LEC 01

Database – where you actually store the information

* Store the data in tabular form (tables)

Database management system – helps you manage data (insert, update, delete)

* Help you structure the DB

Database application – the program we code used to connect to the DBMS

We use different data models to structure the data

Structured data – we use relational model

Semi-structured data – we use XML

OOP model for object-oriented object modelling

Database design

Conceptual design – created using ER Diagrams (just illustrates the concept)

Logical design – more detailed than a conceptual design (this is what we implement)

Relations – tables that do not contain any duplicates or incorrect data

Data definition language (DDL) – deals with tables, use to define the data of a table; let’s you create, alter and delete a table.

Data manipulation language (DML) – Allows us to manipulate data by inserting, update, select and deleting data.

# Conceptual Entity-Relationship Modelling

1. Identify the entities involved (people [students, staff], objects[laptops, routers], places[lecture halls, offices], events[lectures, meetings], etc.)
2. Identify relationships (students enrolled in modules, lecturers conduct lectures, etc.)
3. Identify attributes – properties of an entity (student’s age, name, ID, etc.)
4. Identify primary keys
5. Consider enhanced modelling concepts
6. Check and remove redundancies

**Note: Do not leave out the simple-present verb and arrow in coursework**

Multiplicity is a row in a table.

Multiplicity tells use the number of possible occurrences of an entity related to a single occurrence of the other entity

Participation = 0 🡪 optional participation

Participation = 1 🡪 compulsory participation

Assumptions can be made when drawing up an ER diagram, but these assumptions must be clearly stated

Relationships can be categorized based on their cardinality (max value):

* 1-to-1
* 1-to-many
* Many-to-many

Occurrence is a row

Simple attributes – only one value

Composite attribute – composed of multiple values

Single-value attribute – holds a single value for each occurrence

Multiple-value attribute – holds multiple values

Derived attribute – represents a value that can be derived from the value of another

# Conceptual key

A key is a component whose value is unique for each occurrence in the table (NIC, email).

Searching using a key value will return only one record/occurrence

**Superkey**

**Candidate key**

Potential primary keys

**Primary key**

Selected potential primary key (duh!)

**Alternate key**

The remaining candidate keys that were not selected (aw sad)

A primary key that is not selected as THE primary key (should not be null meaning they must contain a value)

**Note: only logical ERDs have foreign keys, no conceptual ERDs.**

[**https://www.linkedin.com/in/zeinabshabbir/**](https://www.linkedin.com/in/zeinabshabbir/)

TUT 01

Find out file-based system vs DBS

# Exercise 01

**1.1**

A computer lab must be assigned at least one computer

A computer lab can be assigned many computers

A computer does not have to be assigned to a computer lab

A computer can be assigned to only one computer lab

**1.2**

An order must include at least one product

An order can have many products

A product does not have to be included in an order

A product can be included in many orders

**1.3**

A laptop does not have to be allocated to any staff

A laptop can be allocated to one staff member at most

A staff member must be allocated at least one laptop

A staff member can be allocated three laptops at most

**1.4**

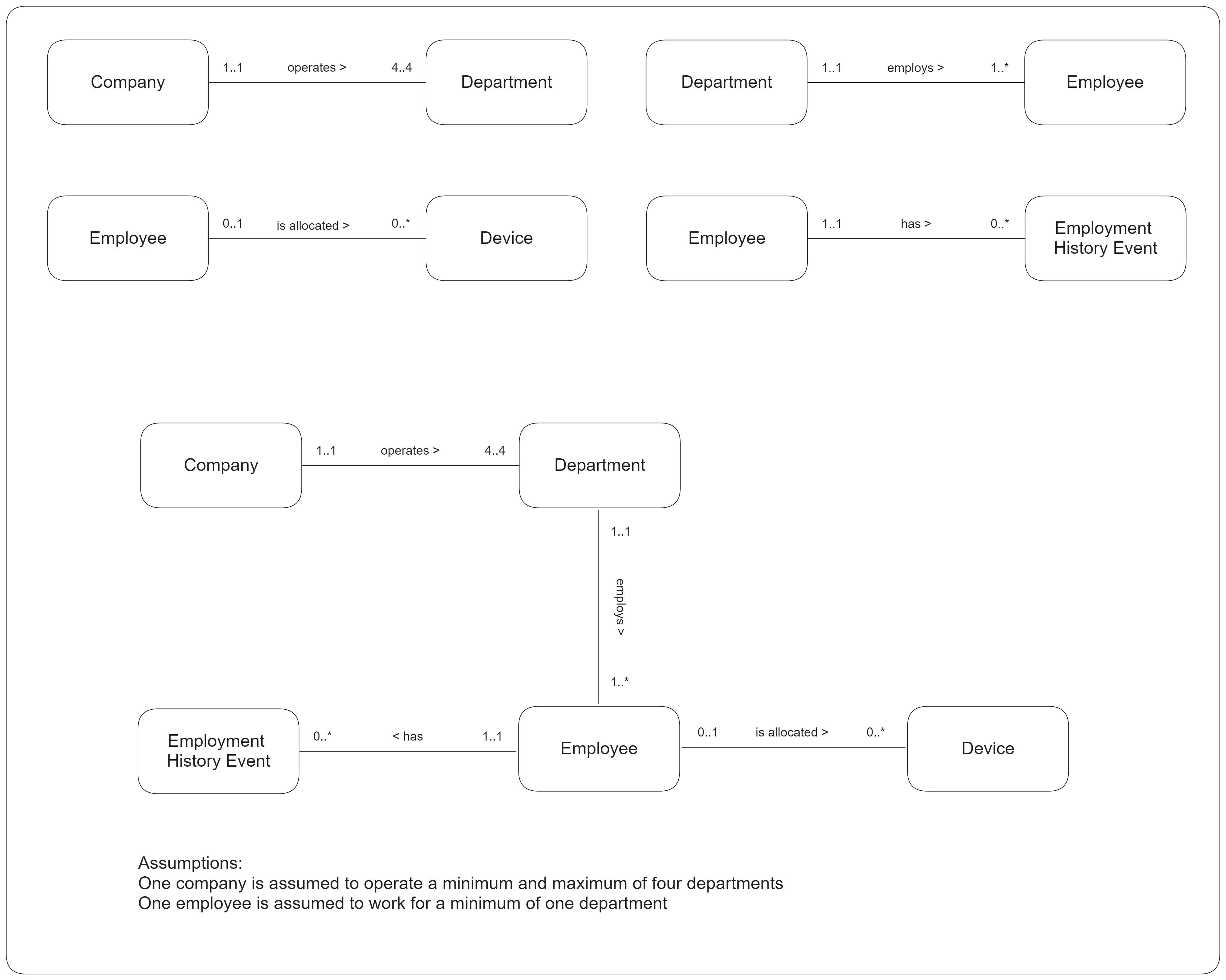
An employee does not have to use a company car

An employee can use only one company car at most

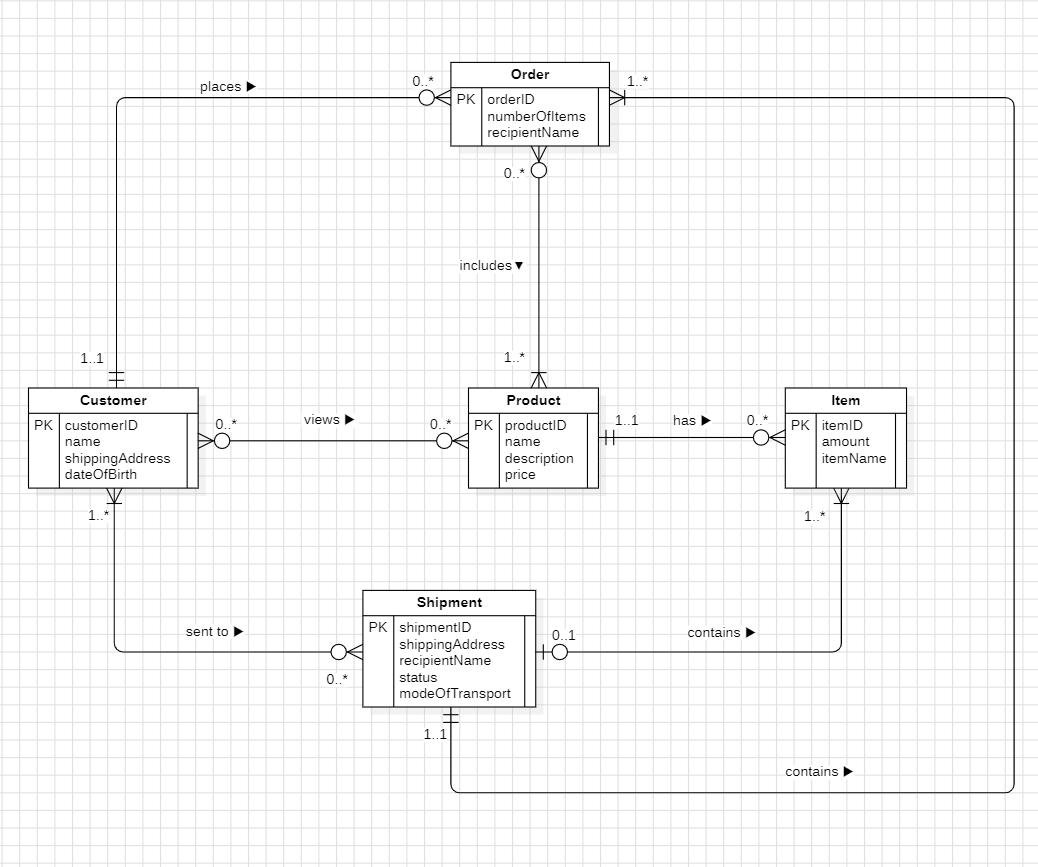
A company car does not have to be used by an employee

A company car can be used by two employees at most

# Exercise 02



# Exercise 03



LEC 03 – Logical ERD

Foreign key – the PK of another table in the current table

E.g.: assume there are two tables: Staff and Branch. In the staff table, the primary key will be staffID and the foreign key can be branchID which is the primary key of the Branch table

Foreign key can be null if participation is optional