ER to Schema

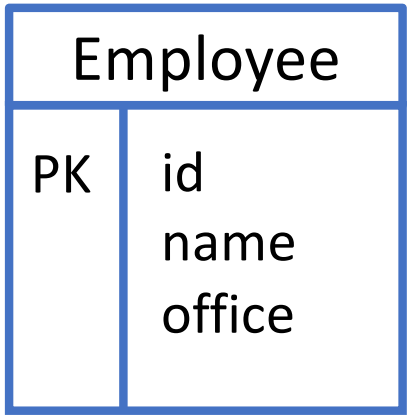
COMP23111 - Database Systems

Gareth Henshall
Lecturer in Computer Science

From ER to schema, guidelines

- **entity:** each entity type becomes a relation (table)
- **attribute.** entity attributes become attributes of the relation
- the **identifier** (*) of the entity becomes the PK of the relation
- **a relationship** is expressed as an FK / joining table

Strong entity type

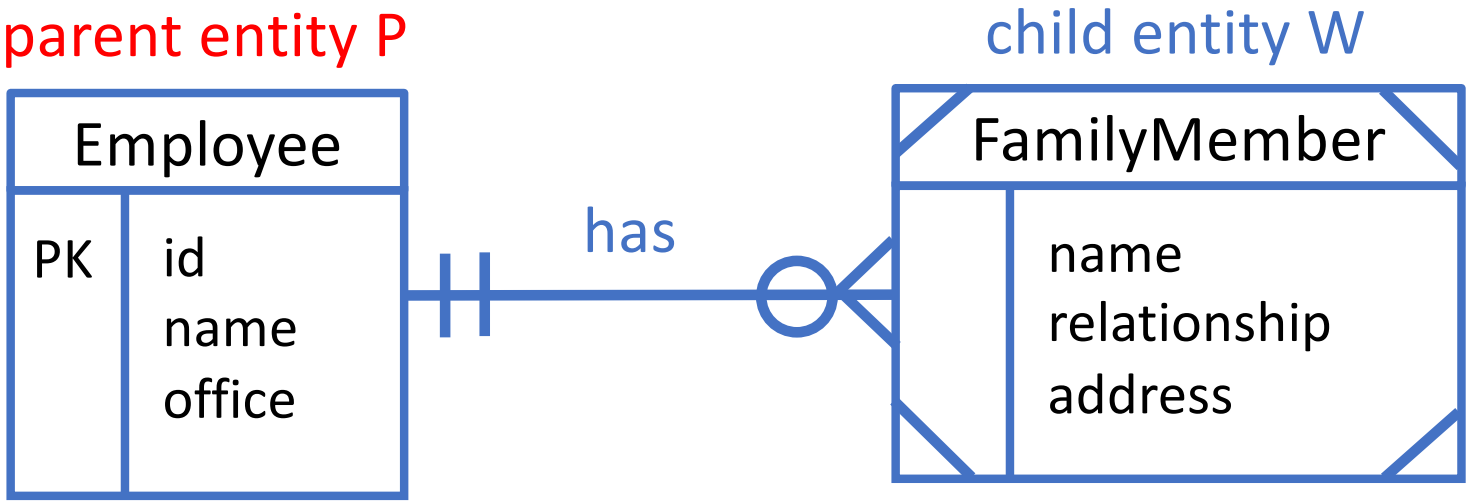


- entity type maps to relation
- constraints, defaults, keys must be given

Employee

	id	name	office
	int	VARCHAR(50)	VARCHAR(100)
constraint	NOT NULL	NOT NULL	can be NULL
default	none	none	NULL
key	PK, AUTO_INCR		

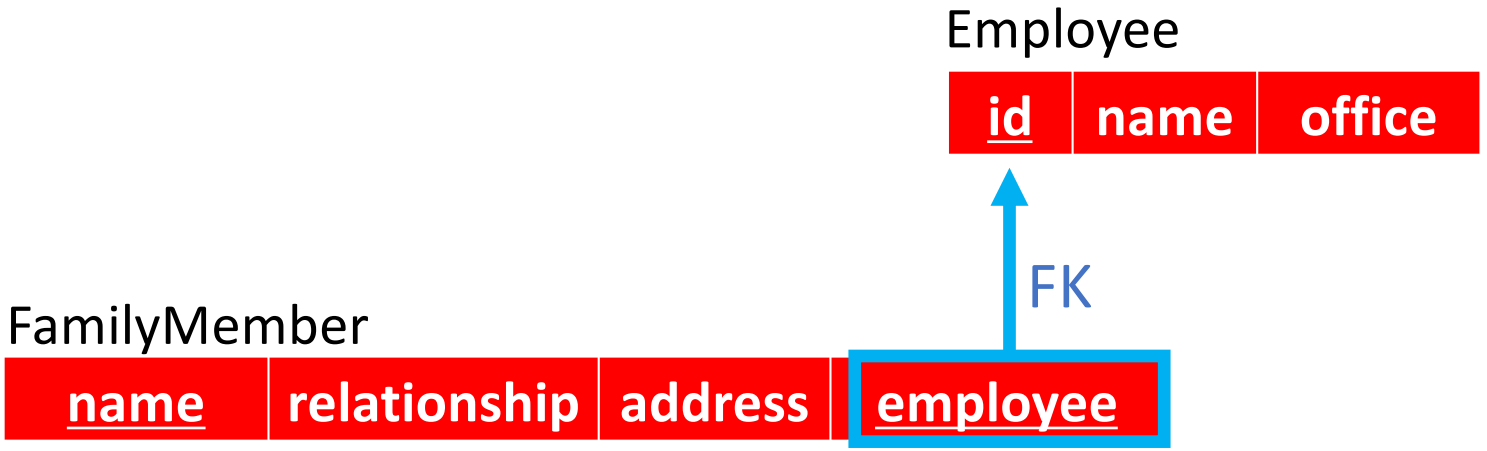
Weak entity



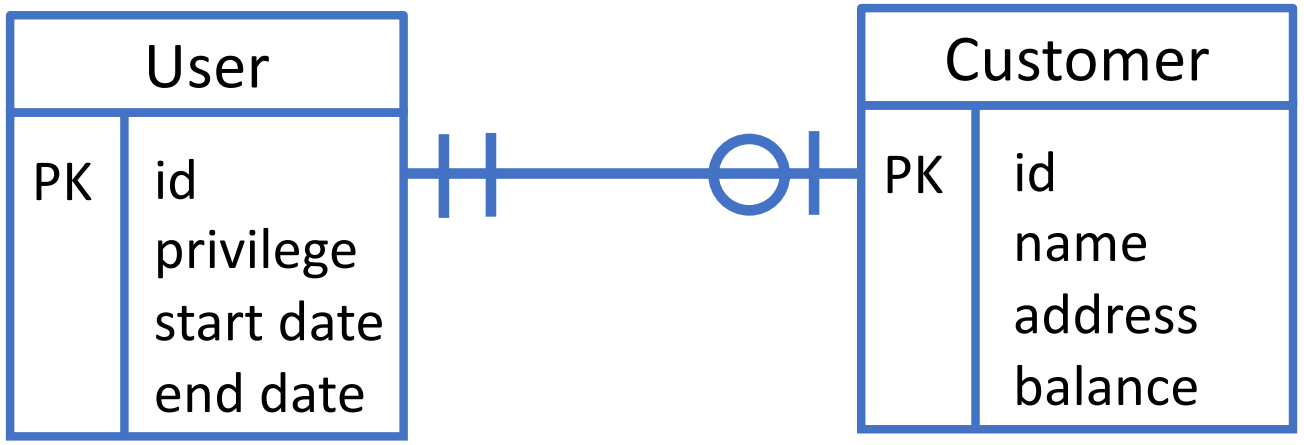
- a weak entity W is identified by a **composite** key
- (P.PK, W.some_attribute)
- in this case: (Employee.PK, FamilyMember.name)

Weak entity

- the PK of FamilyMember is composite key (employee, FamilyMember.name)



1-to-1 relationship



- identify table T that fully participates (| |) and add a FK attribute into T which is the PK of the other table

1-to-1 relationship

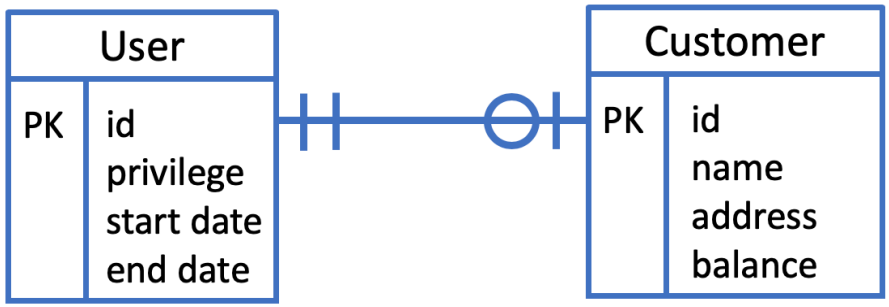
User

id	privilege	start date	end date
97	admin	2015	NULL
98	customer	2017	NULL
99	customer	2017	2018

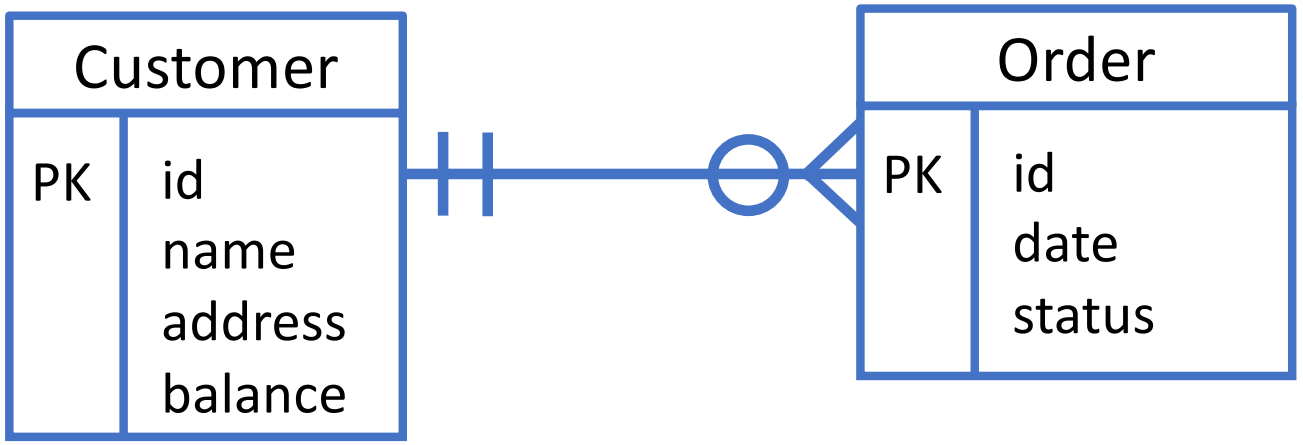
FK cannot be null

Customer

id	name	address	balance	userID
411	Bob	UK	£100	98
412	Carole	USA	£0	99



1-to-many relationship



- map the entities to relations
- identify the relation R on the “many” side of the relationship
- add a FK to R, which is the PK of the table on the “1” side

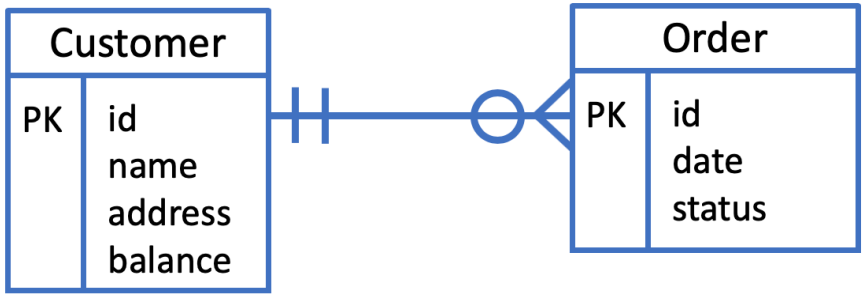
1-to-many

Customer

id	name	address	balance	userID
411	Bob	UK	£100	98
412	Carole	USA	£0	99

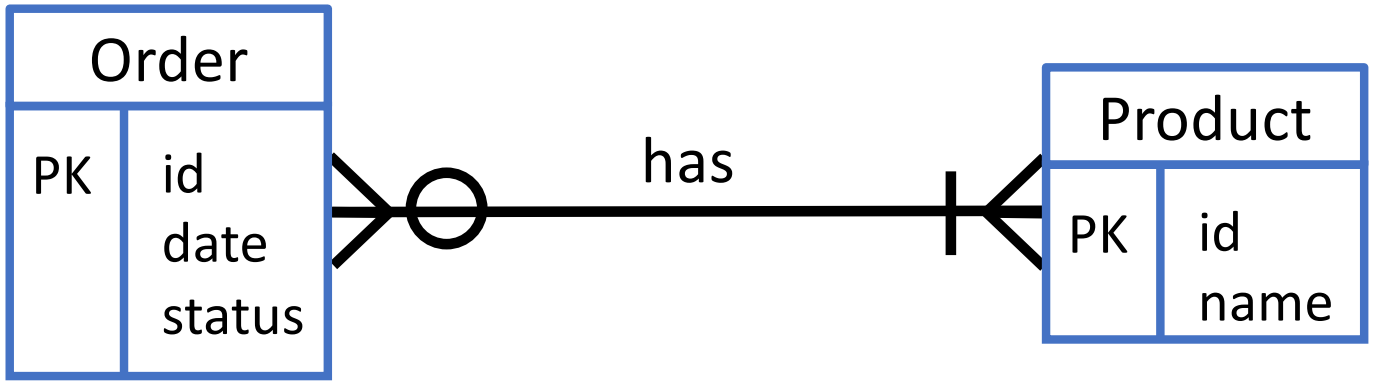
Order

id	date	status	custId
31	2018	fulfilled	412
32	2018	fulfilled	411
33	2018	cancelled	412
34	2019	fulfilled	411
35	2019	pending	411



Many-to-many relationships

- the relationship between Order and Product



Many-to-many relationships

- cannot be directly implemented. Well let's try…
- suppose order 98 is products 1,2,3. Order 101 is products 2,3.

Order

id	date	status	processor	product (FK)
98	010918	paid	Boris	1,2,3
101	021919	pending	David	2,3

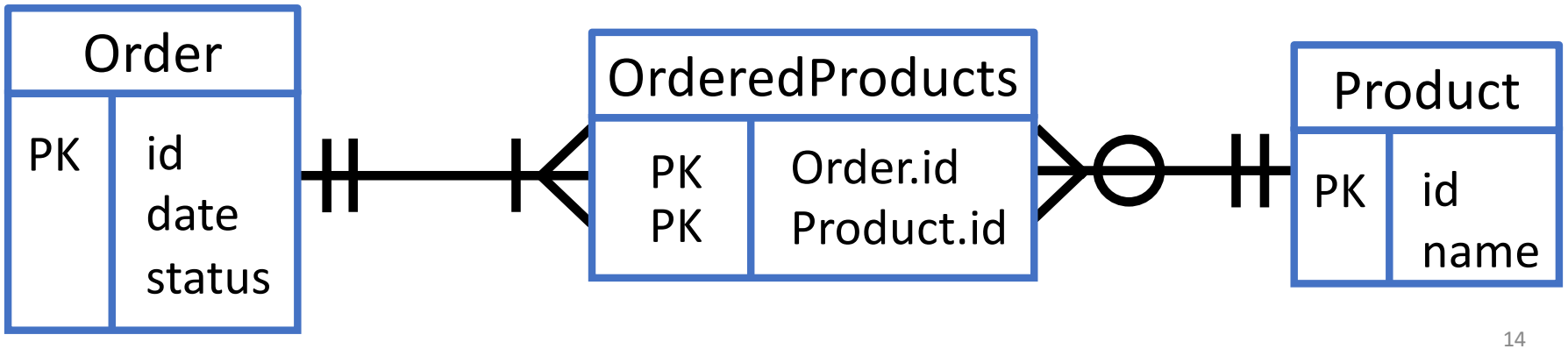
Product

id	name	order (FK)
1	shoes	98
2	guitar	98,101
3	camera	98,101
4	book	NULL

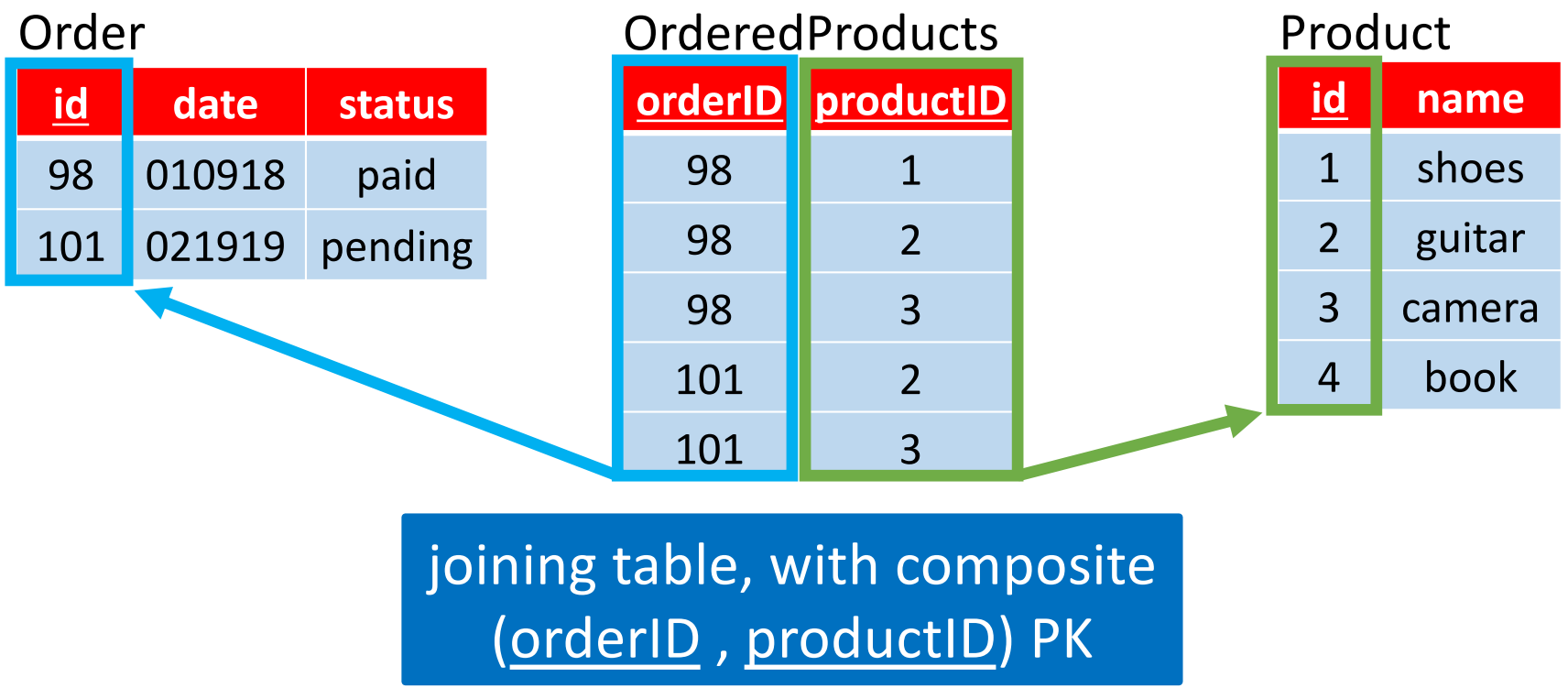
- But we can't store multiple values in a single attribute. This violates atomicity.

Many-to-many relationships

- solution: at design time, introduce new "joining" (aka "bridging" / "intersection" / "junction") entity type OrderedProducts
- holds PKs of each entity in the M-M relationship



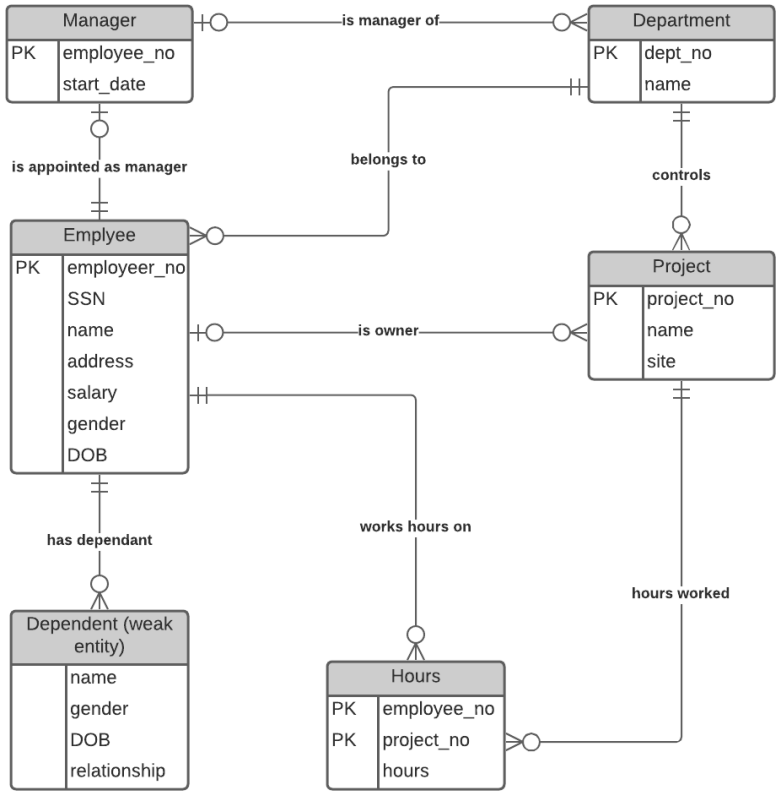
Many-to-many relationships

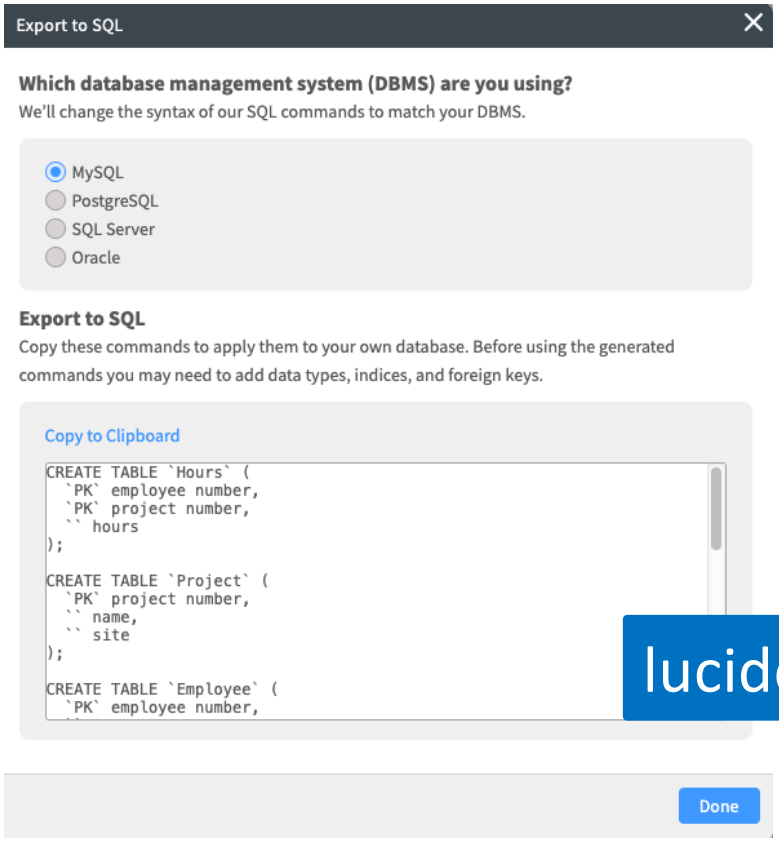


From ER to relations, “auto”

- these guidelines can be auto-applied to an ER diagram, to create the SQL definition of the DB schema
- examples: lucidchart.com (free), erdplus.com (free), visual-paradigm.com (paid) [+ many others]
- but:
  - relies on the ER diagram being sensible, of course
  - data domains, defaults, constraints must be given
  - some tools don’t include relationships (not very useful!)
  - works most effectively for simple schemas

From ER to relations, “auto”





lucidchart.com