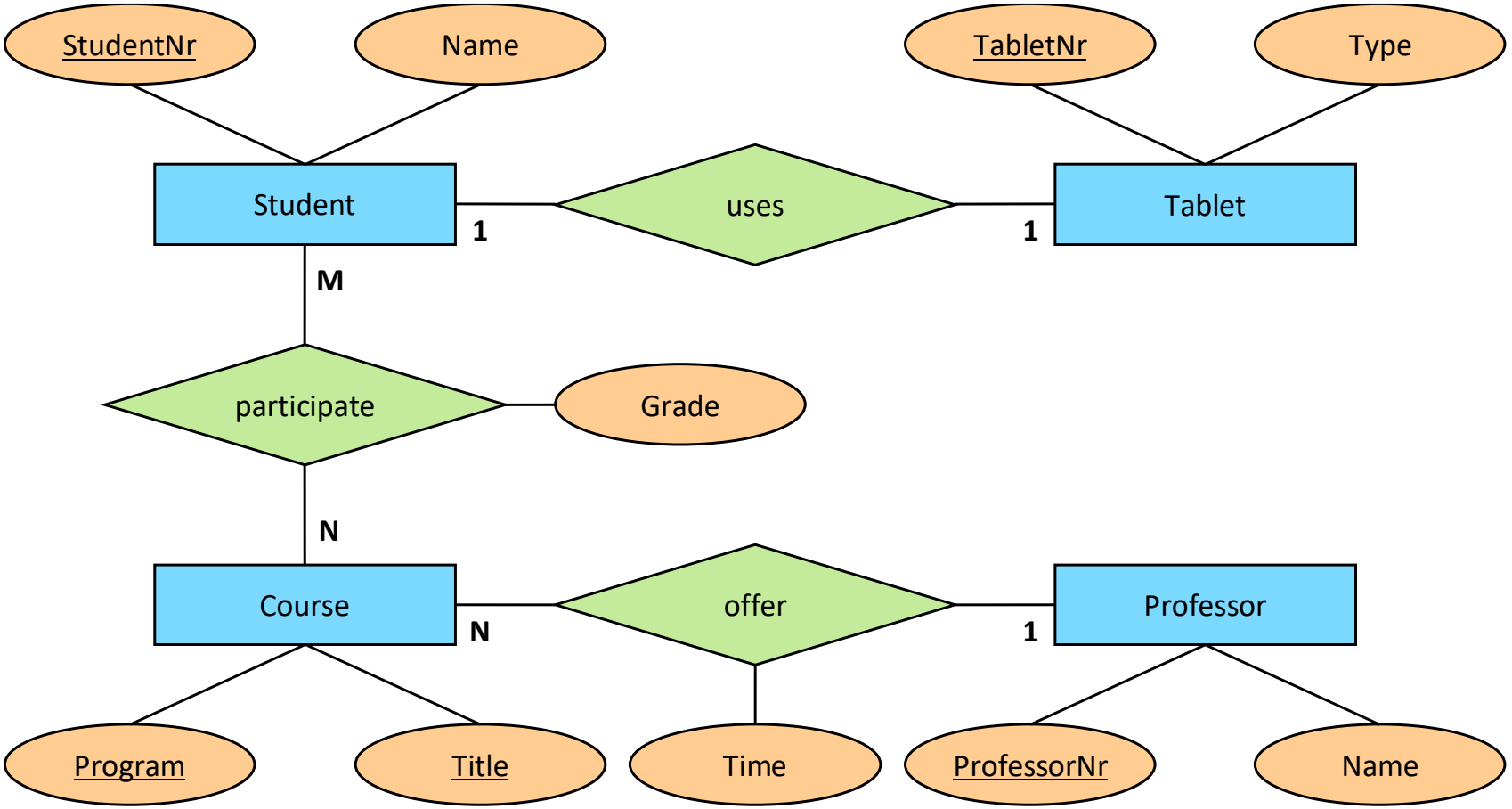
Fachhochschule Kiel
University of Applied Sciences

- Create an Entity-Relationship model from the following data analysis:
  - Professors, lectures, students and their tablets are to be managed in a database.
  - A professor has a name and is uniquely identified by a professor number.
  - A lecture is uniquely identified by its study program and module name.
  - A student has a name and is uniquely identified by a student number.
  - A tablet has a type and is uniquely identified by a device ID.
  - A student uses exactly one tablet. A student attends one or more lectures with different grades.
  - A lecture is held by exactly one professor at one time. Several students with different grades can take part in one lecture.
  - A professor can give several lectures at different times.
  - A tablet is used by exactly one student.

Exercise

Fachhochschule Kiel
University of Applied Sciences

- Translate the following Entity-Relationship model into tables:



Exercise

Fachhochschule Kiel
University of Applied Sciences

- What is the aim of normalizing tables?
- What does the primary key of a table mean?
- Which normal forms exist and how are they built on each other?
- Normalize the following table:

Booking				
SNr	**Student**	**PNr**	**Program**	**Contracts**
1	A	1	MADS	{(1,Cloud),(2,Data)}
2	B	2	ERASMUS	{(2,Data),(3,Tools)}
3	C	1	MADS	{(1,Cloud),(3,Tools)}

Exercise

Fachhochschule Kiel
University of Applied Sciences

- Think of a set of data objects that are related to each other and create the associated Entity-Relationship model based on the data analysis.
- Translate the created Entity-Relationship model into tables and normalize all tables.