



SCC.221 DATA Engineering

2025 - Week 1 – Module Information.
Uraz C Turker

What will you learn this week?



Understand fundamental concepts of databases

Learning outcomes



- Definitions for
 - Data
 - Data Engineering
 - Database
 - Database Management System
 - Basics of Entity Relationship Diagrams

What is a Data?



What is data?

What is a Data?



Data is anything that can be represented using Binary notation.

We can then encode data in File:

0	1	0	0	1	1	1	1	0
0	1	0	0	1	1	1	1	1
0	0	1	1	1	0	1	0	



N
O
:

Source of Data

- Data is being created whenever we:
 - Eat,
 - Travel,
 - Chat,
 - Talk over a line on a phone, etc.

A source of data: Games



The aim of having data?



- is to retrieve information.
- To achieve this, we need to process this data.
- What is the fundamental step to process data?
- To establish a flexible, understandable, and a common representation for the data.

Let us know you agree to cookies

We use cookies to give you the best online experience. Please let us know if you agree to cookies.

BBC Sign in Home News Sport Weather iPlayer Sounds

NEWS

Home | Cost of Living | War in Ukraine | Coronavirus | Climate | UK | World | Business | Politics | Tech

Technology

Facebook's data-sharing deals exposed

19 December 2018 · Comments

What is Data Engineering?



- “Data engineering is the process of designing and building systems that allow people to collect, manage, and analyse data. Data engineers work to make raw data usable for data scientists and business analysts so that organizations can use it to improve their performance.”

What is Data Engineering?



- “Data engineering is the process of designing and building systems that allow people to collect, manage, and analyse data. Data engineers work to make raw data usable for data scientists and business analysts so that organizations can use it to improve their performance.”

Data engineers are responsible for:



- **Data pipelines:** Creating data pipelines (flows) to manage and process large sets of data
- **Data integration:** Ensuring that data from different sources is integrated seamlessly
- **Data quality:** Ensuring that data is of high quality and that the data infrastructure is reliable and efficient
- **Data analysis:** Creating raw data analyses to provide predictive models and show trends
- **Data security:** Managing and storing data securely to protect it from loss or theft
- **Automation:** Creating ways to automate tasks within the data pipeline to improve efficiency
- Data engineers often work for larger organizations and may use tools and technologies such as Hadoop, MongoDB, and Kafka.

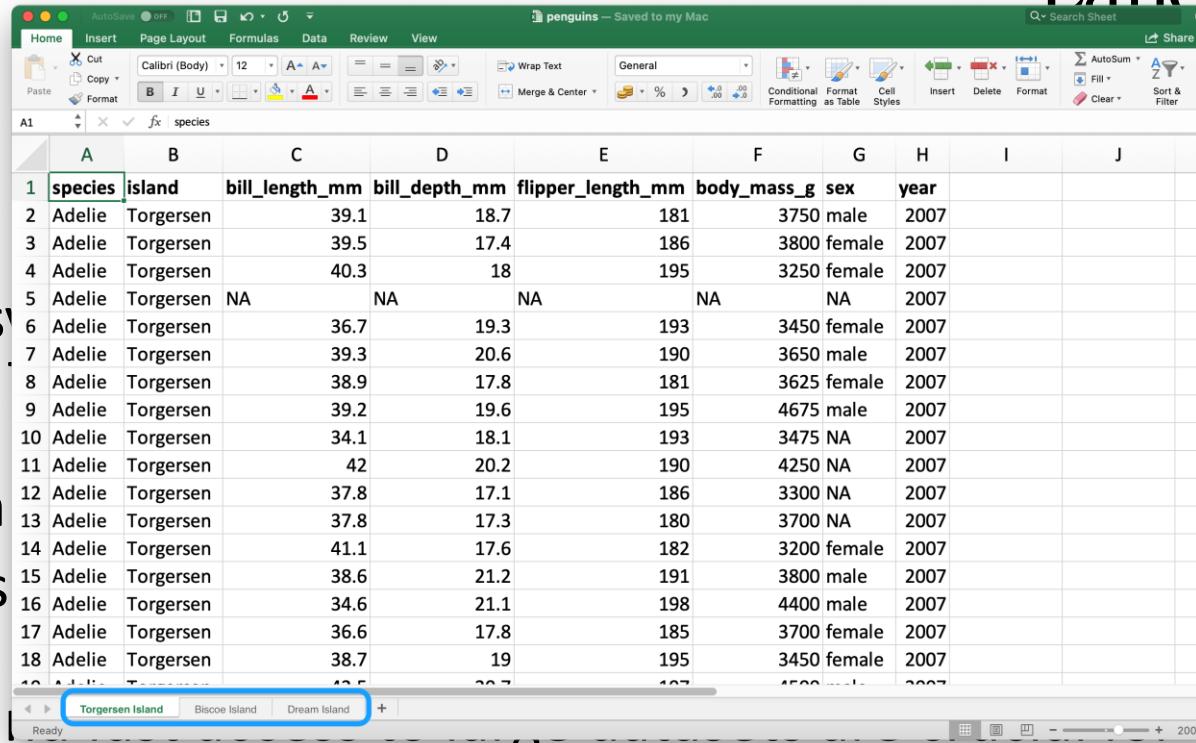
Data engineers are responsible for:



- **Data pipelines:** Creating data pipelines (flows) to manage and process large sets of data
- **Data integration:** Ensuring that data from different sources is integrated seamlessly
- **Data quality:** Ensuring that data is of high quality and that the data infrastructure is reliable and efficient
- **Data analysis:** Creating raw data analyses to provide predictive models and show trends
- **Data security:** Managing and storing data securely to protect it from loss or theft
- **Automation:** Creating ways to automate tasks within the data pipeline to improve efficiency
- Data engineers often work for larger organizations and may **use tools** and technologies such as Hadoop, **MongoDB**, and Kafka.

Databases

- Databases are structured systems for storing, retrieving, and managing data. They include various types (e.g., relational, NoSQL, PostgreSQL).
- Raw data is often stored in databases.
- Data stored in databases is organized into tables and models, which define how data is related.
- Efficient query execution algorithms are designed for databases.
- Secure storage and access control of sensitive data is handled within databases.
- Traditional and cloud-based databases are used to store data, but as the data scales, the infrastructure must also scale.



species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex	year
Adelie	Torgersen	39.1	18.7	181	3750	male	2007
Adelie	Torgersen	39.5	17.4	186	3800	female	2007
Adelie	Torgersen	40.3	18	195	3250	female	2007
Adelie	Torgersen	NA	NA	NA	NA	NA	2007
Adelie	Torgersen	36.7	19.3	193	3450	female	2007
Adelie	Torgersen	39.3	20.6	190	3650	male	2007
Adelie	Torgersen	38.9	17.8	181	3625	female	2007
Adelie	Torgersen	39.2	19.6	195	4675	male	2007
Adelie	Torgersen	34.1	18.1	193	3475	NA	2007
Adelie	Torgersen	42	20.2	190	4250	NA	2007
Adelie	Torgersen	37.8	17.1	186	3300	NA	2007
Adelie	Torgersen	37.8	17.3	180	3700	NA	2007
Adelie	Torgersen	41.1	17.6	182	3200	female	2007
Adelie	Torgersen	38.6	21.2	191	3800	male	2007
Adelie	Torgersen	34.6	21.1	198	4400	male	2007
Adelie	Torgersen	36.6	17.8	185	3700	female	2007
Adelie	Torgersen	38.7	19	195	3450	female	2007

What we had and why did we invent DBMS?



- Previously we had File-processing applications:
 - A set of files are processed to retrieve information.
 - Hard to implement,
 - Optimisation, performance, reliability, adoption, re-use were main issues.
- Standardisation of concepts.
 - Representation,
 - Coding,
 - Formatting, etc.
- Similar question “Why do we have an operating system?”

What is a Data, Database, Database Management System (DBMS)?

Database: It represents a massive amount of persistent data according to **a logical model**

The logical model can be represented graphically by.
UML or **Entity Relationship Diagrams.**

The logical model allows us to reason about the logical representation of the data to **gain the flexibility to establish an excellent data representation power to aid processing.**

What is a Data, Database, Database Management System (DBMS)?

DBMS: Provides efficient, reliable, convenient, safe, multi-user friendly storage and access to massive amounts of persistent data.

What is a Data, Database, Database Management System (DBMS)?

DBMS: Provides efficient, reliable, convenient, safe, multi-user friendly storage and access to massive amounts of persistent data.

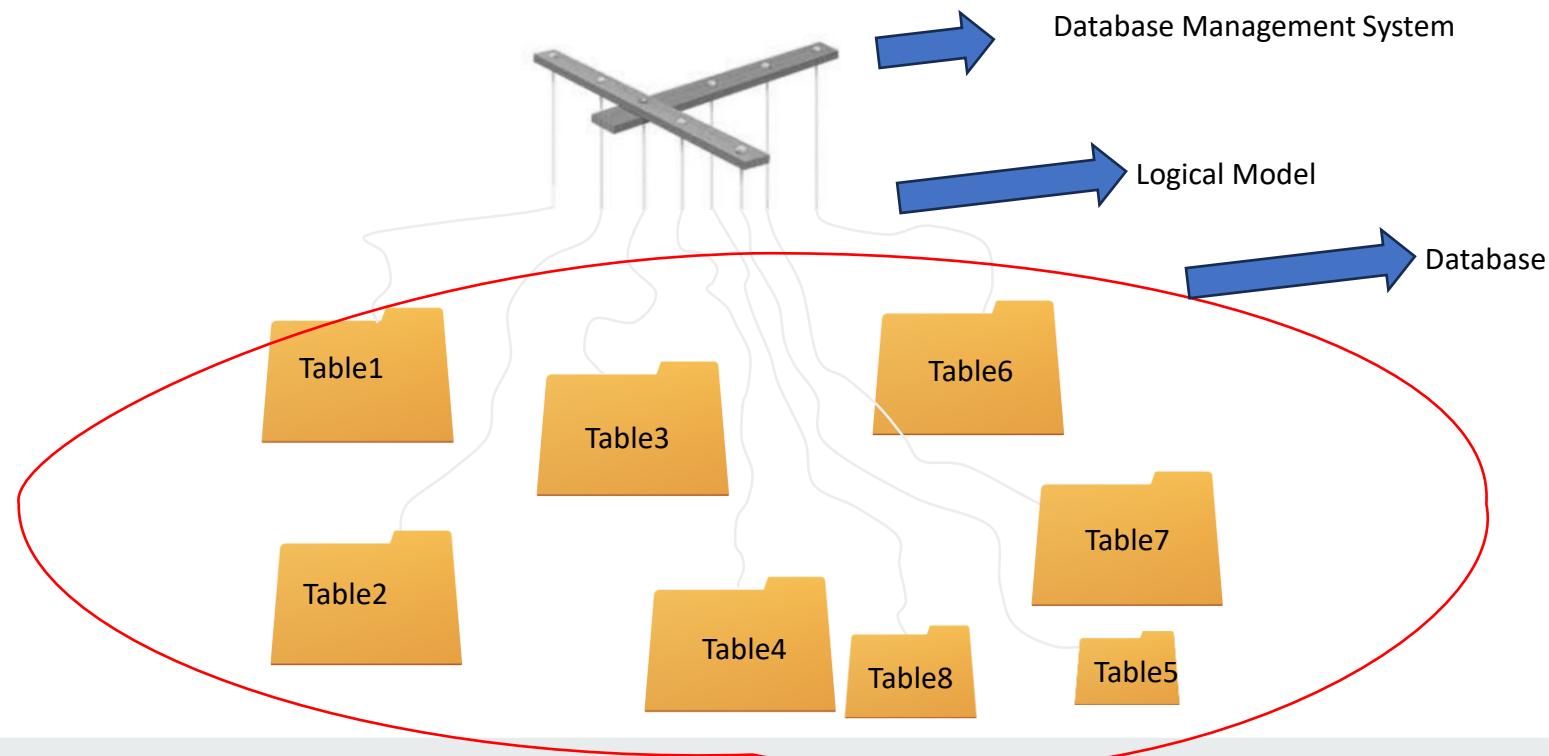
This is why they are very popular.

Relation between DBMS, Database and the Logical Model.



Database:

It represents a massive amount of persistent data according to **a logical model**



Databases in real life.. EVE Online



https://docs.google.com/spreadsheets/d/1bVeIR2hB27e14mzHQfcwsJZP_Rg3u1BPmx6jDN_13fo/edit#gid=0

https://wiki.eveuniversity.org/Static_Data_Export



Databases in real life. Counter Strike.
<https://www.csgodatabase.com/about/>



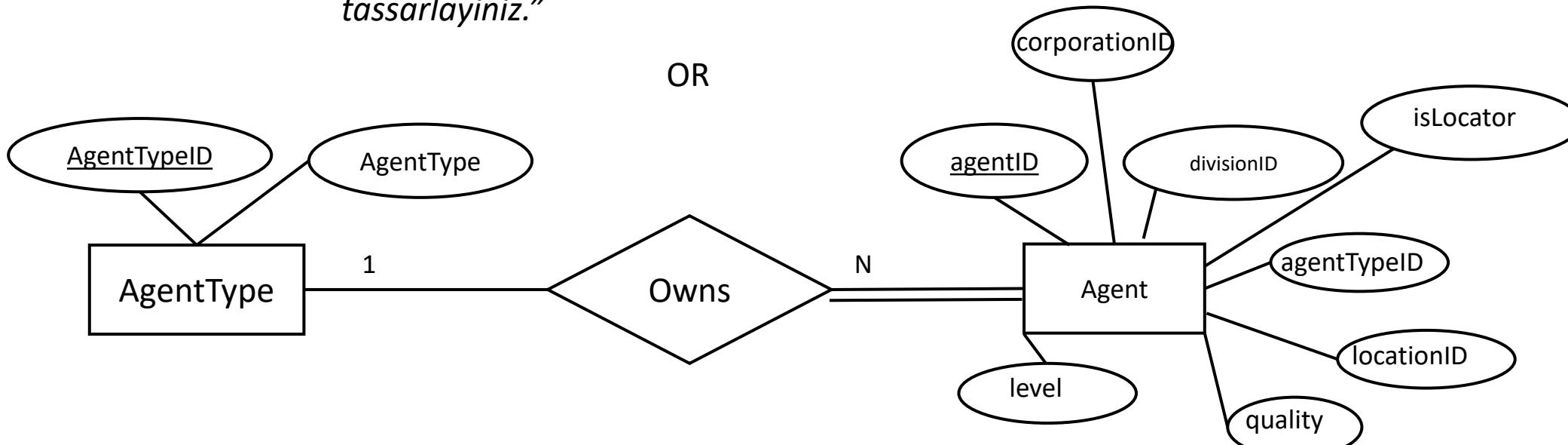
Representing the logic of the database



- Designing a database requires documentation that will be used to i) reveal database requirements and ii) perform debugging/update and maintenance with ease.
- As in the case of many software engineering fields, the document is heavily on graphical representation (similar to UMLs).
- Among others, **Entity-Relationship diagrams** are used to represent the logical structure of the database.

Example

- In EVE Game there are:
- **Agents**
 - Where each agent has **agentID**, **divisionID**, **corporationID**, **locationID**, **level**, **quality**, **agentTypeID**, **isLocator**;
- **AgentTypes**
 - where each type has **agentTypeID**, **agentType**.
- *Query (Eng): “Please design a database where every Agent must have exactly one AgentType, but every AgentType can be associated with several Agents.”*
- *Istem (Tr): “Lütfen, her Agent’in bir AgentType’inin olduğu ve her AgentTypeinin bir veya birden fazla Agent ile ilişkisi olduğu bir veri tabanı tassarlayınız.”*

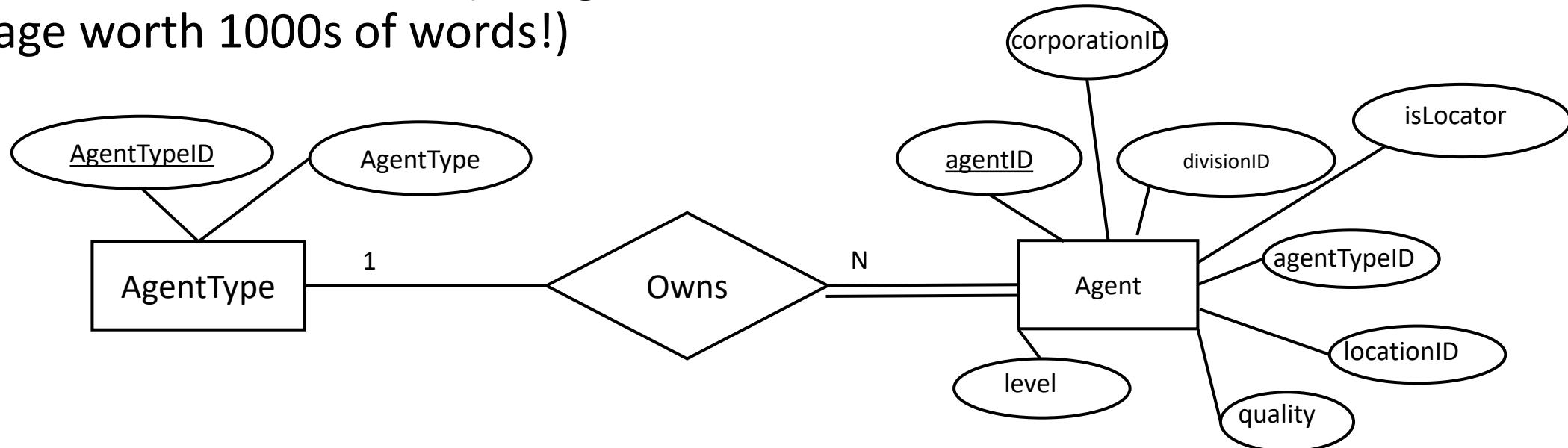


The logical model: Entity relationship model (Chens Representation).

A graphical representation of the logical view of a Database.

Communication method (a single image worth 1000s of words!)

https://wiki.eveuniversity.org/Static_Data_Export
agtAgents.csv.bz2
agtAgentTypes.csv.bz2



Key concepts before proceeding.



- Data Definition Language (DDL) is used to create tables.
- Data Manipulation Language (DML) is used to modify tables.

Key people before proceeding.



- DBMS Implementer.
- DB Designer.
- DB Application Developer.
- DB Admin.



The following slides are here to assure
Uraz that everybody understands ERDs!

Please read Elmasri's (Fundamentals of database systems) book
Chapter 3.

From you...



Show chart data

What more support can be provided?

- Can you please add the slides as pdf?
-

Any other comments

- We can find power points only but it would be really helpful if we can get them as pdf format

From you...



Show chart data

What more support can be provided?

- Can you please add the slides as pdf?

Any other comments

- We can find power points only but it would be really helpful if we can get them as pdf format

Key concepts of ER diagrams: Entity

- **Entity:** An object that has distinguishable **attributes**.

- **Entity:** An apple

- **Attributes:**
 - Colour
 - Size
 - Origin

- **Entity:** AgentType
Attributes:
 - AgentTypeID
 - AgentType

AgentTypeID	AgentType
1	NonAgent
2	BasicAgent

Entity set: A conceptual representation of entities w.r.t the same set of attributes is called an Entity Set.

Colour	Size	Origin
Red	Large	Spain
Green	Medium	Madagascar

Key concepts of ER: Entity

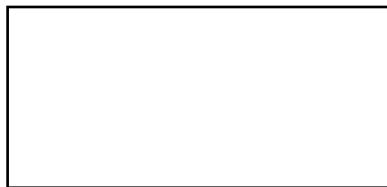


- **Entity:** An object that has distinguishable **attributes**.
- Rectangle is used to represent an entity set.

Key concepts of ER: Entity



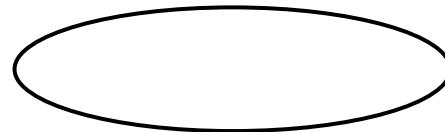
- **Entity:** An object that has distinguishable **attributes**.
- Rectangle is used to represent an entity set.



Key concepts of ER: Entity



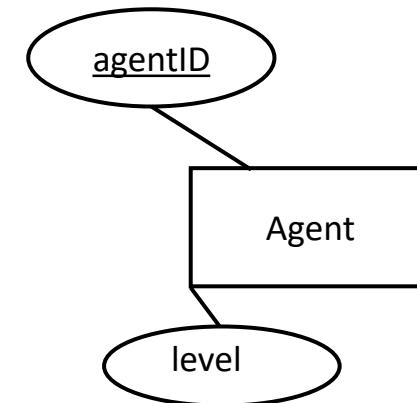
- **Entity:** An object that has distinguishable **attributes**.
- Rectangle is used to represent an entity set.
- Oval is used to represent an attribute.



Key concepts of ER: Entity

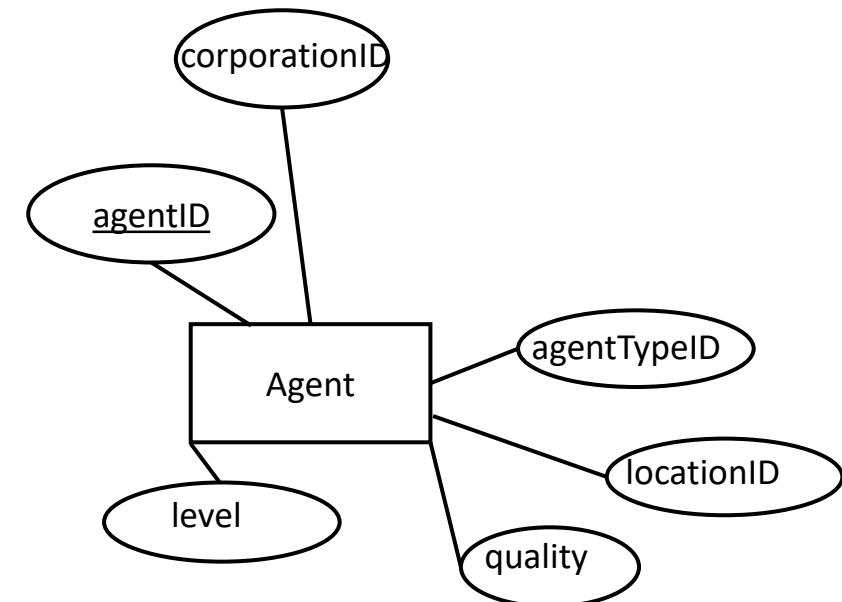


- **Entity:** An object that has distinguishable **attributes**.
- Rectangle is used to represent an entity set.
- Oval is used to represent an attribute.



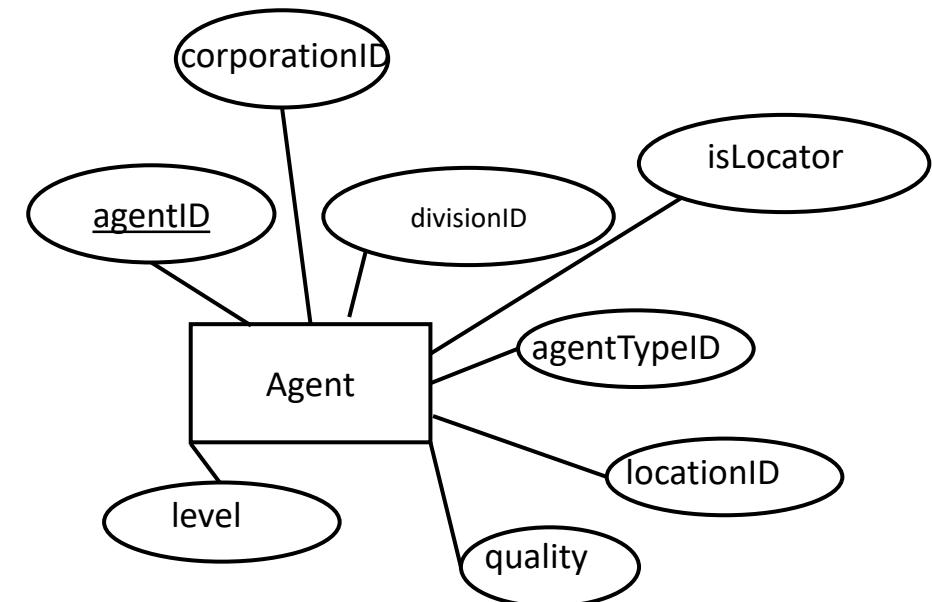
Key concepts of ER: Entity

- **Entity:** An object that has distinguishable **attributes**.
- Rectangle is used to represent an entity set.
- Oval is used to represent an attribute.



Key concepts of ER: Entity

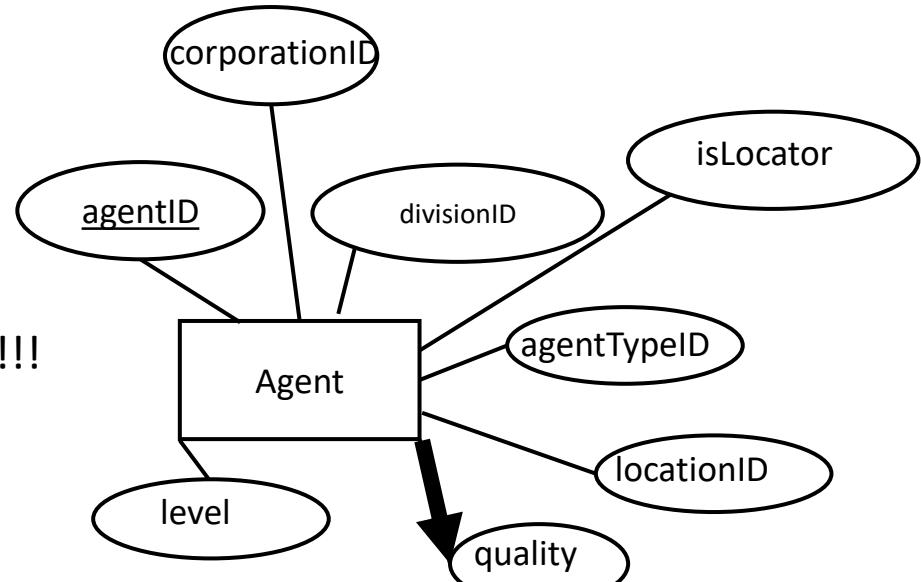
- **Entity:** An object that has distinguishable **attributes**.
- Rectangle is used to represent an entity set.
- Oval is used to represent an attribute.



Key concepts of ER: Entity

- **Entity:** An object that has distinguishable **attributes**.
- Rectangle is used to represent an entity set.
- Oval is used to represent an attribute.

WE NEVER USE ARROWS!!!!

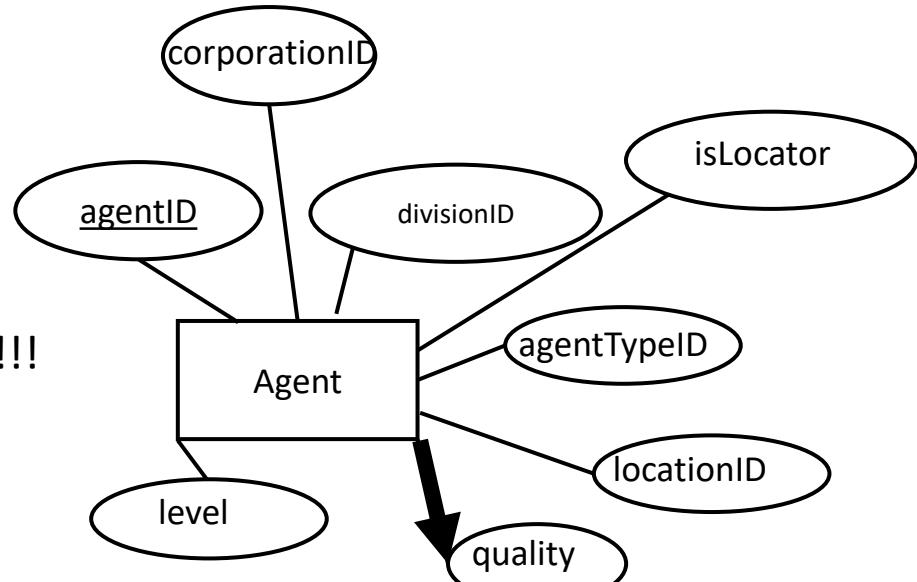


Key concepts of ER: Entity

- **Entity:** An object that has distinguishable **attributes**.
- Rectangle is used to represent an entity set.
- Oval is used to represent an attribute.

WE NEVER USE ARROWS!!!!

Two semesters ago, one student failed
all of the coursework!



Exercise...



Key concepts of ER: Entity



- **Entity:** An object that has distinguishable **attributes**.
- **Entity A Solar System** can have *RegionName*, *ConstellationName*, *System Name*, *X*, *Y*, *Z*, *Security*, *SecurityClass* as its **attributes**.

https://docs.google.com/spreadsheets/d/1bVeIR2hB27e14mzHQfcwsJZP_Rg3u1BPmx6jDN_13fo/edit#gid=0

Eve Solar System as Entity Set

RegionName	ConstellationName	System Name	SystemID	X	Y	Z	Security	SecurityClass
Domain	Throne Worlds	Amarr	30002187	-204748707250009000	40238379936571400	-57621278902421000	1.0	A
Heimatar	Sveipar	Ammold	30002547	-85348665798335600	18749115057675500	45430156768496800	1.0	A
Lonetrek	Sela	Amsen	30001392	-145493013225457000	101594111567536000	137638171422790000	1.0	A
Sinq Laison	Nexus	Bourynes	30002715	-166183387531946000	44475501577819500	26973565770462400	1.0	A
Domain	Parud	Chaven	30003489	-180372429369955000	60008167808112100	-54794737517953000	1.0	A

Some other databases are given in XML format. <https://archive.org/details/stackexchange>

Key concepts of ER: Entity

Consider an Entity Set called CAR
ENTITY SET / Schema / Table

Attributes

Car -> Model, Weight, Length, Max_Speed.

	Model	Weight	Length	Max_Speed
First Entity	BMW 3.21	1400	3.21	200
Second Entity	Toyota_Corolla	1300	3.16	200
Third Entity	Hyundai E.GLS	1400	3.16	210



Key concepts of ER: Entity

Consider an Entity Set called CAR
ENTITY SET / Schema / Table

Attributes

Car -> Model, Weight, Length, Max_Speed.

	Model	Weight	Length	Max_Speed
First Entity	BMW 3.21	1400	3.21	200
Second Entity	Toyota_Corolla	1300	3.16	200
Third Entity	Hyundai E.GLS	1400	3.16	210
Fourth Entity	Hyundai E.GLS	1400	3.16	210

But we have redundancy here.. Third and Fourth Entities are the same!



Key concepts of ER: Entity

Consider an Entity Set called CAR
ENTITY SET / Schema / Table

Attributes

Car -> Model, Weight, Length, Max_Speed.

	Model	Weight	Length	Max_Speed
First Entity	BMW 3.21	1400	3.21	200
Second Entity	Toyota_Corolla	1300	3.16	200
Third Entity	Hyundai E.GLS	1400	3.16	210



To avoid having duplicate entities, we need to be able to identify each row uniquely...
And, so select an **attribute** that distinguishes an entity within the Car entity set.

Key concepts of ER: Entity

Consider an Entity Set called CAR
ENTITY SET / Schema / Table

Attributes

Car -> Model, Weight, Length, Max_Speed.

	Model	Weight	Length	Max_Speed
First Entity	BMW 3.21	1400	3.21	200
Second Entity	Toyota_Corolla	1300	3.16	200
Third Entity	Hyundai E.GLS	1400	3.16	210



To avoid having duplicate entities, we need to be able to identify each row uniquely...
And, so select an **attribute** that distinguishes an entity within the Car entity set.

Model uniquely identifies an entity (the whole row) within this entity set.

Key concepts of ER: Entity

Consider an Entity Set called CAR
ENTITY SET / Schema / Table

First Entity
Second Entity
Third Entity

Attributes

Car -> Model, Weight, Length, Max_Speed.

	<u>Model</u>	Weight	Length	Max_Speed
First Entity	BMW 3.21	1400	3.21	200
Second Entity	Toyota_Corolla	1300	3.16	200
Third Entity	Hyundai E.GLS	1400	3.16	210

To avoid having duplicate entities, we need to be able to identify each row uniquely...
And, so select an **attribute** that distinguishes an entity within the Car entity set.

Model uniquely identifies an entity (the whole row) within this entity set.

44

The attribute that uniquely identifies entities within the entity set is called a '**key attribute**'.



Key Attribute and Primary Key Attribute

There can be more than one key attribute for a given Entity set.

Model	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1200	3.18	220
Hyundai E.GLS	1100	3.16	210

The designer has to select key attribute(s) as the **Primary Key Attribute** according to the context.

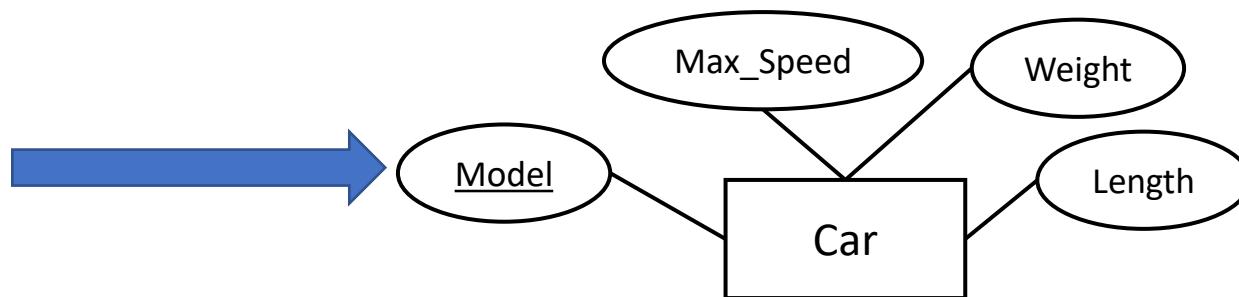
“Car Models should not repeat.”

Key concepts of ER: Entity

- Car -> Model, Weight, Length, Max_Speed.

<u>Model</u>	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

The primary key attributes are represented with solid underlines (Model is selected).



Exercise...



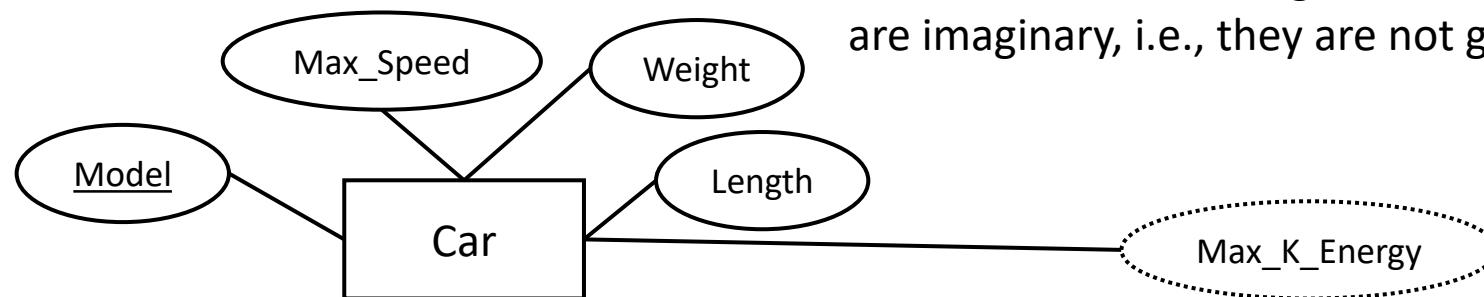
Key concepts of ER: Entity

- Car -> Model, Weight, Length, Max_Speed.

Model	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

Some attributes are computed: Derived attributes.

Derived attributes are given using **dashed lines**. They are imaginary, i.e., they are not given in the table.



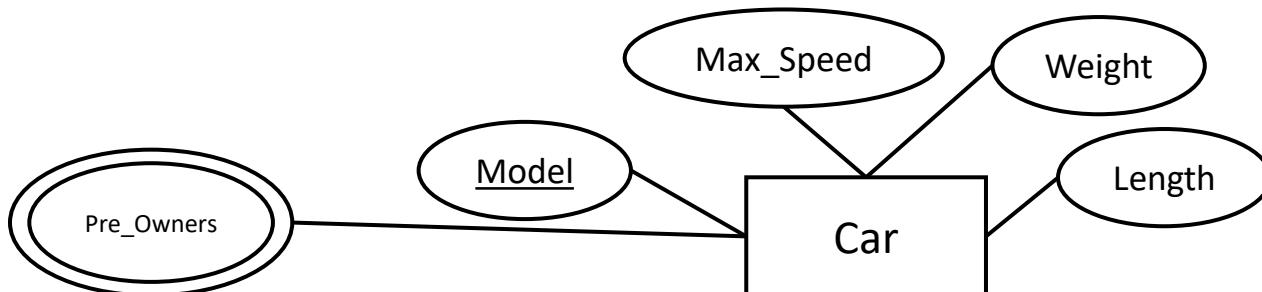
Key concepts of ER: Entity

- Car -> Model, Weight, Length, Max_Speed.

Model	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

Some attributes have multi-values: Multi-valued attributes.

Multi-valued attributes are given **using double lines**.



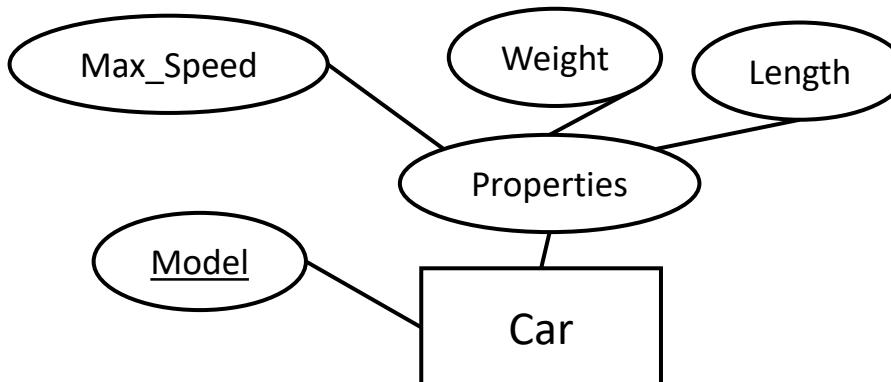
Key concepts of ER: Entity



- Car -> Model, Weight, Length, Max_Speed.

Model	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

We can create composite attributes using others.



Composite attributes are given as a **group of attributes**, and they are imaginary, i.e., a composite attribute is not given in the table.

Key concepts of ER: Entity



- **Entity**: An object that has distinguishable attributes.
- What other entity sets are there for EVE Online?

https://wiki.eveuniversity.org/Static_Data_Export



Key concepts of ER: Entity

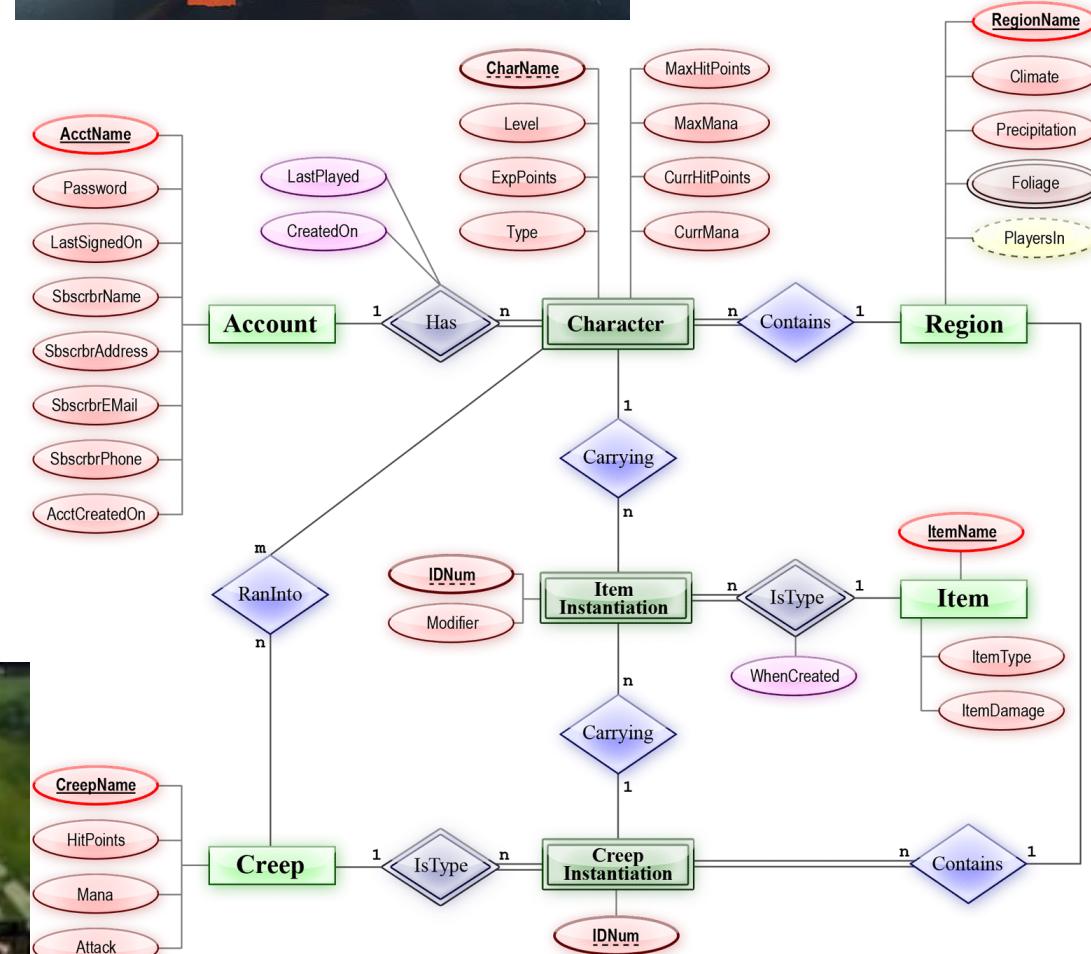
- **Entity:** An object that has distinguishable attributes.
- What entities and attributes can we have in multiplayer computer games?

Entity: Accounts

- Attributes:
 - AccountName
 - Password
 - ...

Entity: Character

- Attributes
 - CharacterName
 - Level
 - ...



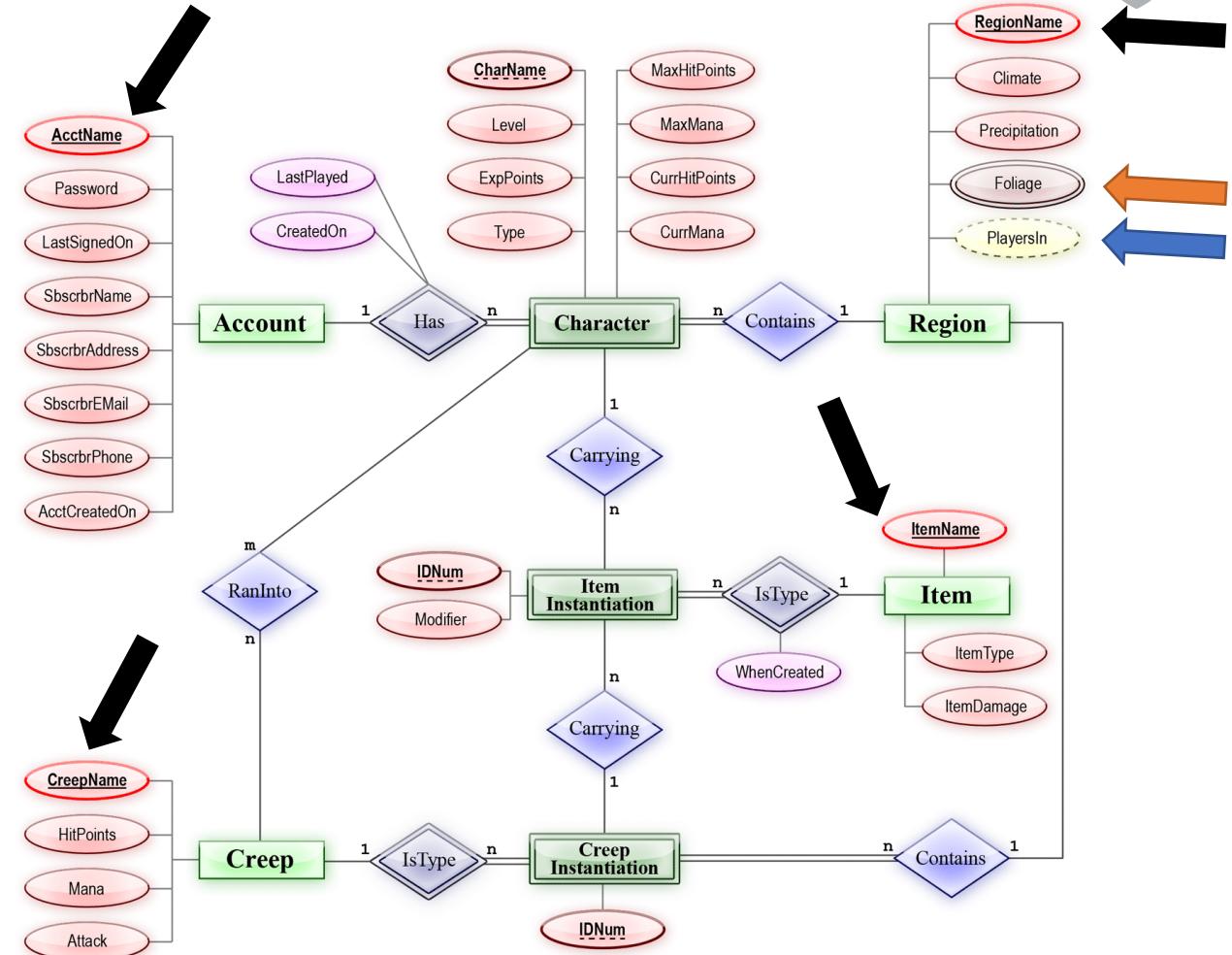
Exercise...



Check yourself!

- Which are multi-valued attributes?
- Which are Derived Attributes?
- Which are the primary key attributes?

DOTA Database



Relations



-
- In real life, an application usually interacts with several Entity Sets.
 - Why?



Relations



-
- In real life, an application usually interacts with several Entity Sets.
 - Solar System



Relations



- In real life, an application usually interacts with several Entity Sets.
- Solar System, Space Ship



Relations



-
- In real life, an application usually interacts with several Entity Sets.
 - Solar System, Space Ship, Commander. Etc..



Relations



- In real life, an application usually interacts with several Entity Sets/Tables.
- Keeping everything in a single table will lead to poor performance as everything will be kept in a single file. Queries like *“Tell me the names of commanders that can pilot Space_Shuttle Falcon”* will take ages.
-



Relations



- In real life, an application usually interacts with several Entity Sets/Tables.
- Keeping everything in a single table will lead to poor performance as everything will be kept in a single file. Queries like “*Tell me the names of commanders that can pilot Space_Shuttle Falcon*” will take ages.
- Therefore, a database has several tables instead of a single table.



Relations



Template for 2022 annual reporting on Plan S approved Transformative Journals

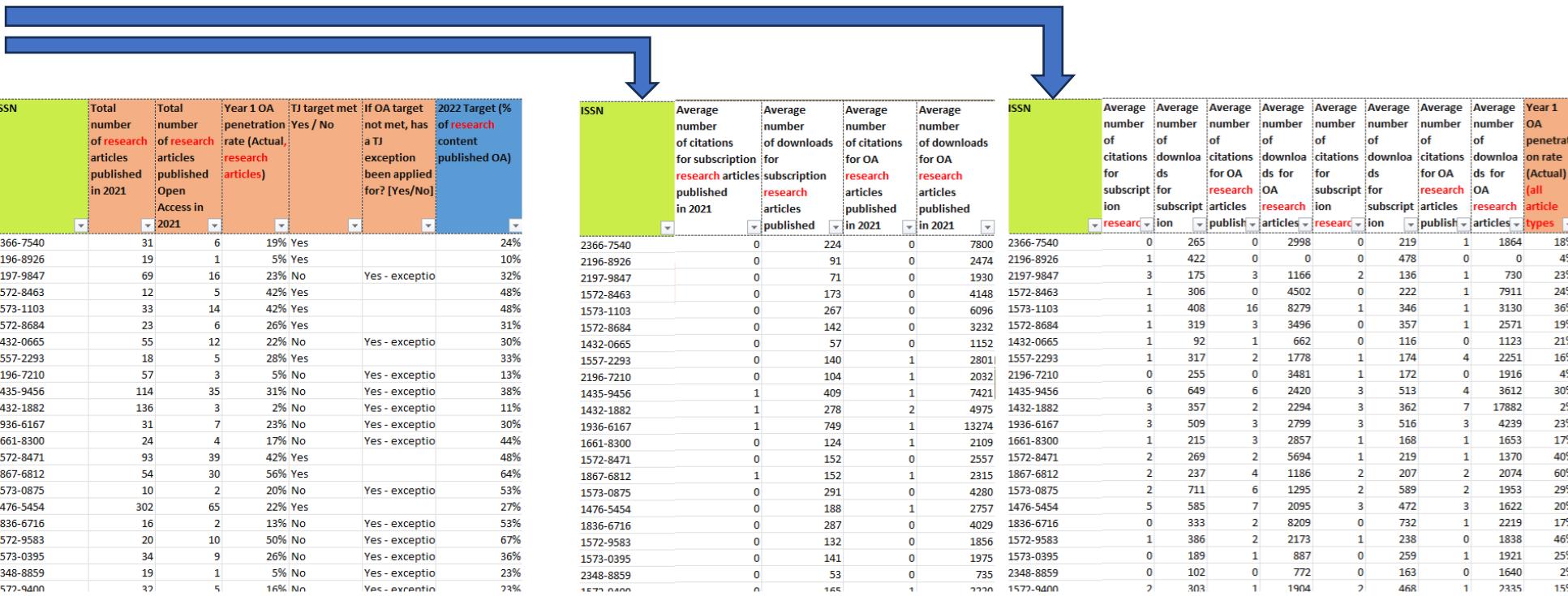
Send to info@coalition-s.org no later than 1 June 2022

URL to page demonstrating how institutions purchasing a subscription to a Transformative Journal will pay only for remaining subscription <https://www.springernature.com/g>

URL where data below is published: <https://www.springernature.com/g>

Name of journal	ISSN	2020 - Year 0 data					2021 - Year 1 data					2022 - Year 2 OA Target	Information on usage comparing subscription and OA content (2021 pubs)					ADDITIONAL DATA FIELDS PROVIDED BY SN						
		Total number of research articles published in 2020	Total number of research articles published Open Access in 2020	Year 0 OA penetration rate (research articles)	Year 1 target [As agreed previously]	Total number of research articles published in 2021	Total number of research articles published Open Access in 2021	Year 1 OA penetration rate (Actual research articles)	TJ target met Yes / No	If OA target not met, has a TJ exception been applied for? [Yes/No]	Average number of citations for subscription research articles published in 2021	Average number of downloads for subscription research articles published	Average number of citations for OA research articles published in 2021	Average number of downloads for OA research articles published in 2021	Comments, observations	Average number of citations for subscription research	Average number of downloads for subscription research	Average number of citations for OA research	Average number of downloads for OA research	Average number of citations for subscription research	Average number of downloads for subscription research	Average number of citations for OA research	Average number of downloads for OA research	Year 1 OA penetration rate (Actual)
Advances in Neurodevelopmental Disorders	2366-7540	40	4	10%	15%	31	6	19%	Yes		24%	0	224	0	7800	0	265	0	2998	0	219	1	1864	18%
The Analysis of Verbal Behavior	2196-8926	11	0	0%	5%	19	1	5%	Yes		10%	0	91	0	2474	1	422	0	0	0	478	0	0	4%
Research in the Mathematical Sciences	2197-9847	32	7	22%	27%	69	16	23%	No	Yes - exceptio	32%	0	71	0	1930	3	175	3	1166	2	136	1	730	23%
Foundations of Chemistry	1572-8463	28	8	29%	34%	12	5	42%	Yes		48%	0	173	0	4148	1	306	0	4502	0	222	1	7911	24%
Continental Philosophy Review	1573-1103	29	5	17%	22%	33	14	42%	Yes		48%	0	267	0	6096	1	408	16	8279	1	346	1	3130	36%
International Journal for Philosophy of Religion	1572-8684	29	5	17%	22%	23	6	26%	Yes		31%	0	142	0	3232	1	319	3	3496	0	357	1	2571	19%
Archive for Mathematical Logic	1432-0665	45	9	20%	25%	55	12	22%	No	Yes - exceptio	30%	0	57	0	1152	1	92	1	662	0	116	0	1123	21%
Journal of Maritime Archaeology	1557-2293	16	1	6%	11%	18	5	28%	Yes		33%	0	140	1	2801	1	317	2	1778	1	174	4	2251	16%
Transportation Infrastructure Geotechnology	2196-7210	39	1	3%	8%	57	3	5%	No	Yes - exceptio	13%	0	104	1	2032	0	255	0	3481	1	172	0	1916	4%
Animal Cognition	1435-9456	96	27	28%	33%	114	35	31%	No	Yes - exceptio	38%	1	409	1	7421	6	649	6	2420	3	513	4	3612	30%
Multimedia Systems	1432-1882	74	1	1%	6%	136	3	2%	No	Yes - exceptio	11%	1	278	2	4975	3	357	2	2294	3	362	7	17882	2%
Studies in Comparative International Development	1936-6167	20	4	20%	25%	31	7	23%	No	Yes - exceptio	30%	1	749	1	13274	3	509	3	2799	3	516	3	4239	23%
Logica Universalis	1661-8300	24	8	33%	38%	24	4	17%	No	Yes - exceptio	44%	0	124	1	2109	1	215	3	2857	1	168	1	1653	17%
Foundations of Science	1572-8471	68	22	32%	37%	93	39	42%	Yes		48%	0	152	0	2557	2	269	2	5694	1	219	1	1370	40%
PalZ	1867-6812	21	10	48%	55%	54	30	56%	Yes		64%	1	152	1	2315	2	237	4	1186	2	207	2	2074	60%
Origins of Life and Evolution of Biospheres	1573-0875	10	4	40%	46%	10	2	20%	No	Yes - exceptio	53%	0	291	0	4280	2	711	6	1295	2	589	2	1953	29%
Eye	1476-5454	288	34	12%	17%	302	65	22%	Yes		27%	0	188	1	2757	5	585	7	2095	3	472	3	1622	20%
Monash Bioethics Review	1836-6716	15	6	40%	46%	16	2	13%	No	Yes - exceptio	53%	0	287	0	4029	0	333	2	8209	0	732	1	2219	17%
Journal of Logic, Language and Information	1572-9583	10	5	50%	58%	20	10	50%	No	Yes - exceptio	67%	0	132	0	1856	1	386	2	2173	1	238	0	1838	46%
Journal of Indian Philosophy	1573-0395	34	9	26%	31%	34	9	26%	No	Yes - exceptio	36%	0	141	0	1975	0	189	1	887	0	259	1	1921	25%
Journal of Fetal Medicine	2348-8859	24	3	13%	18%	19	1	5%	No	Yes - exceptio	23%	0	53	0	735	0	102	0	772	0	163	0	1640	2%
Health Services and Outcomes Research Methodology	1572-9400	24	3	13%	18%	32	5	16%	No	Yes - exceptio	23%	0	165	1	2220	2	303	1	1904	2	468	1	2335	15%
Linguistics and Philosophy	1573-0549	22	8	36%	42%	28	10	36%	No	Yes - exceptio	48%	0	183	0	2431	1	248	2	652	1	273	2	3674	36%

Relations



The diagram illustrates the relationship between three data sources:

- Table 1 (Left):** Journal Metrics for 2021. It includes columns for ISSN, Total number of research articles published in 2021, Total number of research articles published Open Access in 2021, Year 1 OA penetration rate (Actual research articles), TJ target met Yes / No, If OA target not met, has a TJ exception been applied for? [Yes/No], and 2022 Target (% of research content published OA).
- Table 2 (Middle):** Average citations and downloads for 2021 research articles. It includes columns for ISSN, Average number of citations for subscription research articles published in 2021, Average number of downloads for subscription research articles published in 2021, Average number of citations for OA research articles published in 2021, and Average number of downloads for OA research articles published in 2021.
- Table 3 (Right):** Detailed metrics for 2021 research articles. It includes columns for ISSN, Average number of citations for subscription research articles published in 2021, Average number of downloads for subscription research articles published in 2021, Average number of citations for OA research articles published in 2021, and Average number of downloads for OA research articles published in 2021.

Relations

- To answer queries like:
 - “Tell me a Commander that can pilot Space_Shuttle Falcon”.
- We need to **relate** entity sets logically.
- A **Relation** is a logical model for associating two or more **Entities**.
 - *Commander Turker can pilot Falcon.*
- A **Relationship set** is the set of all these relations.

Key concepts of ER: Relation

- Relating two or more entity sets requires the designer to set a **mapping between the entities belonging the Entity Sets under Relation.**

Can more than one mechanic repair the same car?
Can a mechanic repair more than one car?

We will see...

Model Entity set

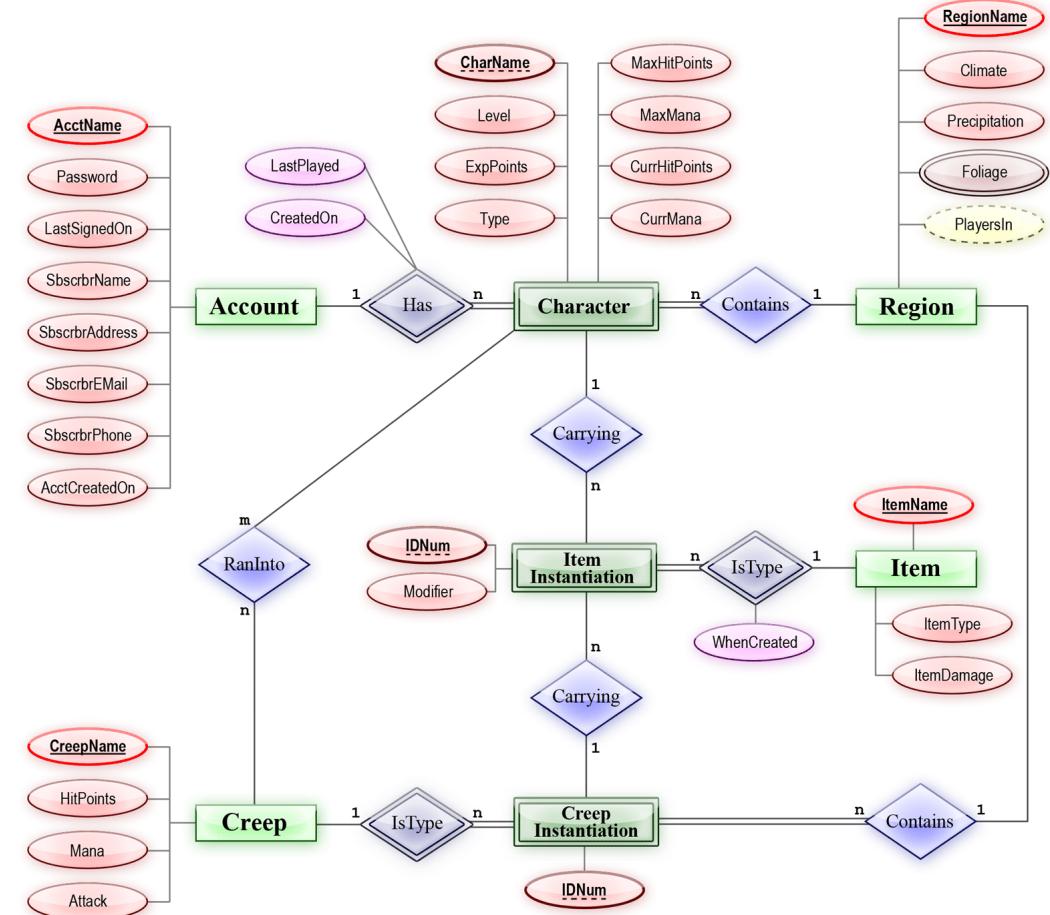
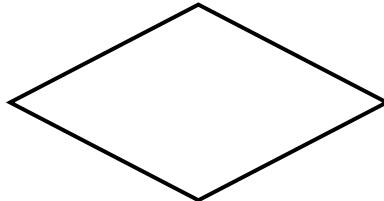
Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

Mechanic Entity set

NIN	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

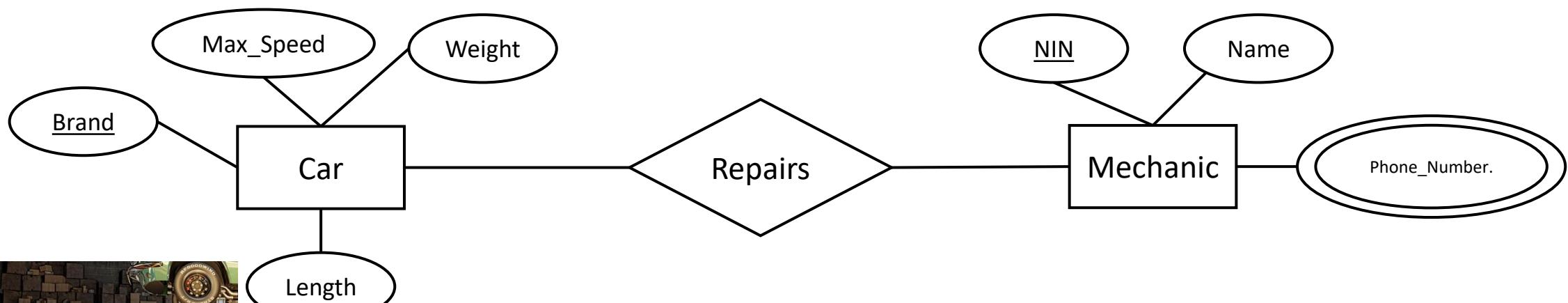
Key concepts of ER: Relation

A diamond represents a **Relationship set**.

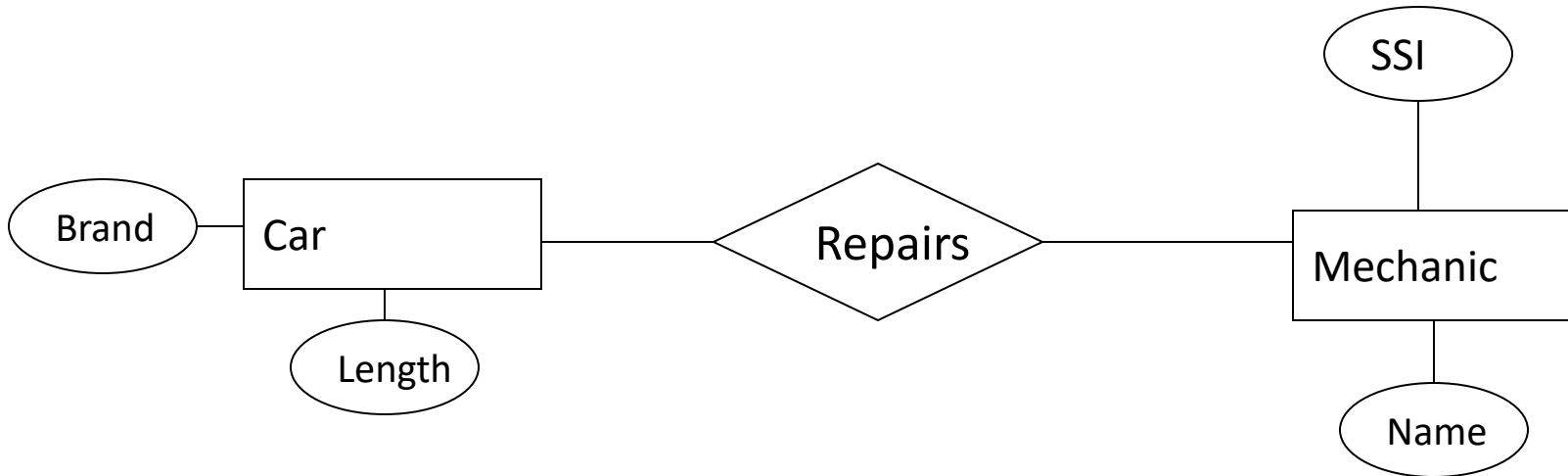


Key concepts of ER: Relation

A relationship set 'Repairs' relates two Entity Sets.

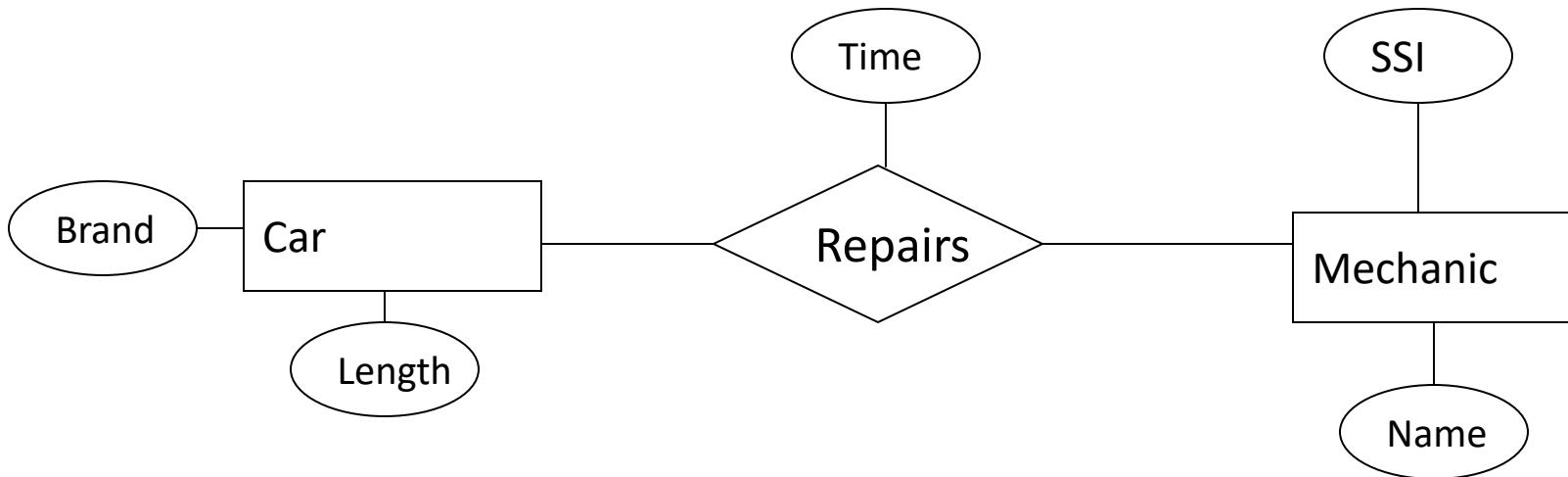


ER Model



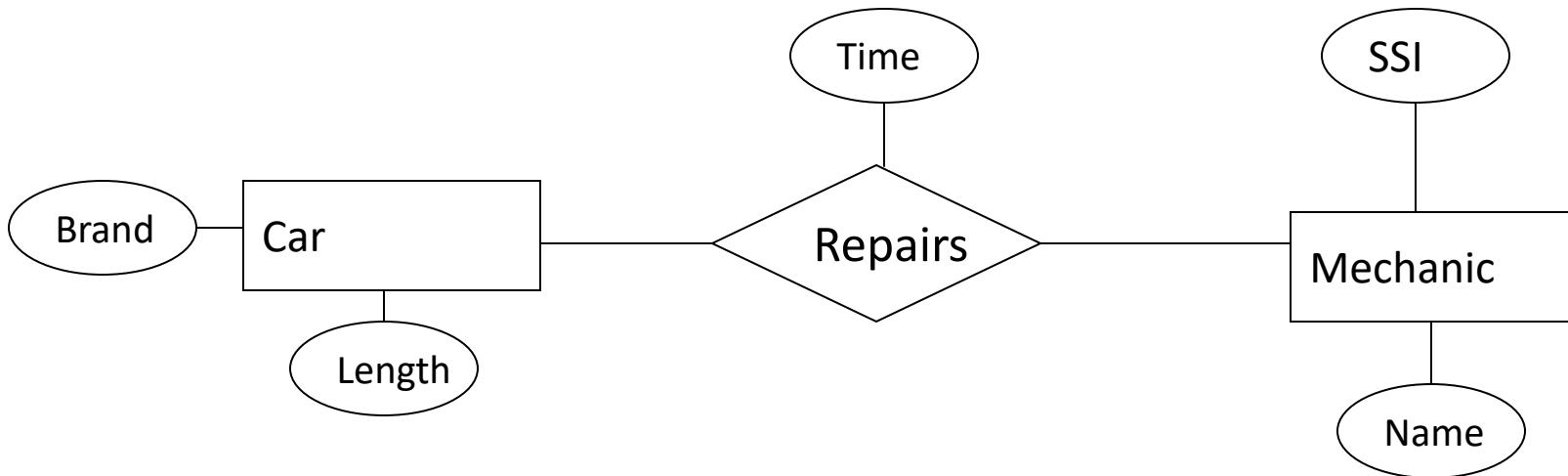
- Relationship sets may also have attributes.

ER Model



- Relationship sets may also have attributes.
- We will talk about the key in a relationship set later.

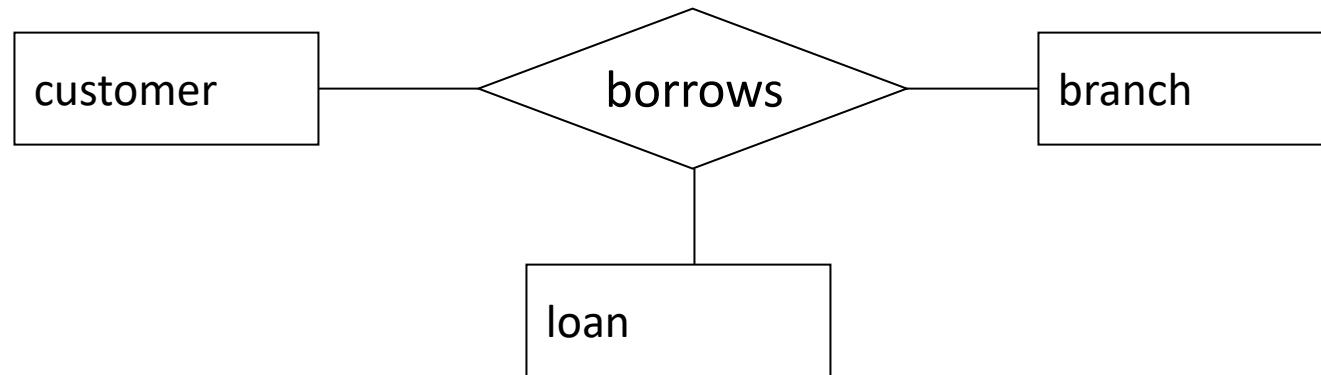
ER Model



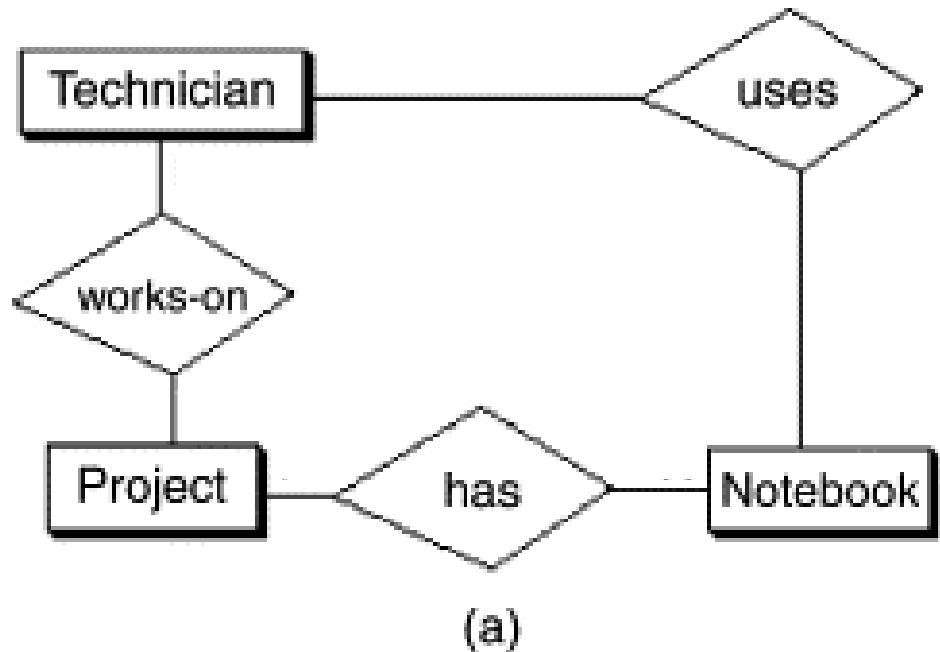
- **Degree of a relationship set** is the number of different entity sets that participate in a relationship
- **Binary relationship** sets involve two entity sets



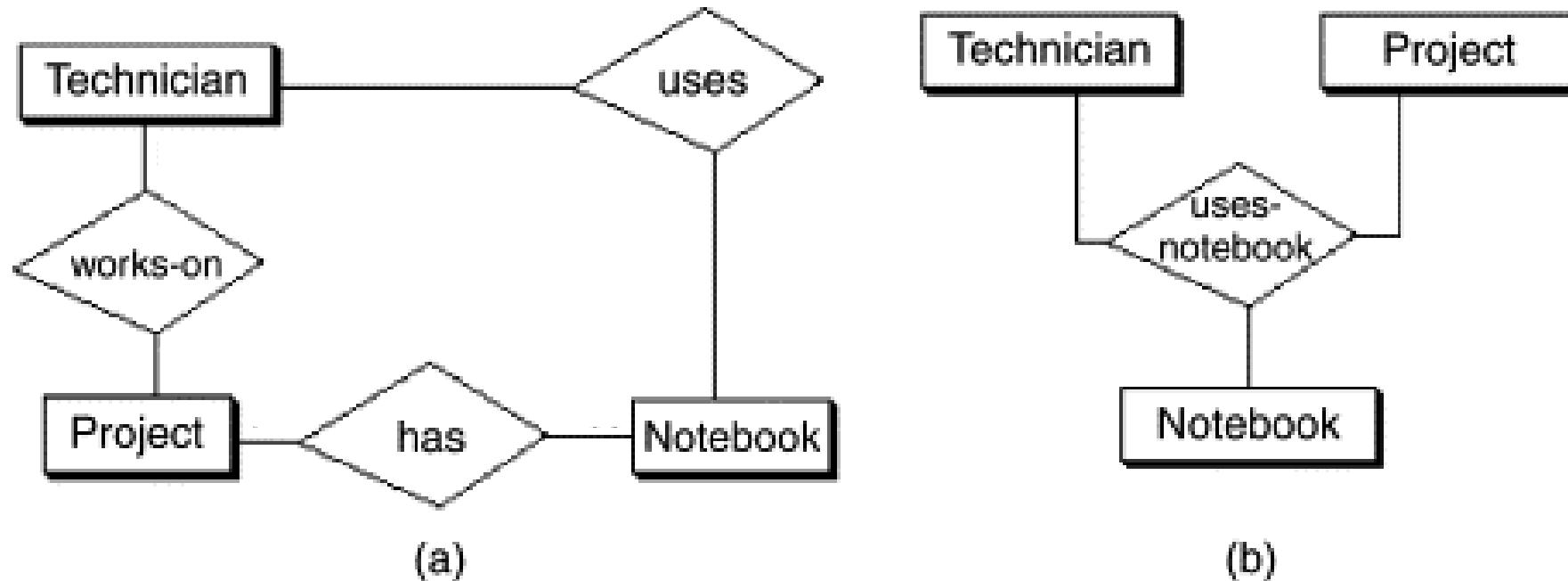
- Ternary relationship sets involve three entity sets.



Ternary relationship sets.



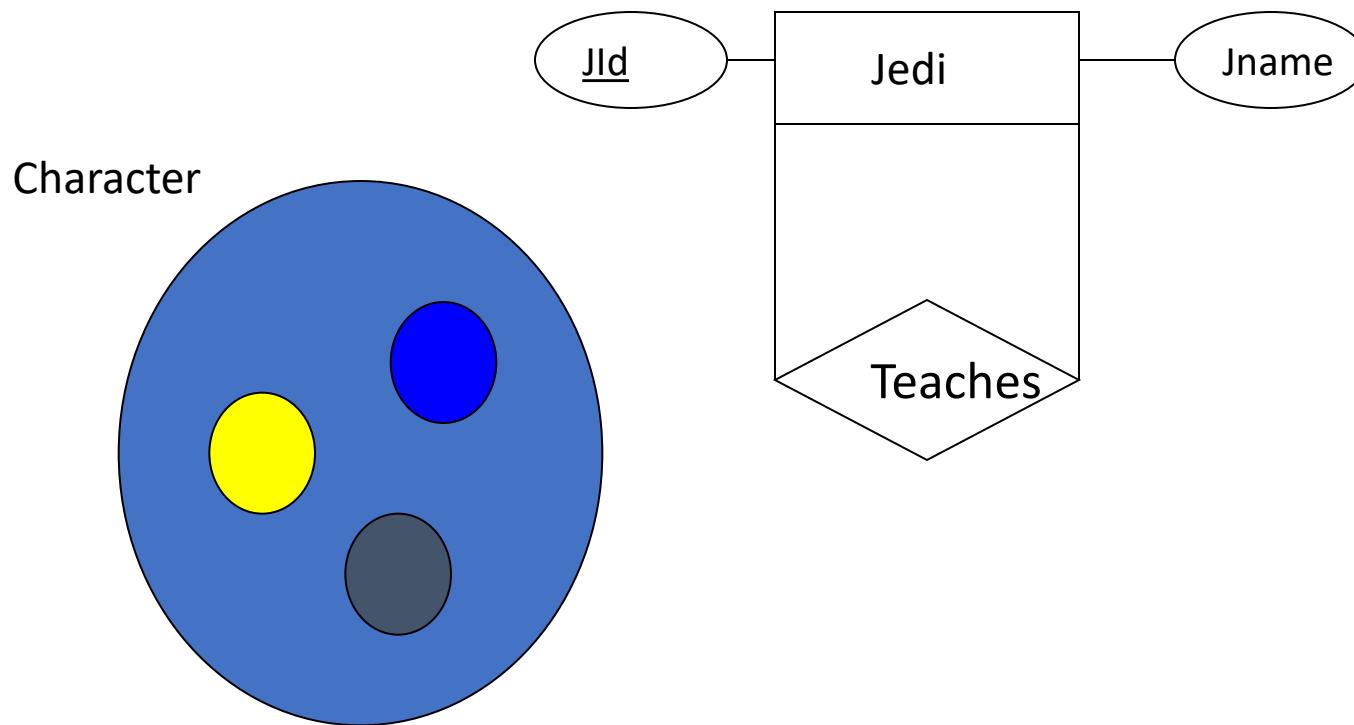
Ternary relationship sets.



ER Model



- We may have relationships among the entities that belong to the same entity set
- each entity has a role in such a relationship

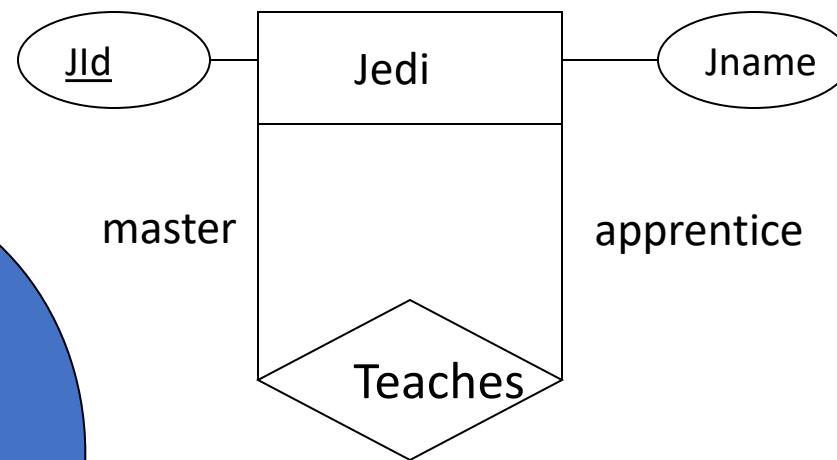
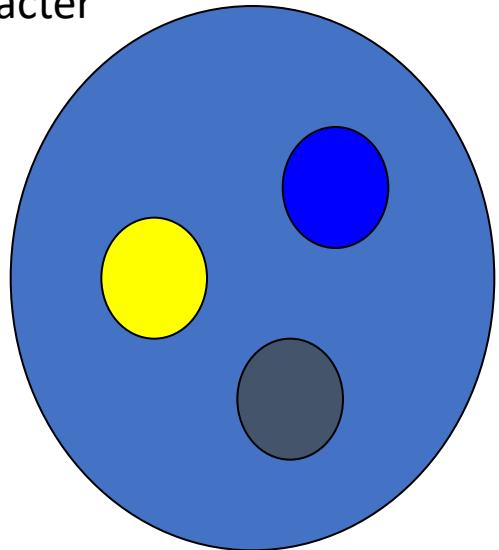


ER Model



- We may have relationships among the entities that belong to the same entity set
- **each entity has a role in such a relationship.**

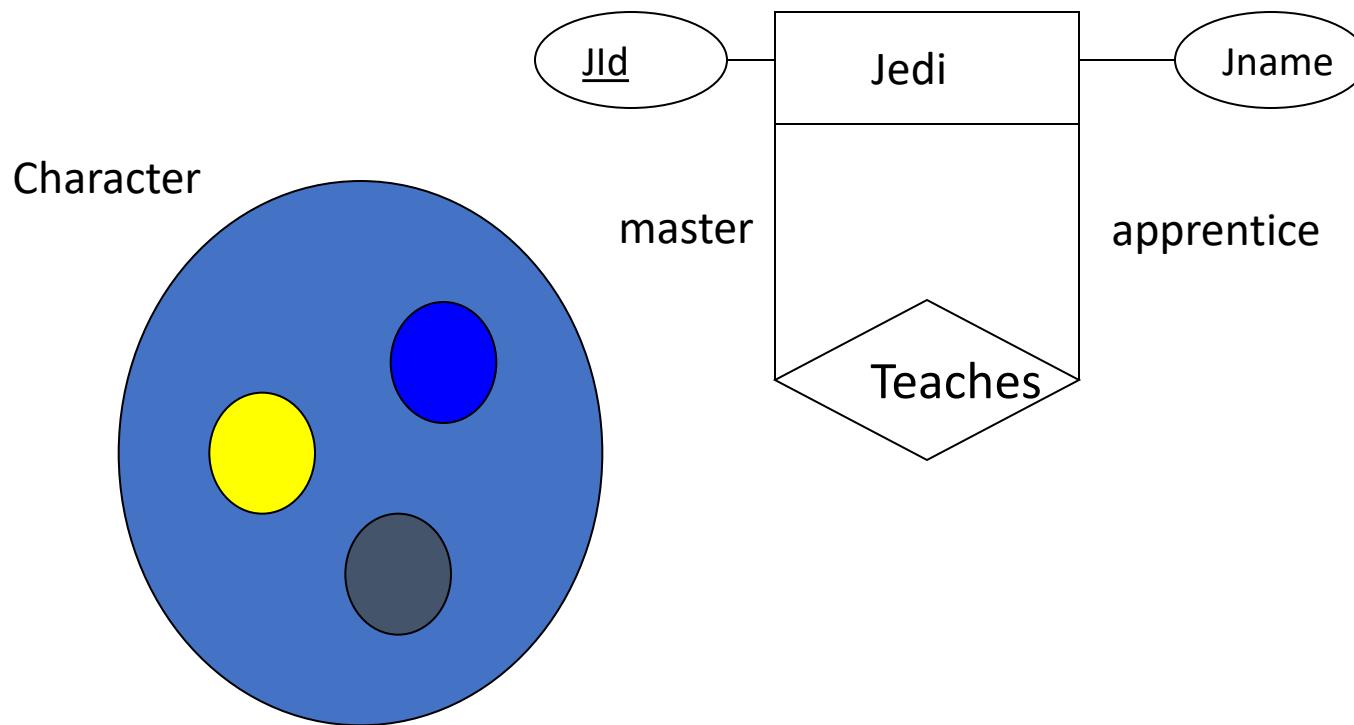
Character



ER Model



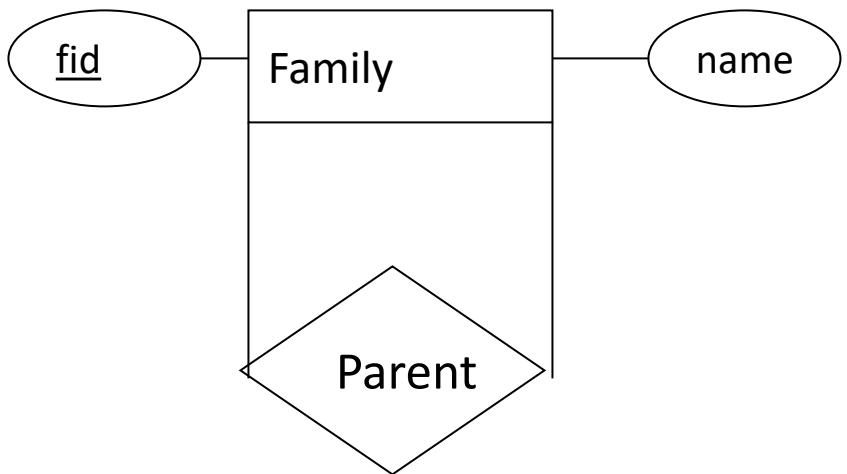
- We may have relationships among the entities that belong to the same entity set
- each entity has a role in such a relationship.
- **What is the degree of the following relationship set (unary, binary, or ternary)?**



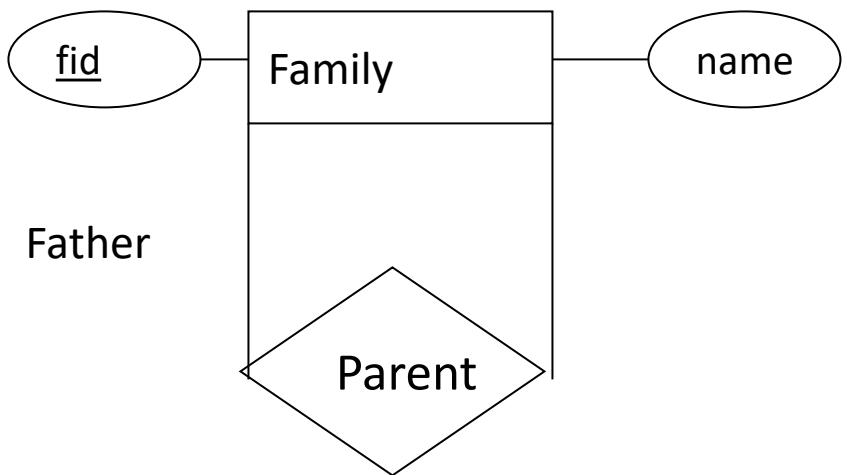
ER Model



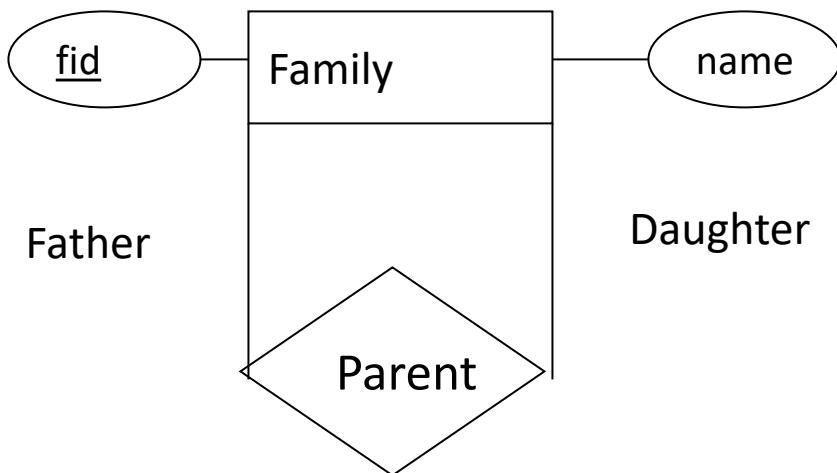
ER Model



ER Model

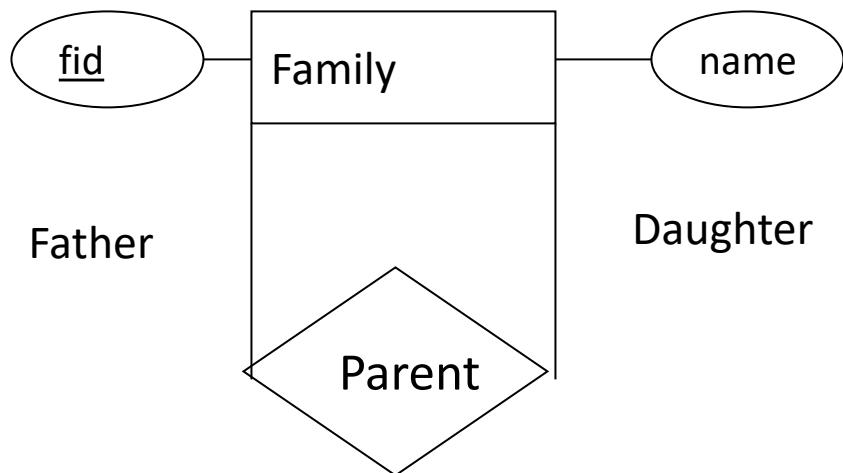


ER Model



ER Model

Fid	Name	Name
12	Mr Bean	N/A
21	Thomas Shelby	Roby Shelby
55	Thanos	Gamora



Key concepts of ER: Cardinalities.



VERY IMPORTANT CONTENT!!!!

Model Entity set

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

Mechanic Entity set

NIN	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

Given an entity (E) from one entity set, what is the relation of this entity with the entities in the other entity sets?

Can more than one mechanic repair BMW 3.21?

Can Tom repair more than one type of car?

Key concepts of ER: Cardinalities.



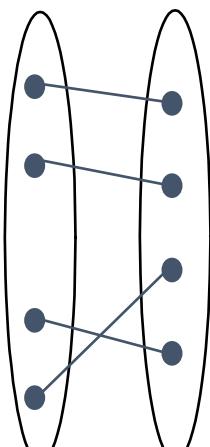
<u>Brand</u>	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

<u>NIN</u>	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237



Mapping cardinalities

- Recall: Relating two or more entity sets requires the designer to set a **mapping between the entities belonging the Entity Sets under Relation**.
- One-to-One relationship (ex: marriage relationship set between husbands and wives)



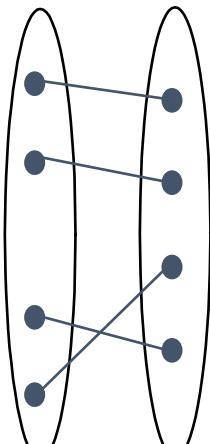
1-to-1

SSI	Name	Phone_Number	Brand	Weight	Length	Max_Speed
87542702	Tom	75315567, 75315264	BMW 3.21	1400	3.21	200
68201937	Uraz	75335521, 75334567	Toyota_Corolla	1300	3.18	200
23139827	Nick	75315544, 75315237	Hyundai_E.GLS	1400	3.16	210

- Relationship sets are created while creating tables in the relationship.
- This sometimes leads to new tables (as in this case) or leads to the addition of new attributes in the tables in the relationship (you will see)

Mapping cardinalities

- Recall: Relating two or more entity sets requires the designer to set a **mapping between the entities belonging the Entity Sets under Relation**.
- One-to-One relationship (ex: marriage relationship set between husbands and wives)



1-to-1

SSI	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai_E.GLS	1400	3.16	210

Relationship Set

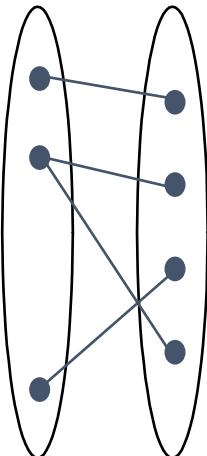
Name	Brand
Tom	Toyota..
Nick	Hyundai..
Uraz	BMW..

Mapping cardinalities

- One-to-One (ex: marriage relationship set between husbands and wives)
- One-to-Many (ex?)

Name	Brand
Uraz	BMW
Uraz	Hyundai
Tom	Toyota

Relationship Set



1-to Many

NIN	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

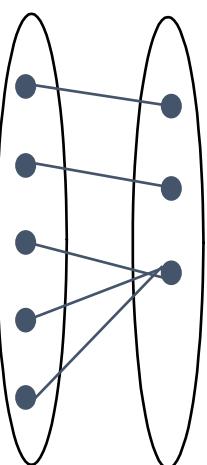
Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

Mapping cardinalities

- One-to-One (ex: marriage relationship set between husbands and wives)
- Many-to-One (ex?)

Name	Brand
Uraz	BMW
Nick	Toyota
Tom	BMW

Relationship Set



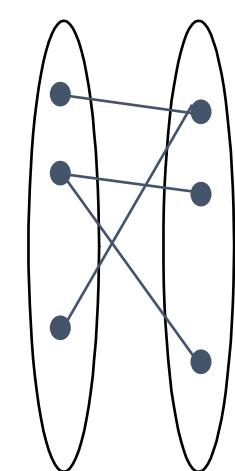
NIN	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

Many-to-1

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

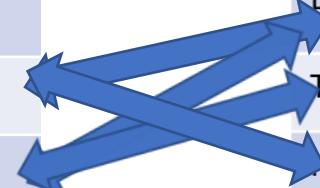
Mapping cardinalities

- One-to-One (ex: marriage relationship set between husbands and wives)
- Many-to-Many (ex?)



NIN	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

Many-to-Many



Name	Brand
Uraz	BMW
Uraz	Hyundai
Nick	Toyota
Nick	BMW

Relationship Set

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210



Welcome to WEEK 2....

We will start at 17:00...



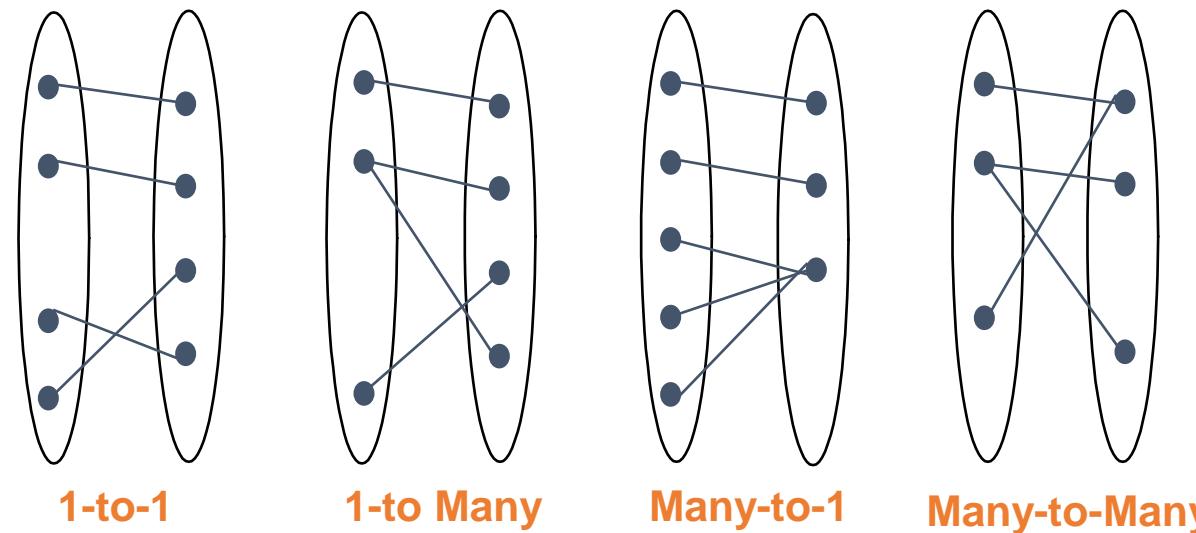
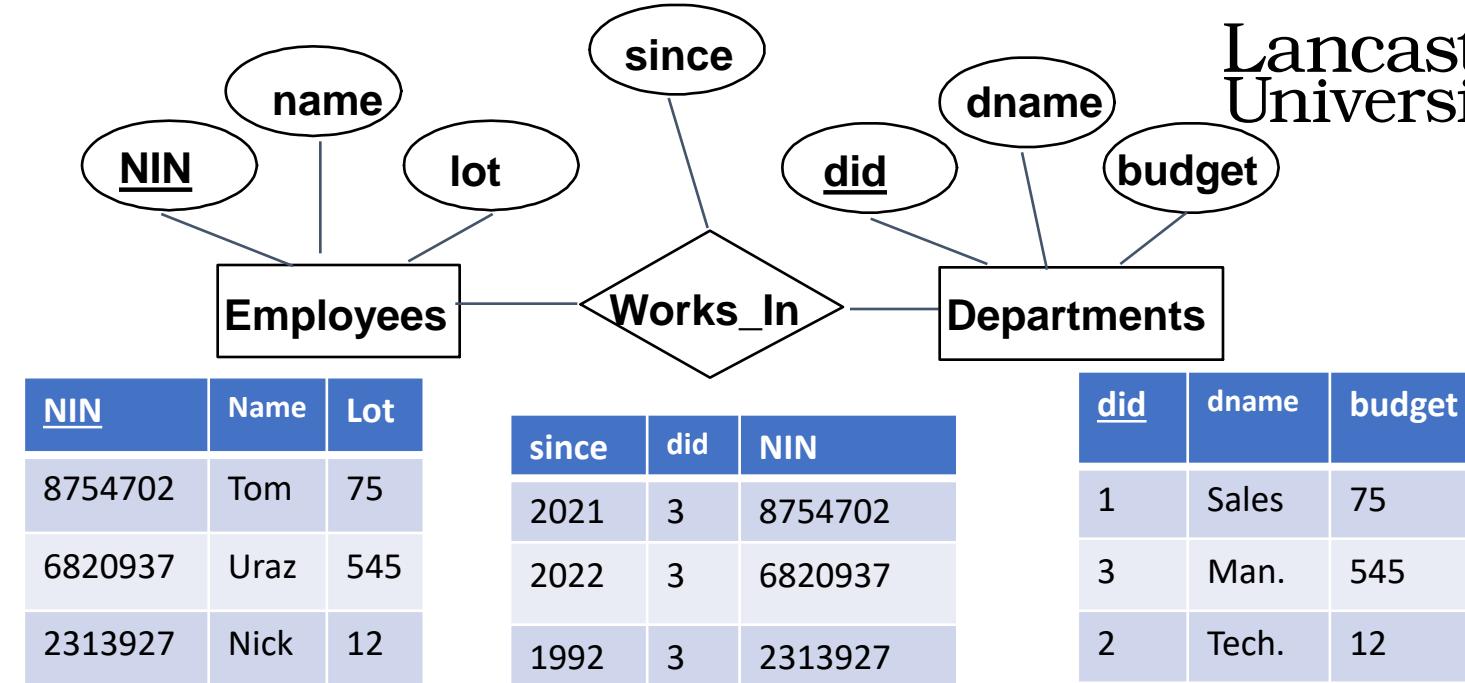
FROM YOU...

What did we do in last lecture?

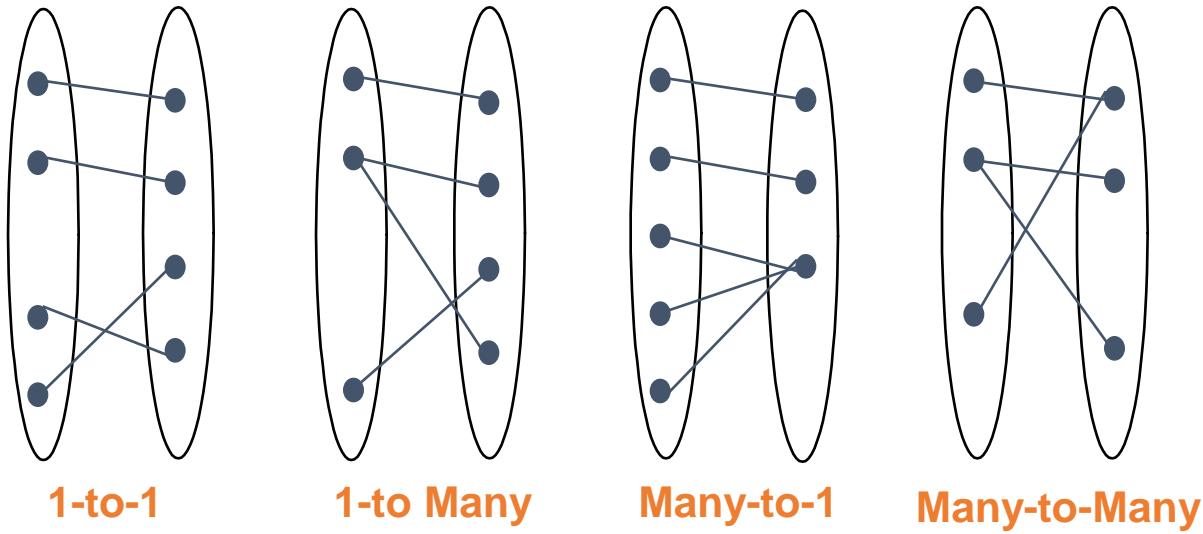
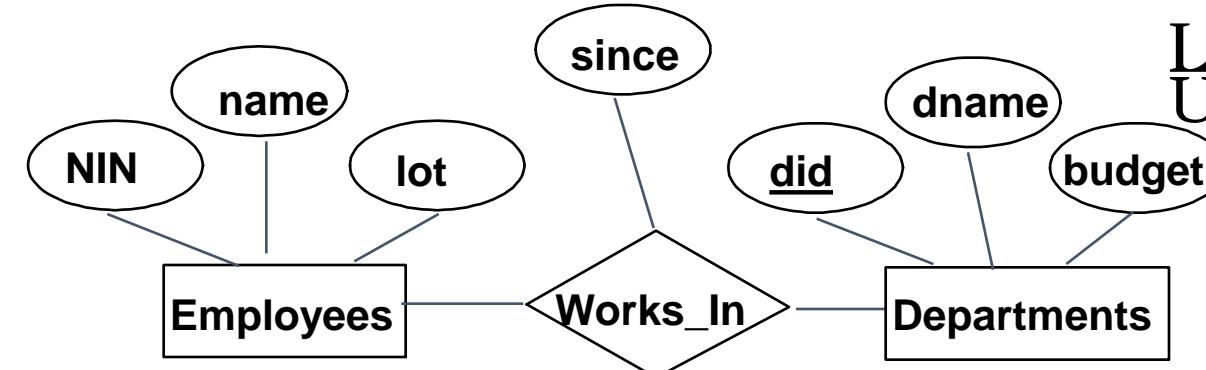


- Entities
- Relating entities: *Relations*
- Mapping between entities
 - 1 to 1 relations.
 - 1 to many relations.
 - Many to one relations.
 - Many to many relations.

- Consider the `works_in` relationship
 - If an employee can work in at most one department and a department can have multiple employees
 - What type of relationship is that?
-



- Consider the `works_in` relationship
 - If an employee can work in several departments and a department can have multiple employees
 - What type of relationship is that?
-

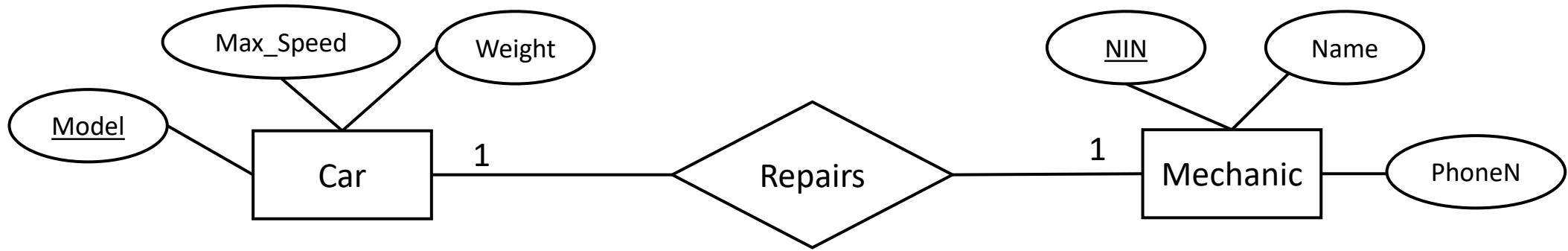


How do we encode cardinality into ER diagrams?

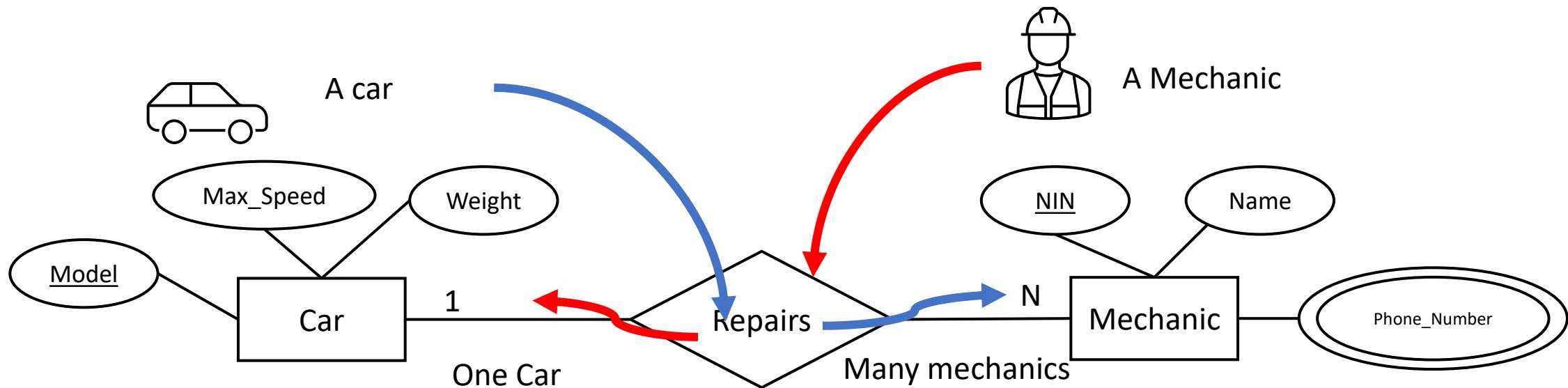


- We use **Chen's notation** (Always look at the opposite direction)
- 1:1 is for one-to-one
- 1:N is for one-to-many
- N:1 is for many-to-one
- N:M is for many-to-many.

1:1



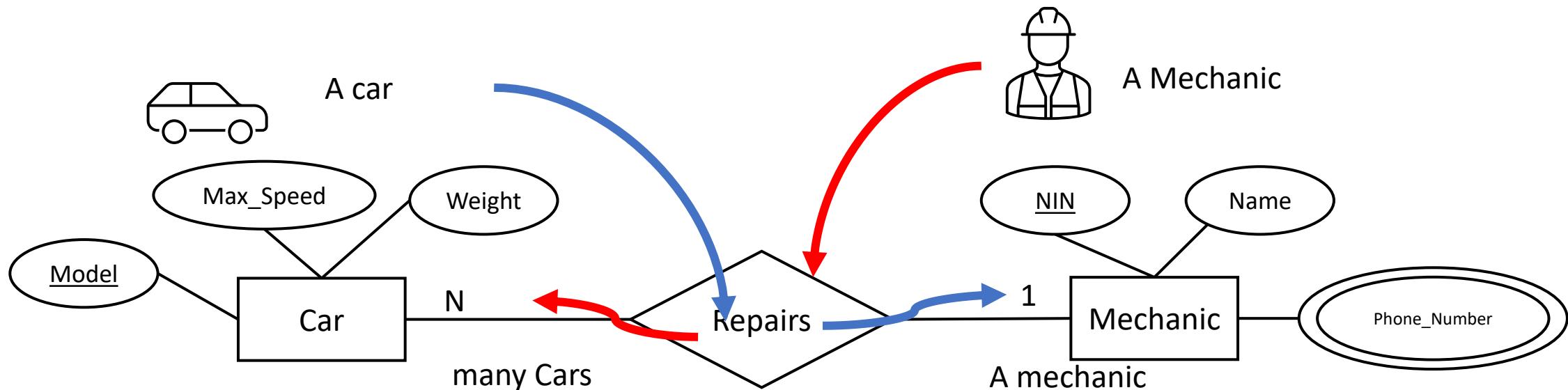
1:N One to many



A car can be repaired by many mechanics.
A mechanic can repair one car.



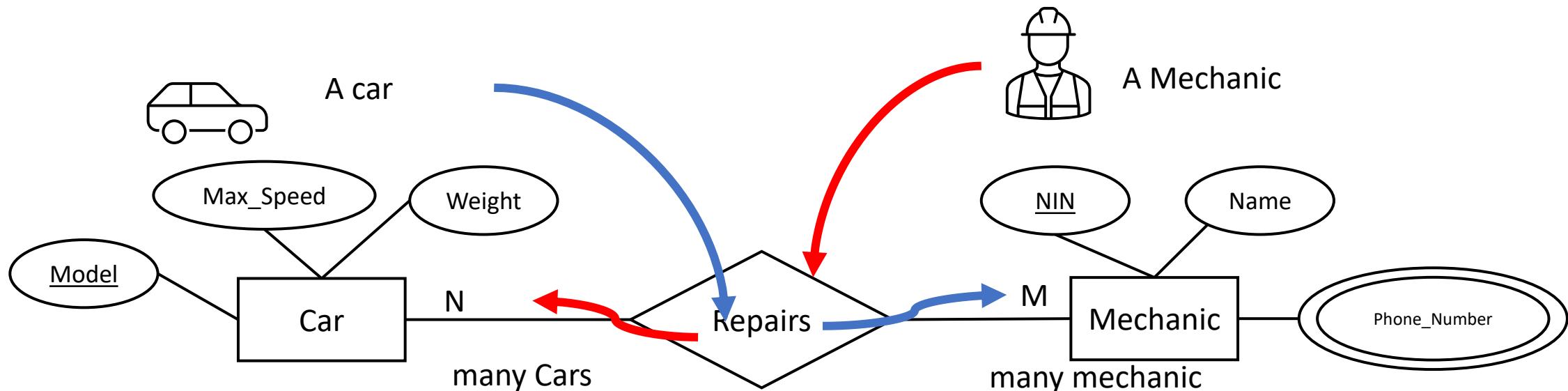
N:1 Many to one



A car can be repaired by a mechanic.
A mechanic can repair many cars.



N:N many to many



A car can be repaired by many mechanics.
A mechanic can repair many cars.



Key concepts of ER: Cardinalities.



IMPORTANT CONTENT!!

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

NIN	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

Given an entity (E) from one entity set, what is the relation of this entity with the entities in the other entity sets?

Can more than one mechanic repair BMW 3.21?

Can Tom repair more than one type of car?

Key concepts of ER: Participation Constraints.



IMPORTANT CONTENT!!

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

NIN	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

Given an entity (E) from one entity set, what is the relation of this entity with the entities in the other entity sets?

Can more than one mechanic repair BMW 3.21?

Can Tom repair more than one type of car?

Can there be a mechanic who does not know how to repair a car?

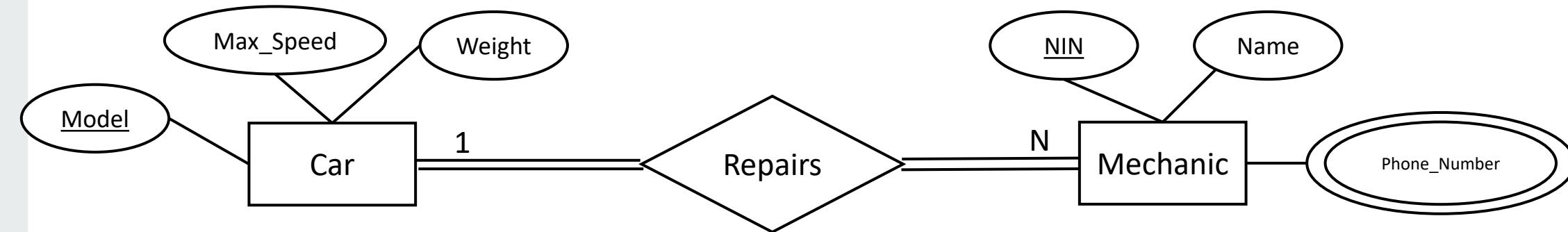
Can there be a car that cannot be repaired?

Participation constraints

- Can there be a mechanic that cannot repair a car?
- If not, we need to state that there is a **Total Participation**.
 - Total Participation implies that if an entity exists in an entity set, it must relate with at least one entity in the other entity set.
 - A **double line** identifies total participation.
- If so, then it is **Partial Participation**
 - A **single line** identifies partial participation.

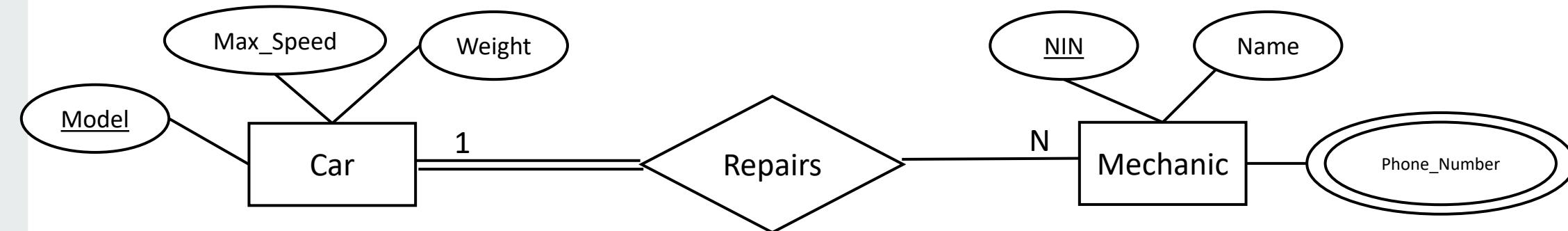
Participation constraints

- For each car, there **must be at least one** mechanic.
- Each mechanic must repair **exactly one** car.



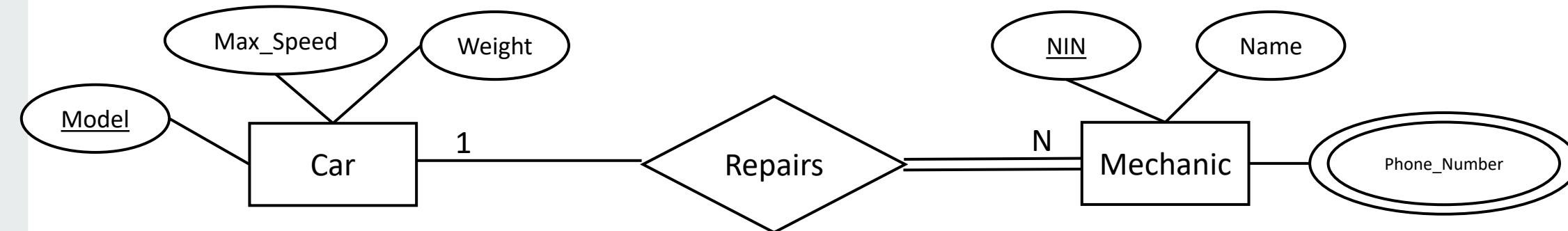
Participation constraints

- For each car, there must be at least one mechanic.
- Each mechanic repairs **at most** one car.



Participation constraints

- For each car, there **may be several mechanics**.
- Each mechanic must repair exactly one car.





Exercise...

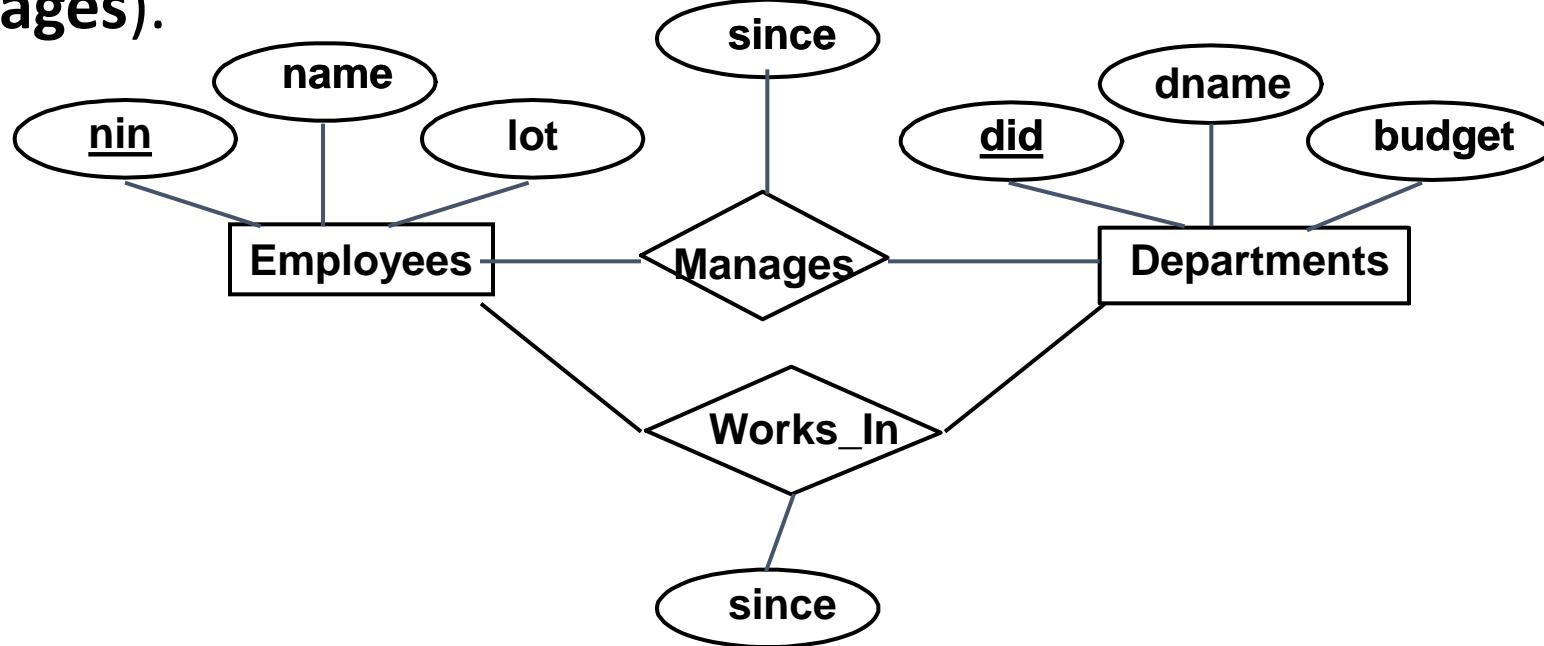
<https://forms.office.com/e/5NcDL8BN86?origin=lprLink>



Let's fill this.



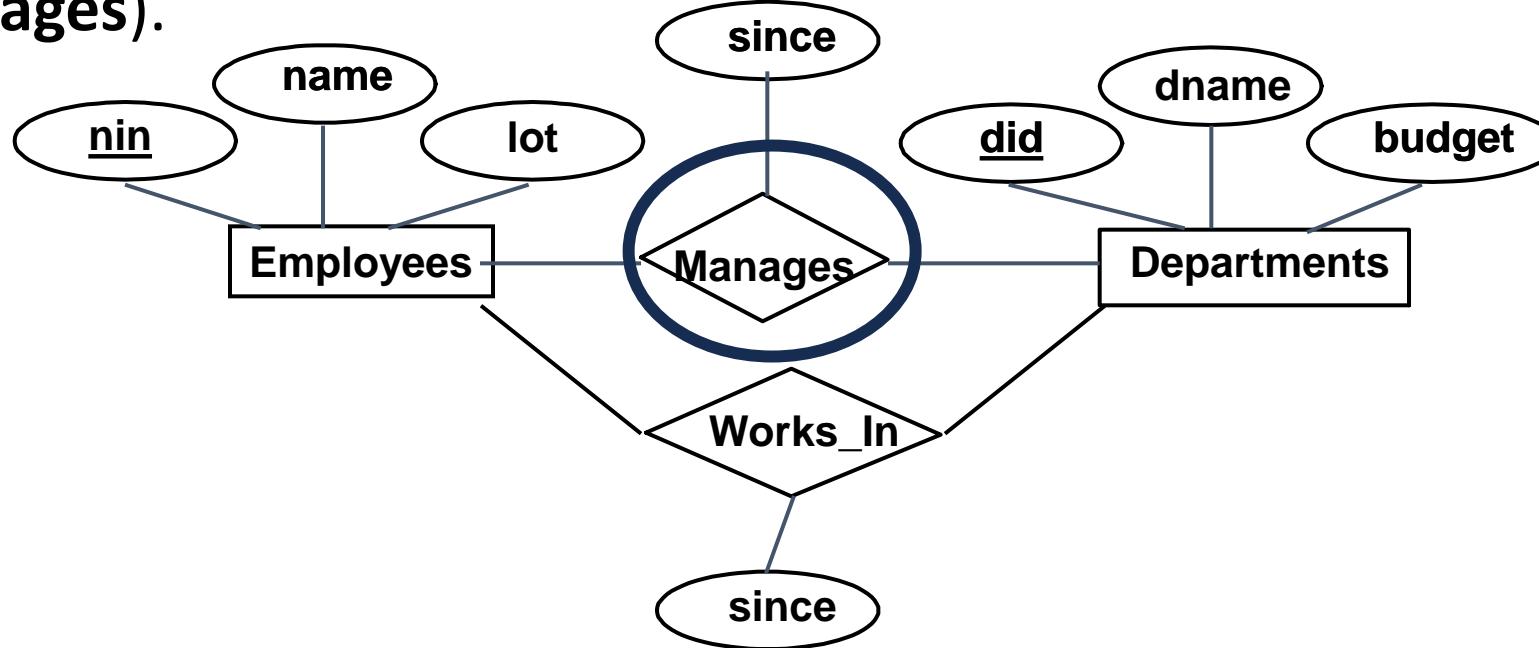
- Consider the **cardinality** between Employees and Departments (**Manages**).



Let's fill this.



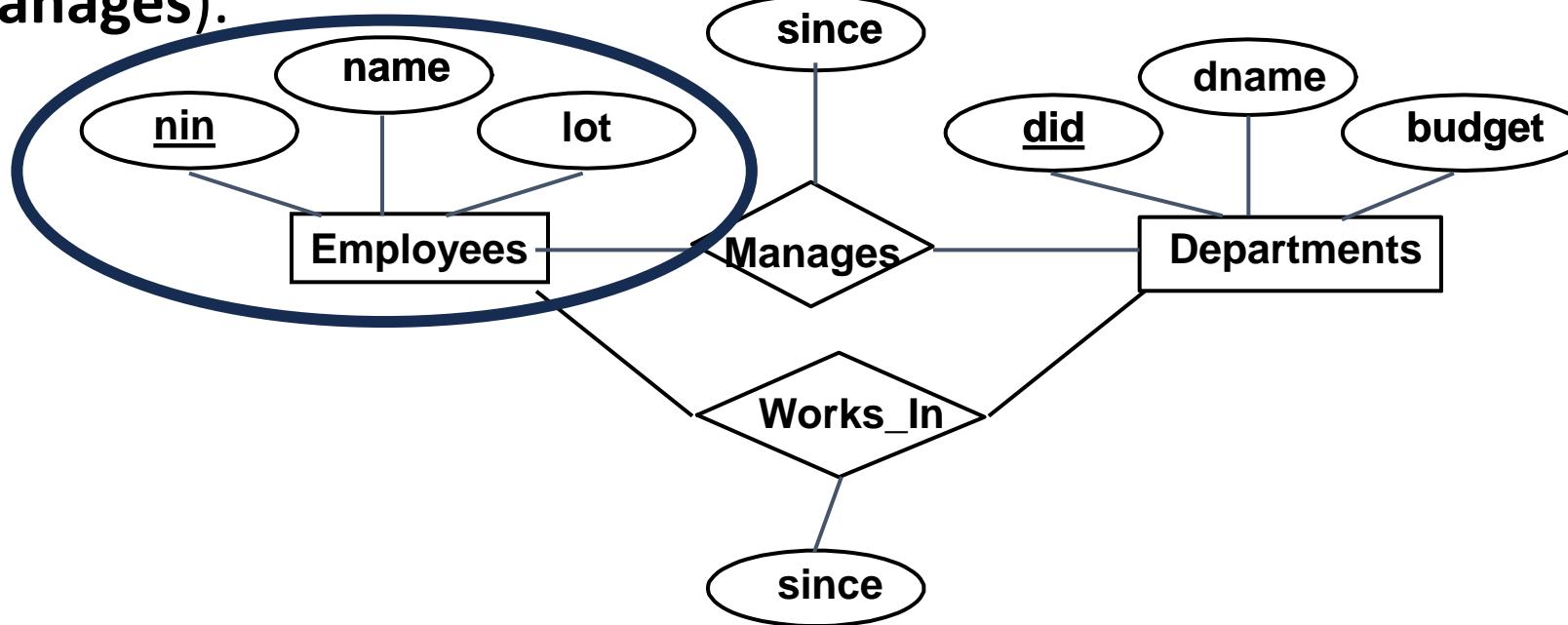
- Consider the **cardinality** between Employees and Departments (**Manages**).



Let's fill this.



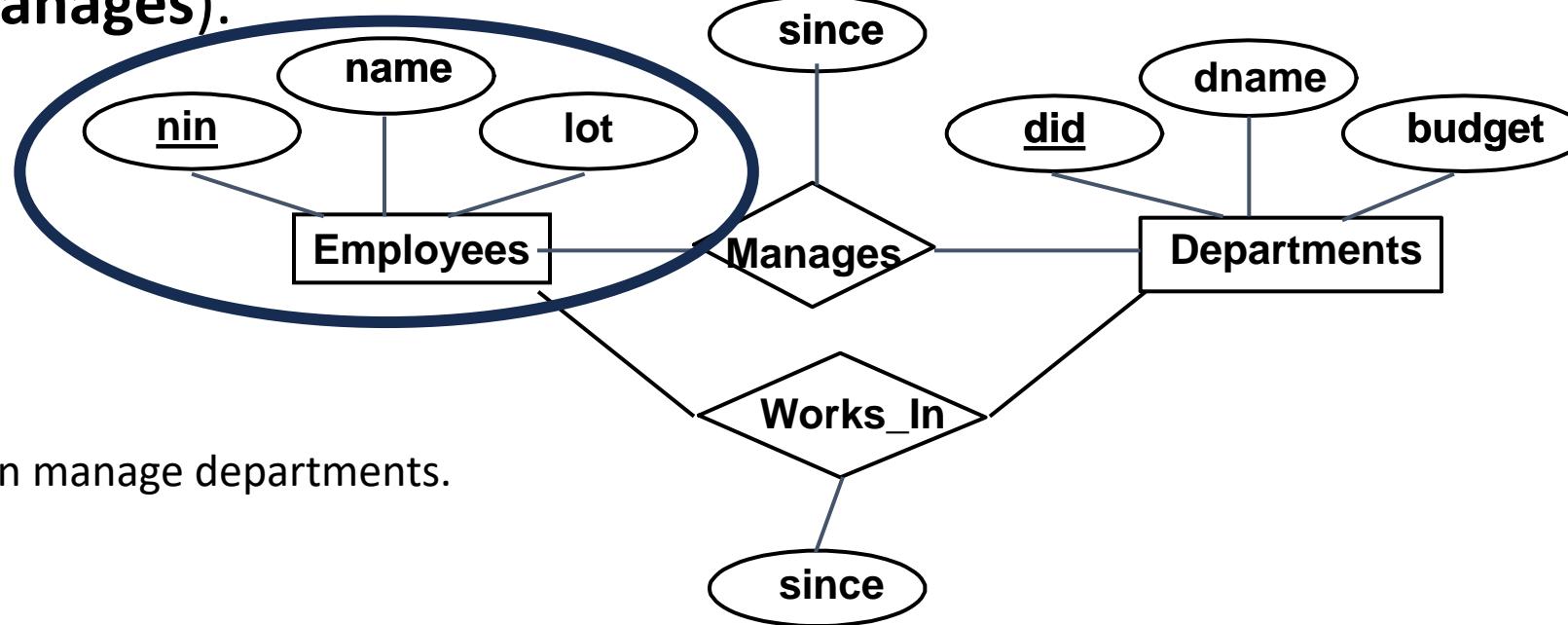
- Consider the **cardinality** between Employees and Departments (**Manages**).



Let's fill this.



- Consider the **cardinality** between Employees and Departments (**Manages**).

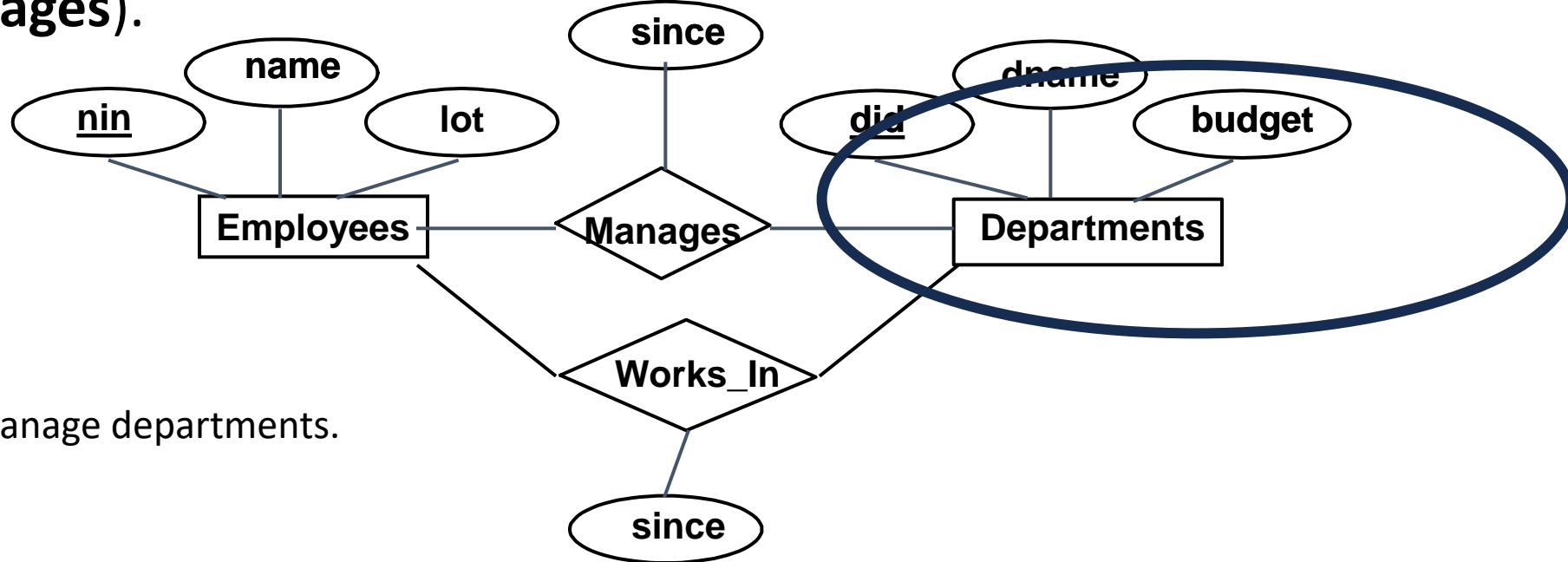


An employee can manage departments.

Let's fill this.



- Consider the **cardinality** between Employees and Departments (**Manages**).

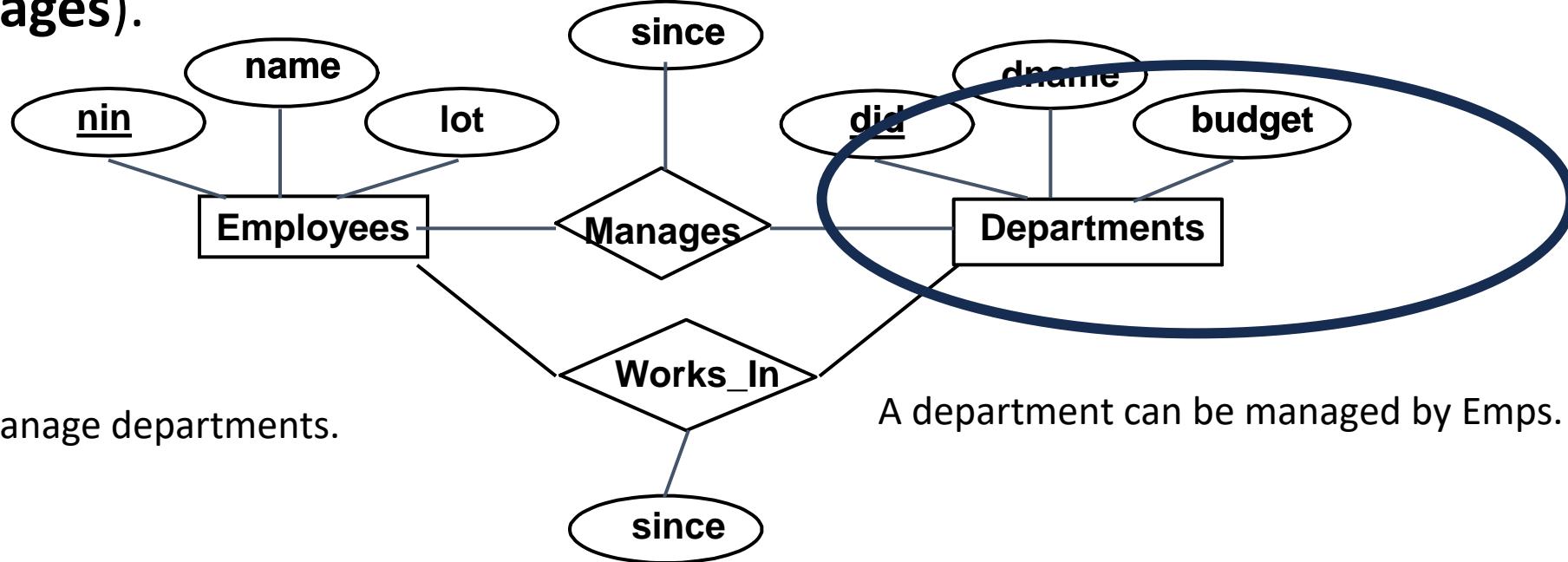


An employee can manage departments.

Let's fill this.



- Consider the **cardinality** between Employees and Departments (**Manages**).



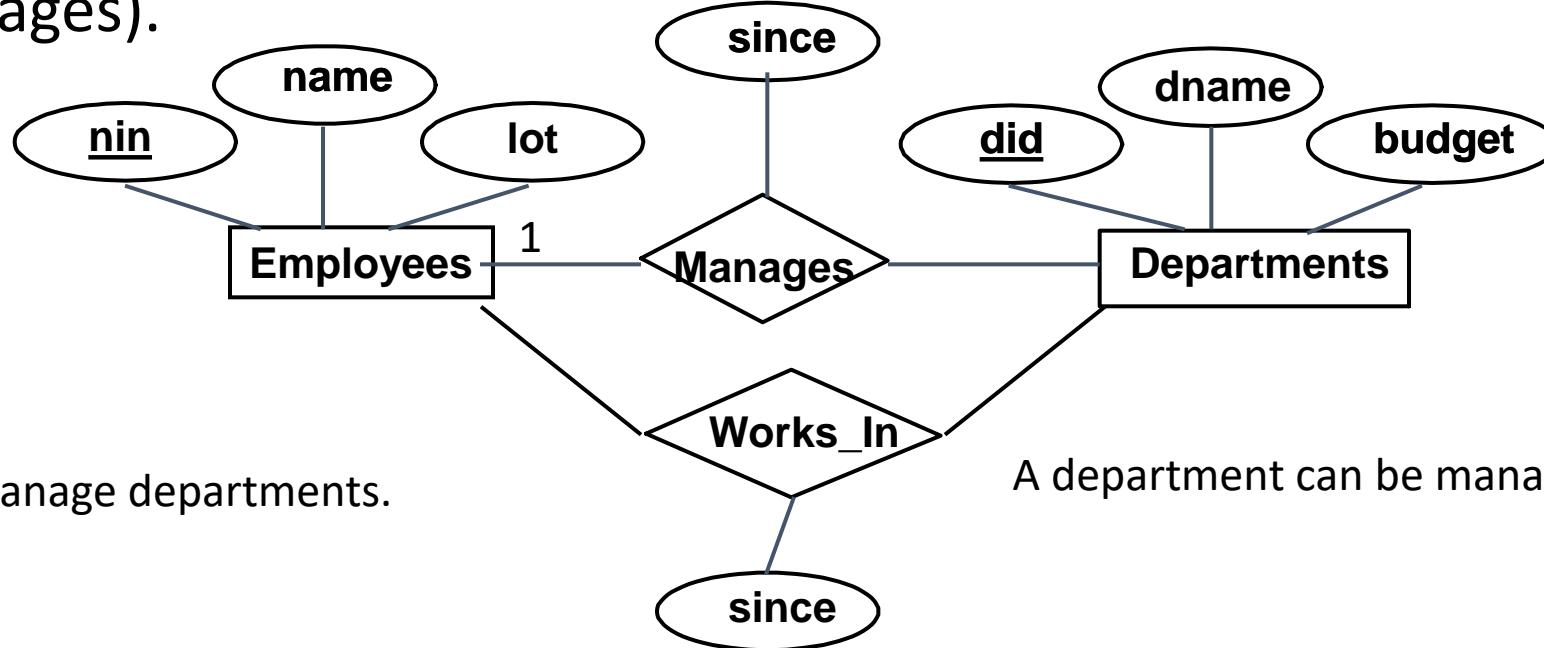
An employee can manage departments.

A department can be managed by Emps.

Let's fill this.



- Consider the **cardinality** between Employees and Departments (Manages).



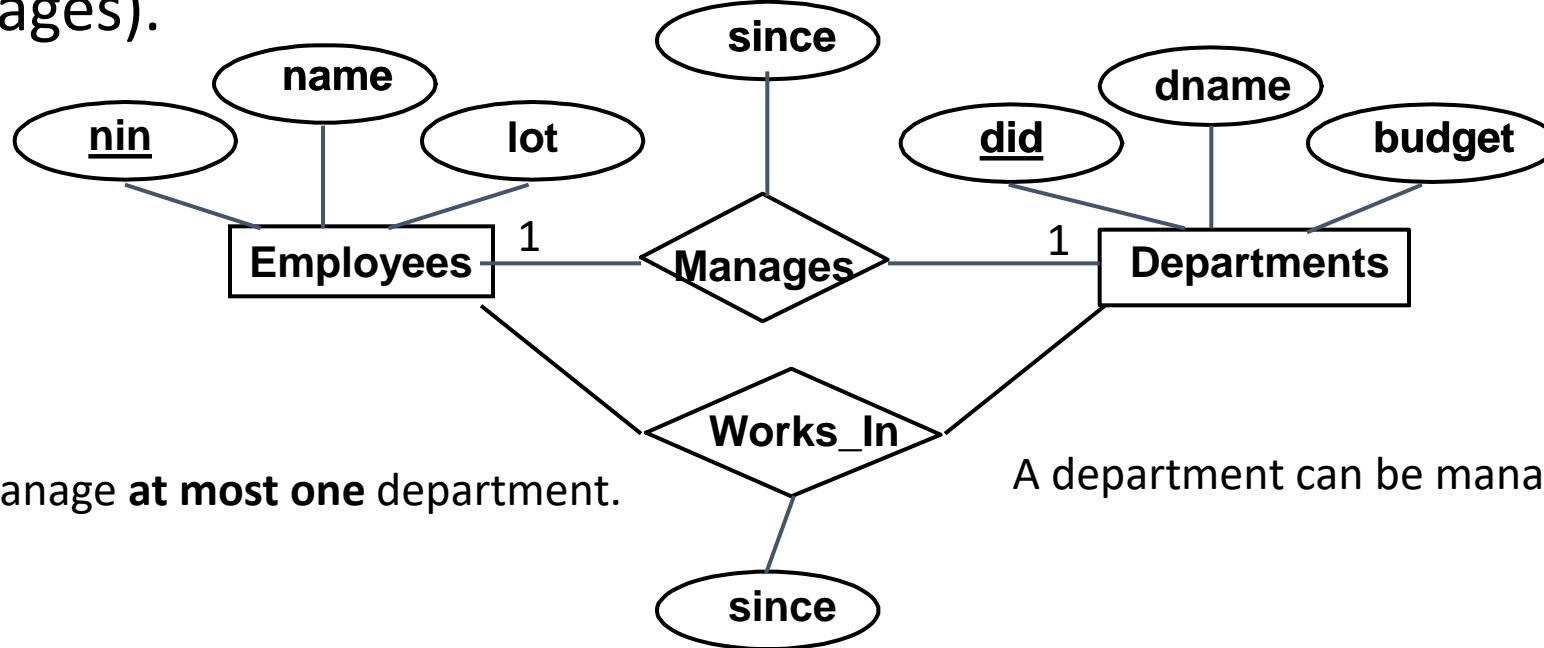
An employee can manage departments.

A department can be managed by **at most one** Emp.

Let's fill this.



- Consider the **cardinality** between Employees and Departments (Manages).



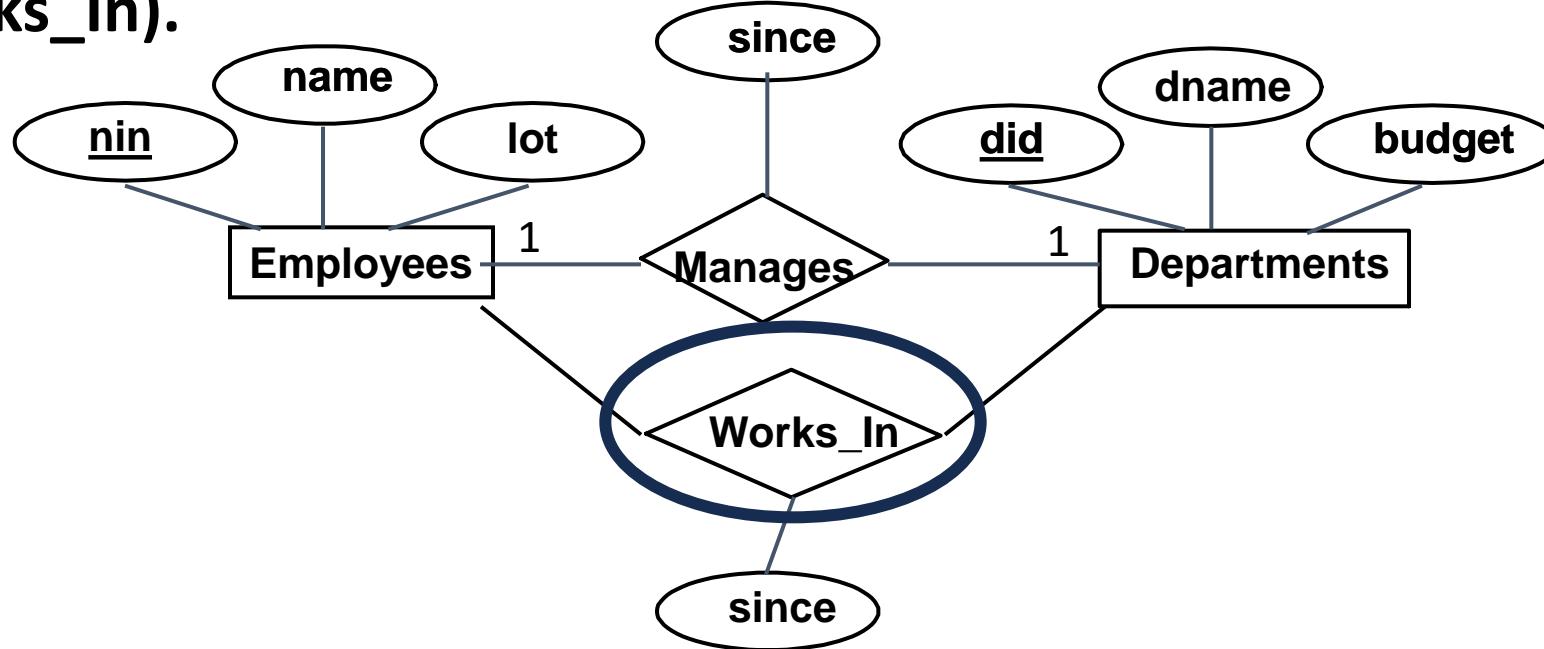
An employee can manage at **most one** department.

A department can be managed by zero or one Emps.

Let's fill this.



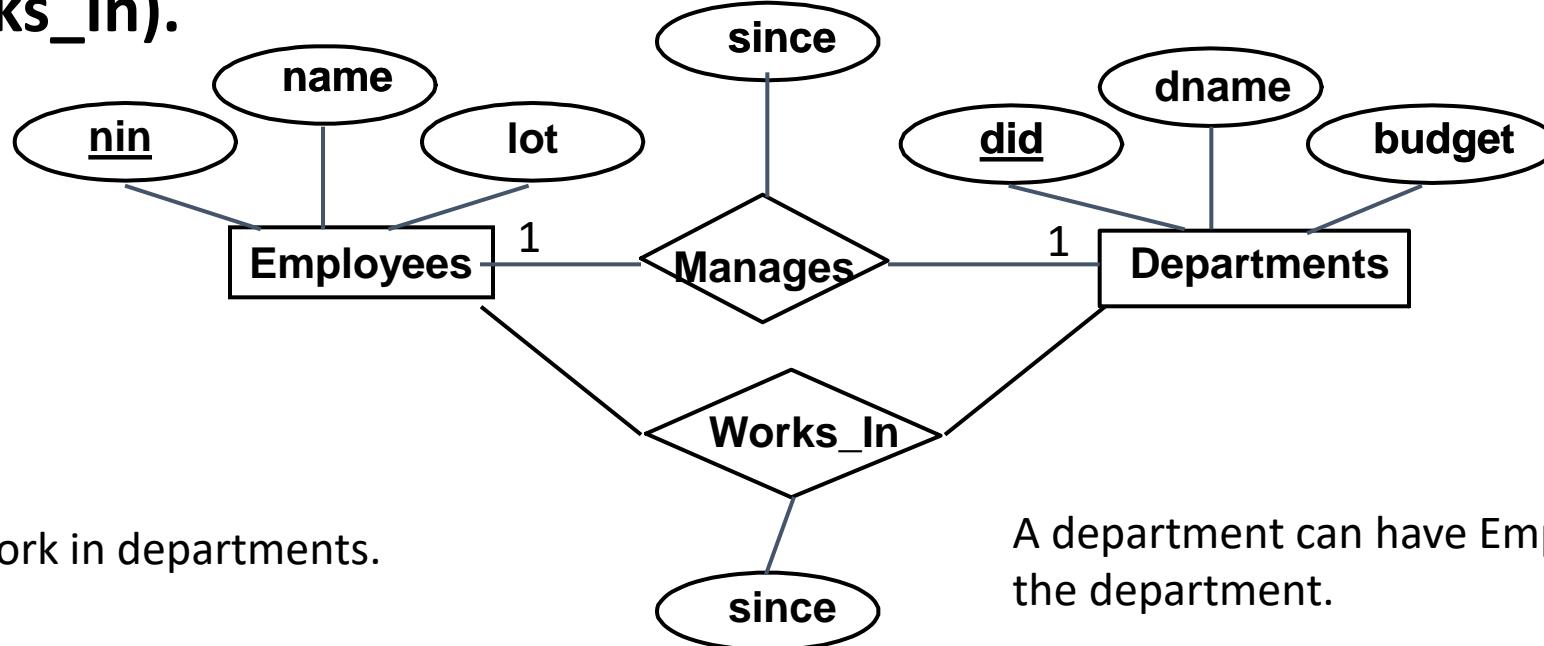
- Consider the **cardinality** between Employees and Departments (**Works_In**).



Let's fill this.



- Consider the **cardinality** between Employees and Departments (**Works_In**).



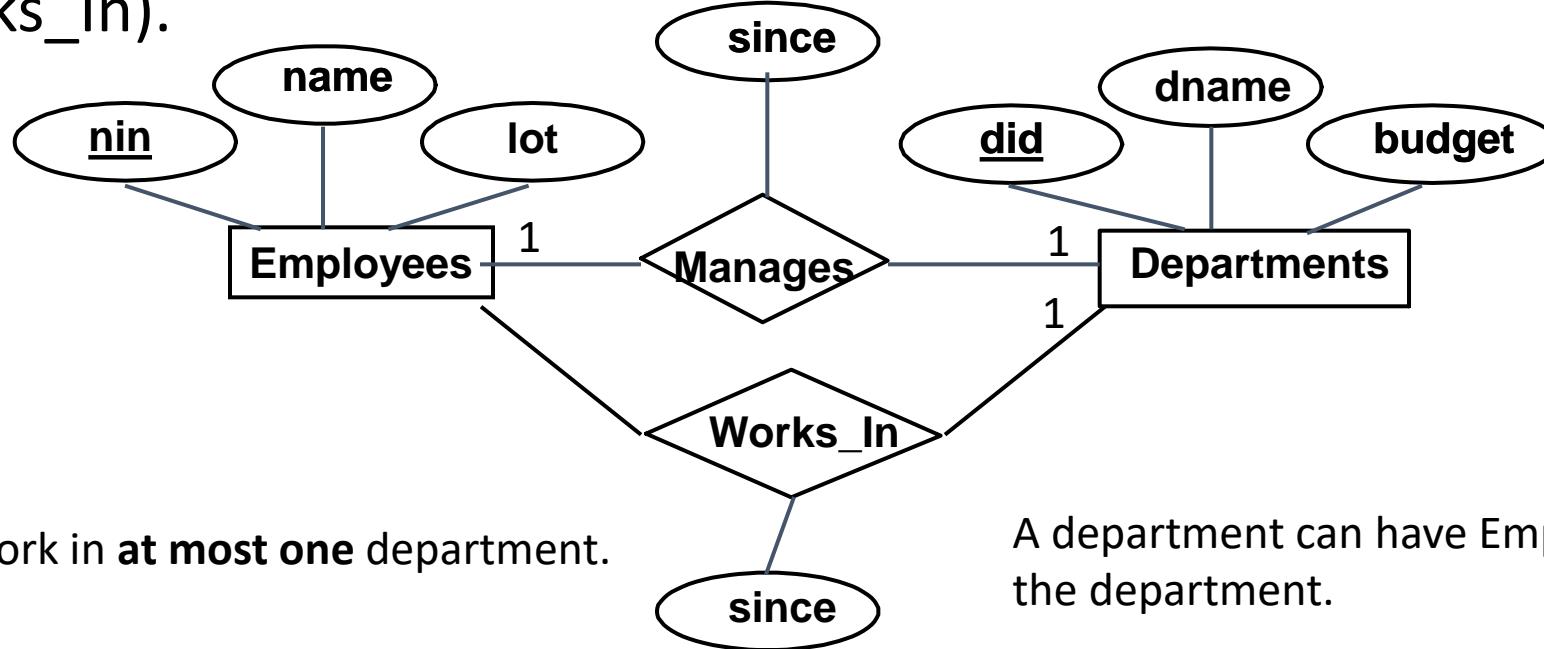
An employee can work in departments.

A department can have Emps. working in the department.

Let's fill this.



- Consider the **cardinality** between Employees and Departments (Works_In).



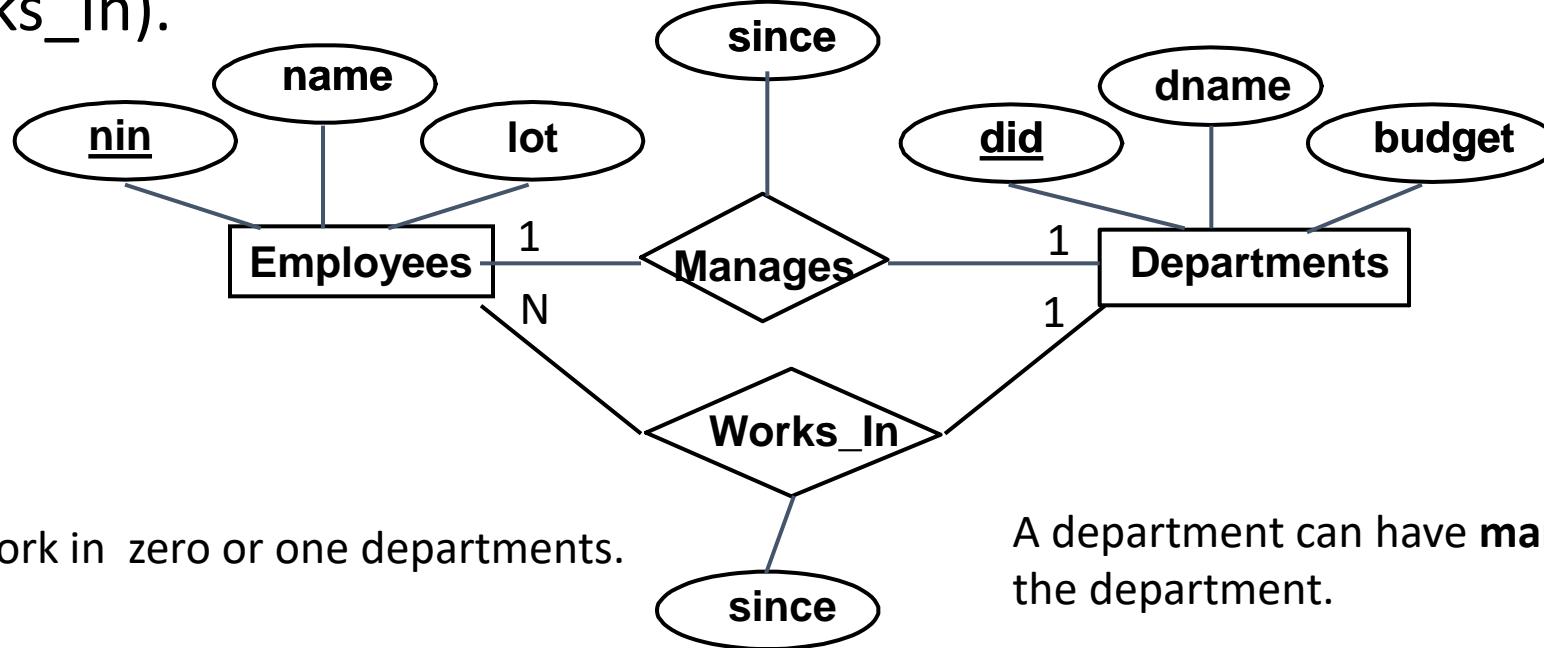
An employee can work in **at most one** department.

A department can have Emps. working in the department.

Let's fill this.



- Consider the **cardinality** between Employees and Departments (Works_In).



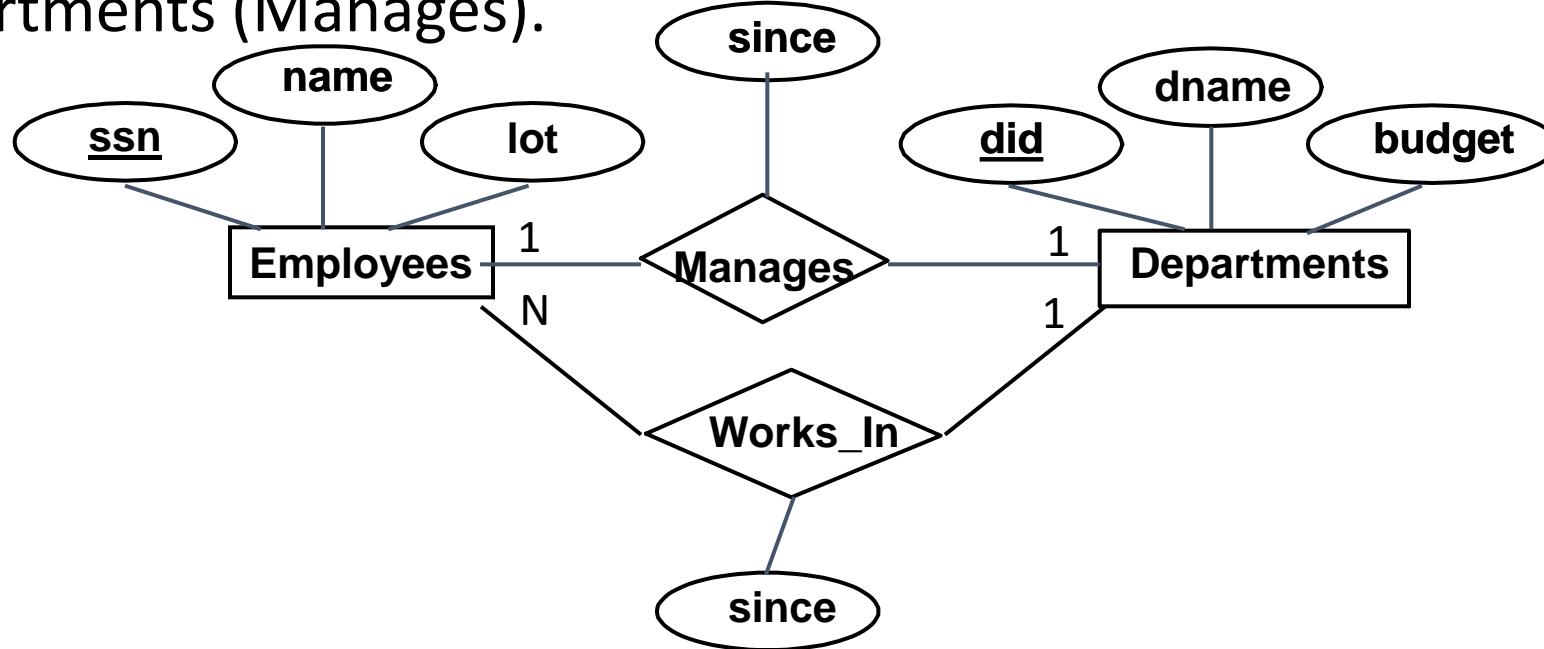
An employee can work in zero or one departments.

A department can have many Emps. working in the department.

Let's fill this.



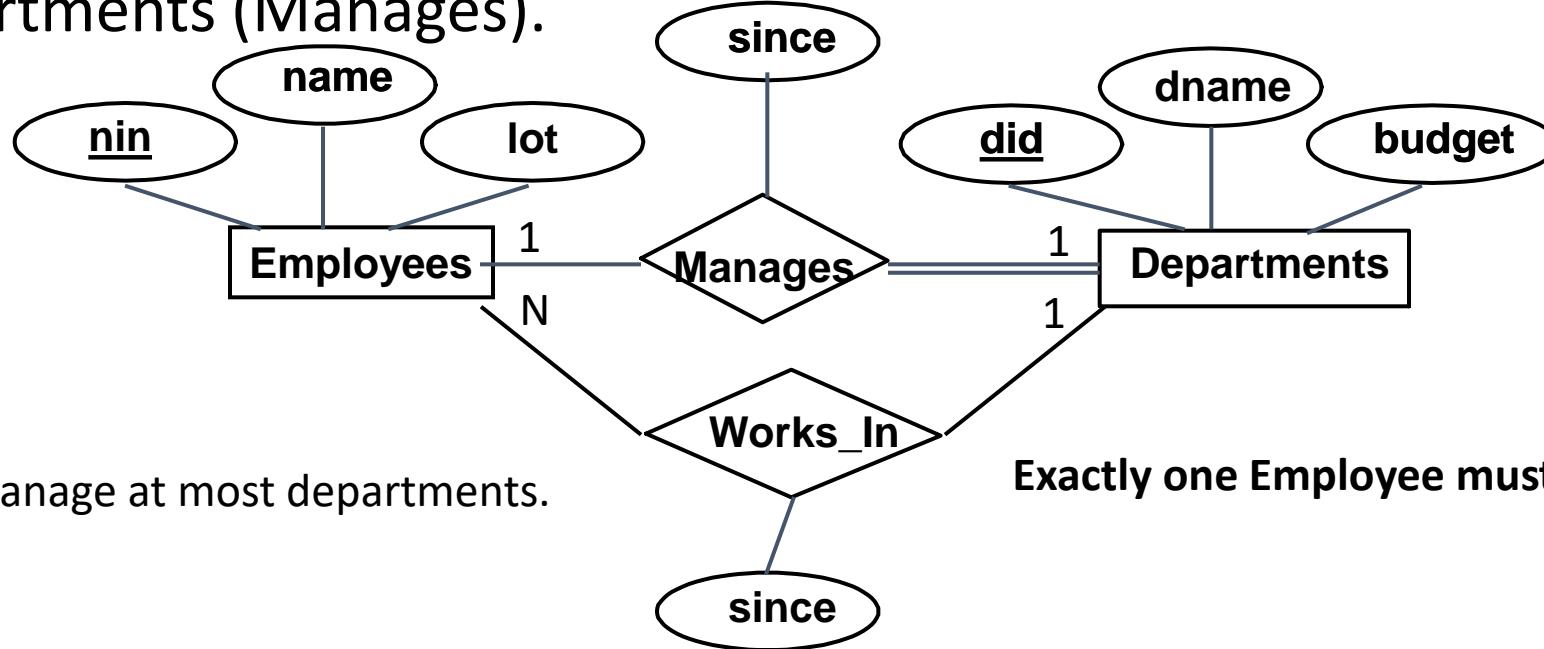
- Consider the **participation constraints** between Employees and Departments (Manages).



Let's fill this.



- Consider the **participation constraints** between Employees and Departments (Manages).



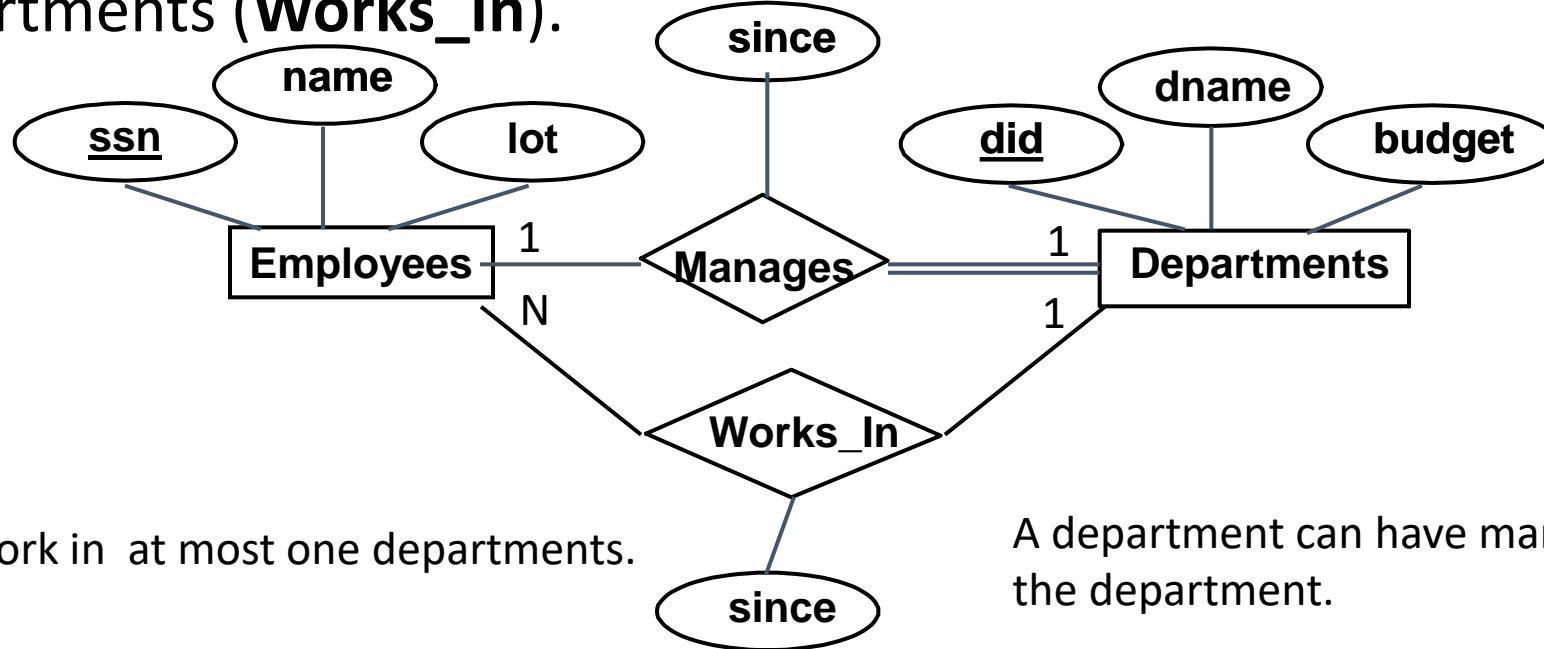
An employee can manage at most departments.

Exactly one Employee must manage a department.

Let's fill this.



- Consider the **participation constraints** between Employees and Departments (**Works_In**).



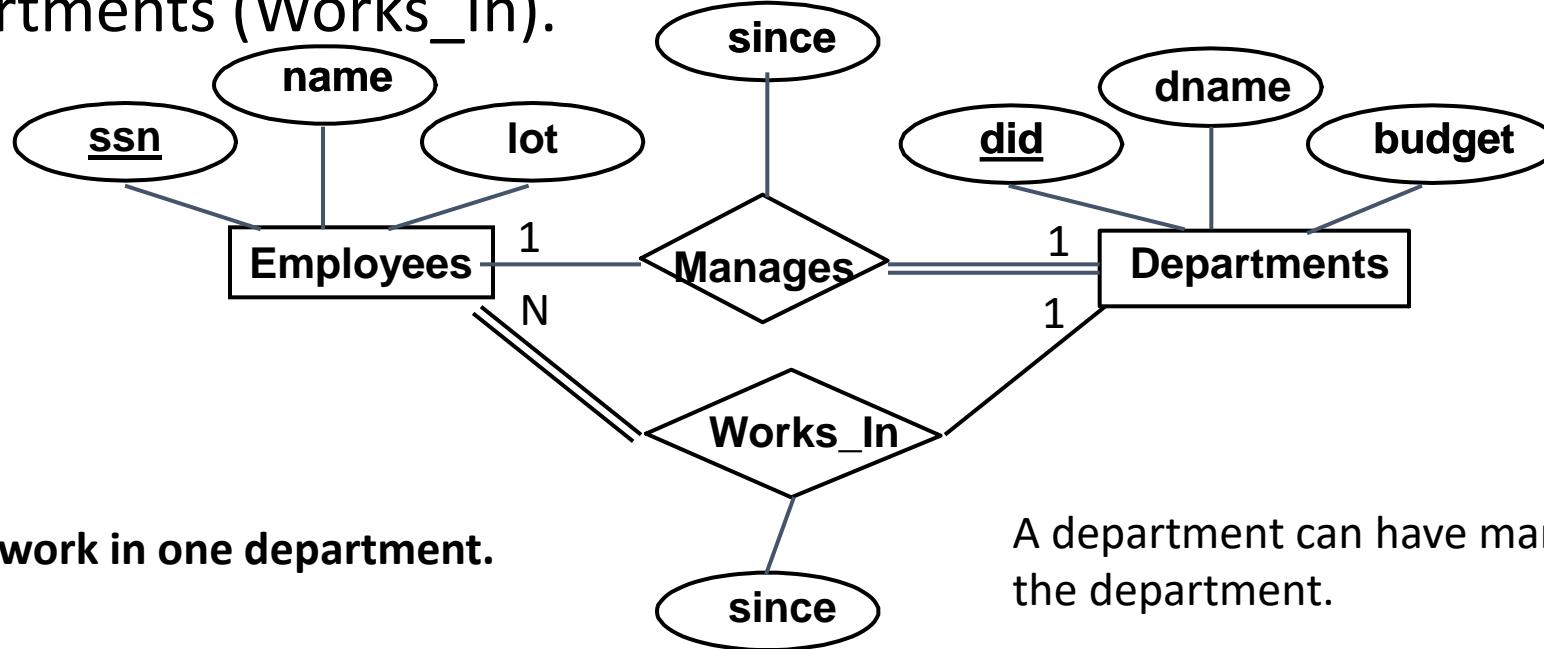
An employee can work in at most one departments.

A department can have many Emps. working in the department.

Let's fill this.



- Consider the **participation constraints** between Employees and Departments (`Works_In`).



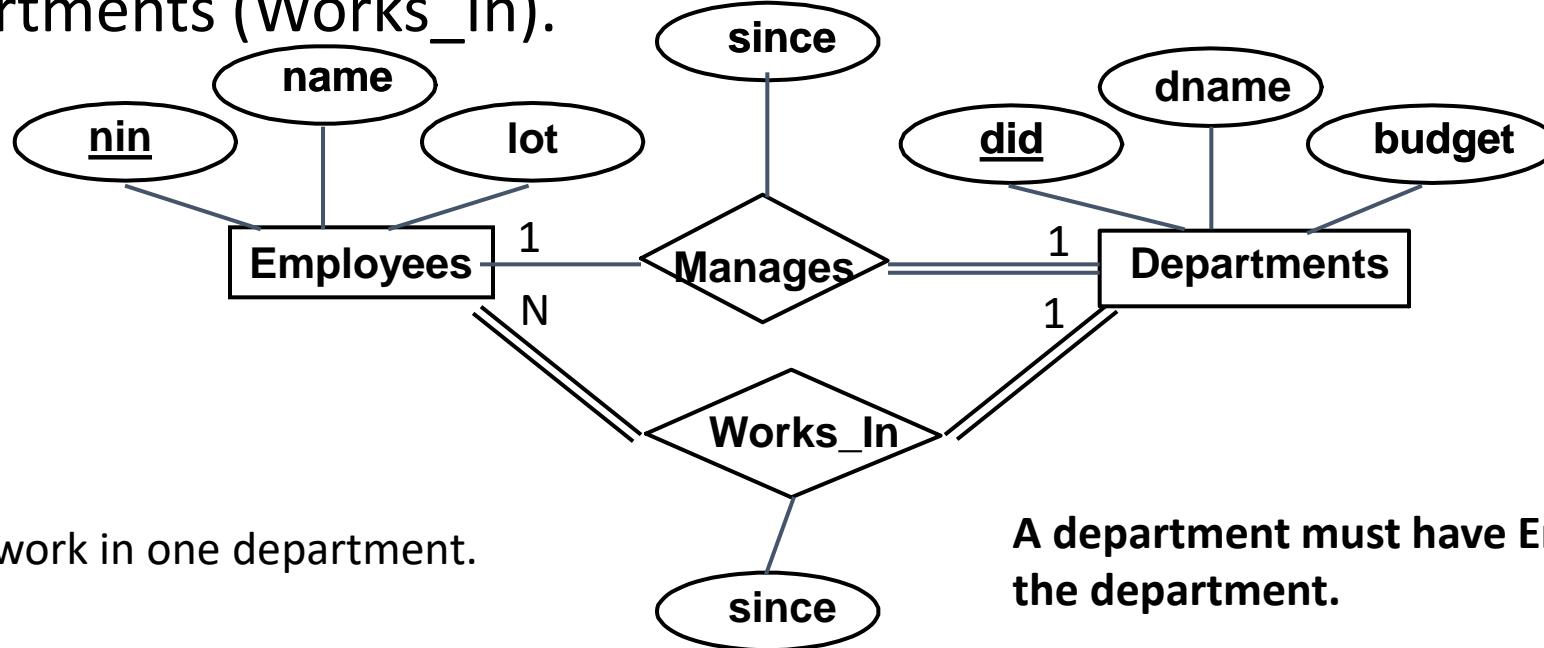
An employee must work in one department.

A department can have many Emps. working in the department.

Let's fill this.



- Consider the **participation constraints** between Employees and Departments (`Works_In`).



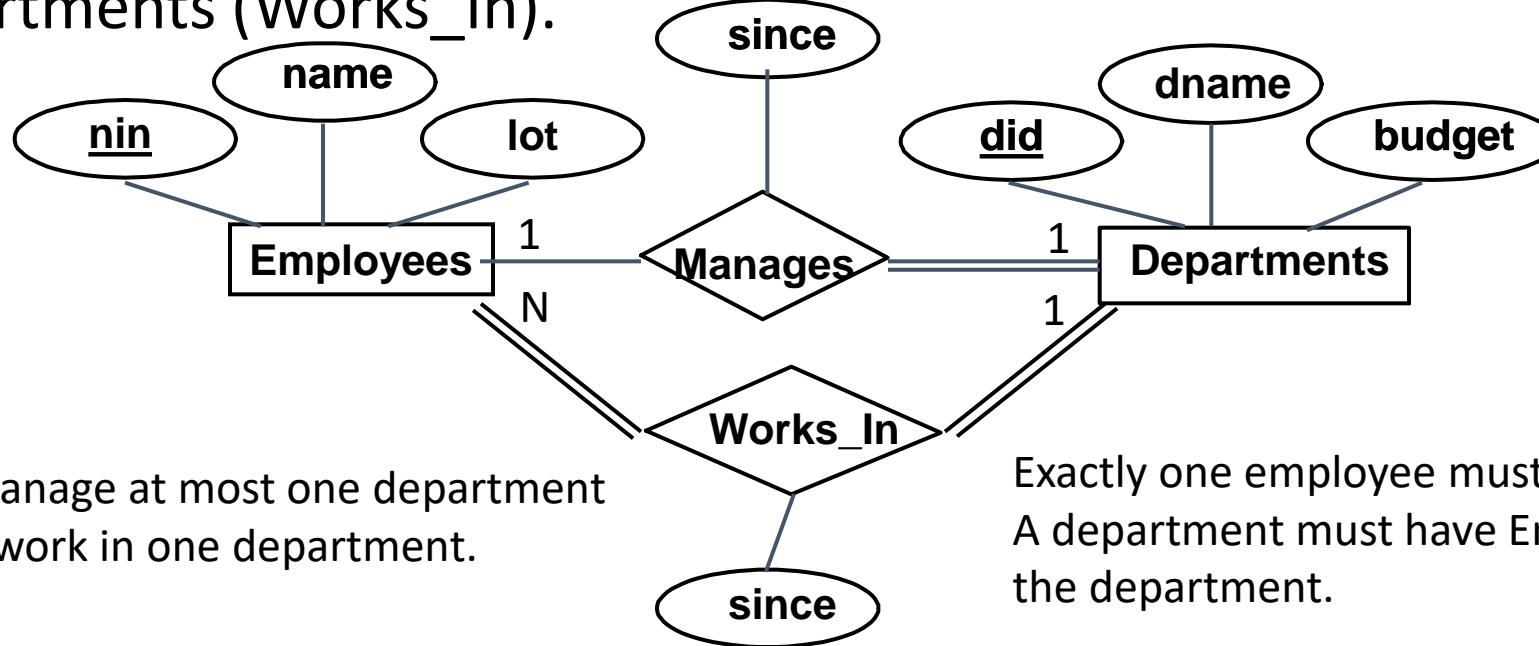
An employee must work in one department.

A department must have Emps. working in the department.

Let's fill this.



- Consider the **participation constraints** between Employees and Departments (`Works_In`).

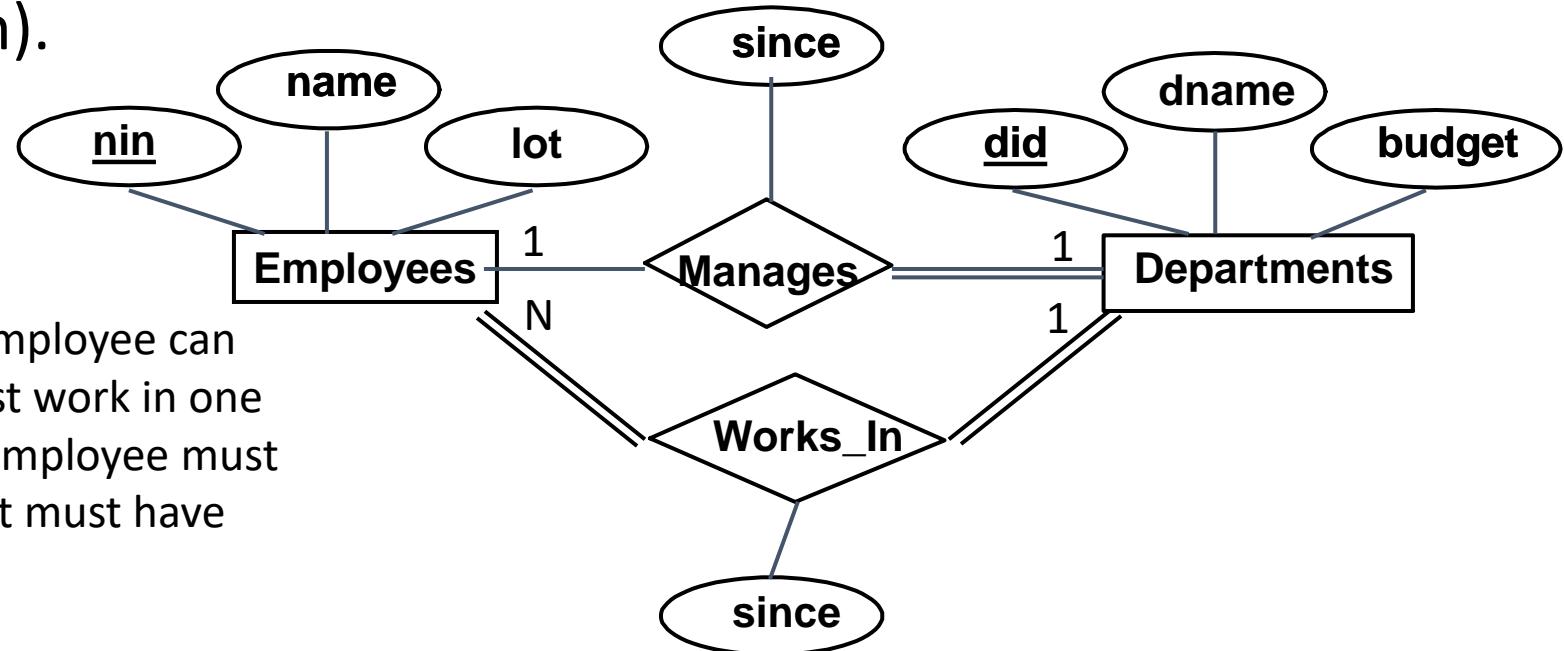


An employee can manage at most one department
An employee must work in one department.

Exactly one employee must manage a department.
A department must have Emps. working in the department.

Let's fill this.

- Consider the **participation constraints** between Employees and Departments (Works_In).



"Please draw an ER diagram where an employee can manage at most one department or must work in one department providing that exactly one employee must manage a department and a department must have Emps. working in the department."

Weak entity and weak relation sets.



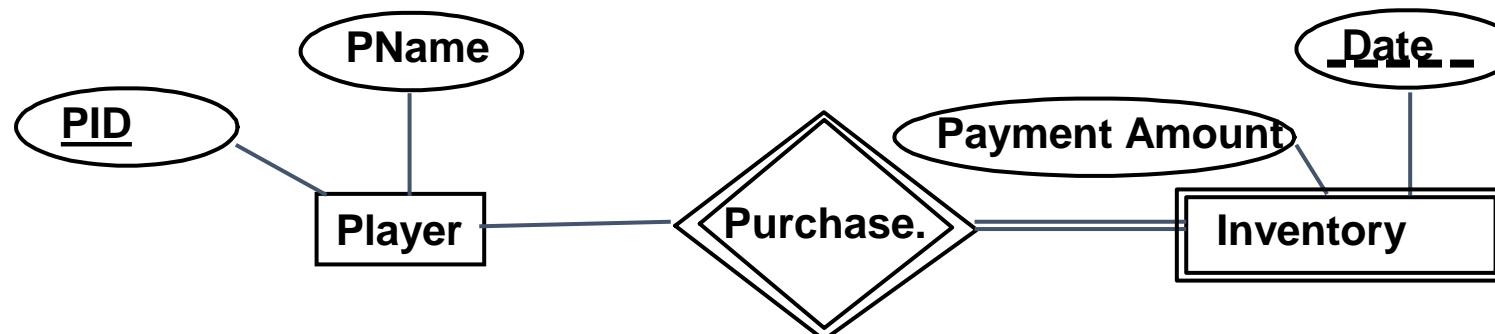
Consider the following case.

A player makes in-game equipment (armour, ammunition, etc) purchases using imaginary money (coins). The game server has to keep the information as long as the player plays the game.

Assume we do not want to keep the purchase information when the player deletes their account, and we want to automate this deletion operation.

The DBMS automatically remove all the redundant purchase data from the tables.

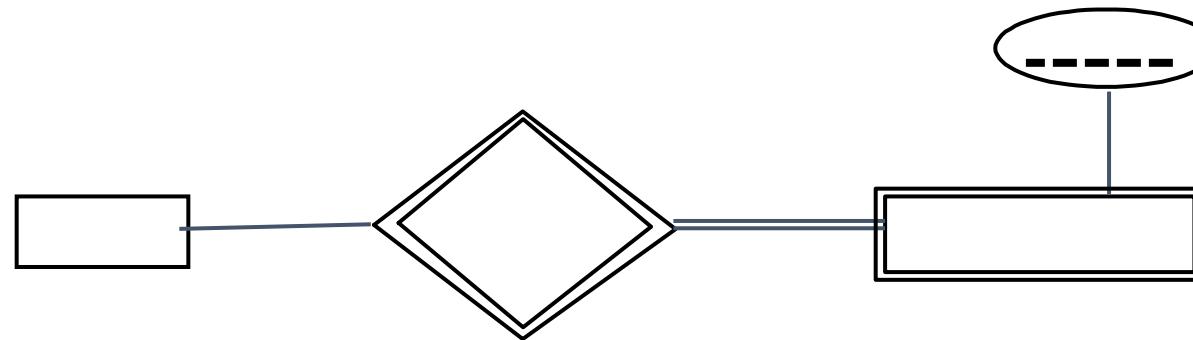
We use Weak-Entity and Relation sets to represent this.



Weak Entity-Relation Sets



Represented by a **double-lined diamond**, a **double-lined rectangle** connected by double lines. The weak key attribute is represented by a dashed underline.

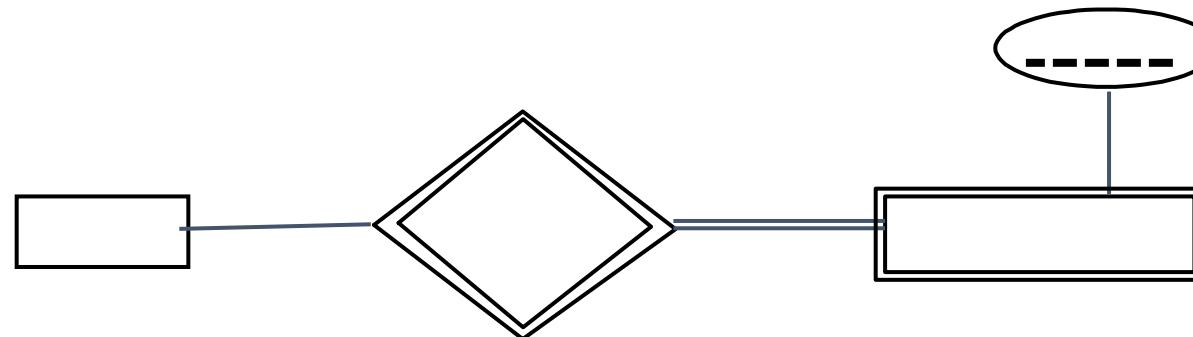


Weak Entity-Relation Sets



Represented by a **double-lined diamond**, a **double-lined rectangle** connected by double lines. The weak key attribute is represented by a dashed underline.

The double-lined rectangle is the **subject**, the single-lined rectangle is the **owner**.



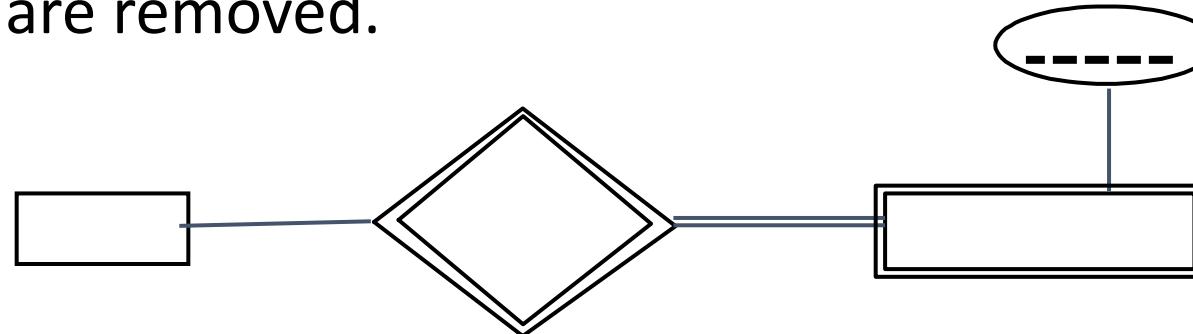
Weak Entity-Relation Sets



Represented by a **double-lined diamond**, a **double-lined rectangle** connected by double lines. The weak key attribute is represented by a dashed underline.

The double-lined rectangle is the **subject**, the single-lined rectangle is the **owner**.

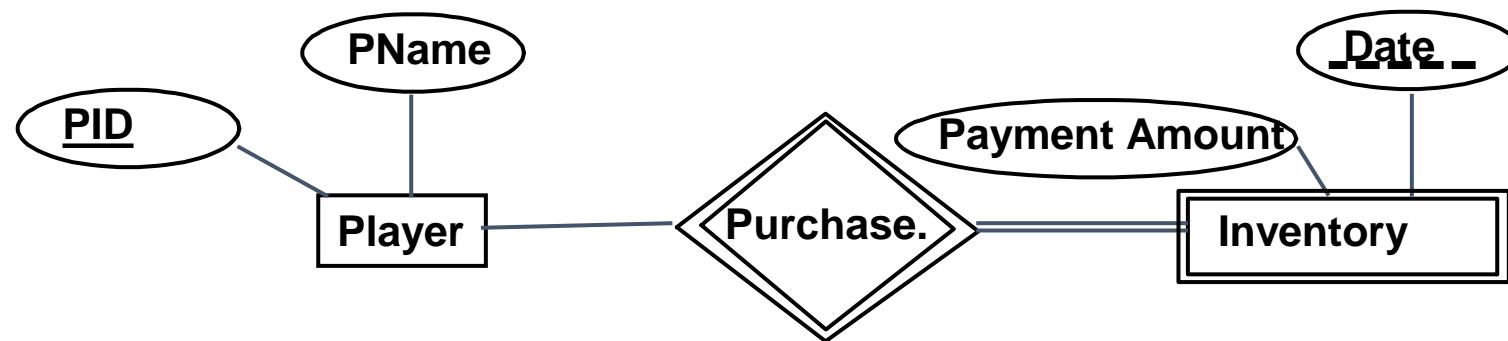
When an entity in the owner table is removed, all the related entries in the subject table are removed.



Weak entity set and weak relation sets.



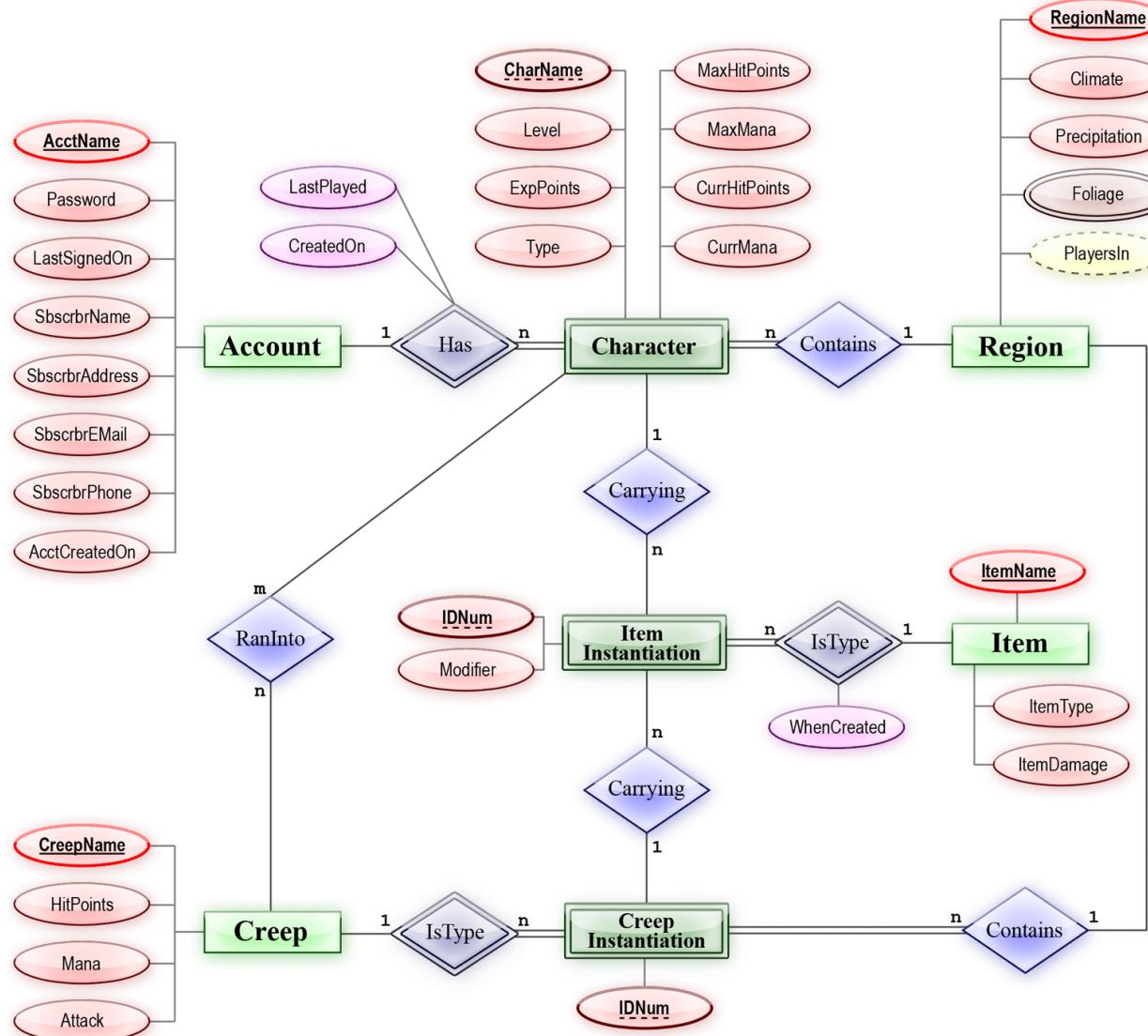
- Weak Entity sets do not possess a Primary Key; they **possess a Weak Key**.
- The primary key of the Owner table and the Weak Key of the Subject table constitutes a key for the Subject Table. (PID, Data is a key for Inventory)



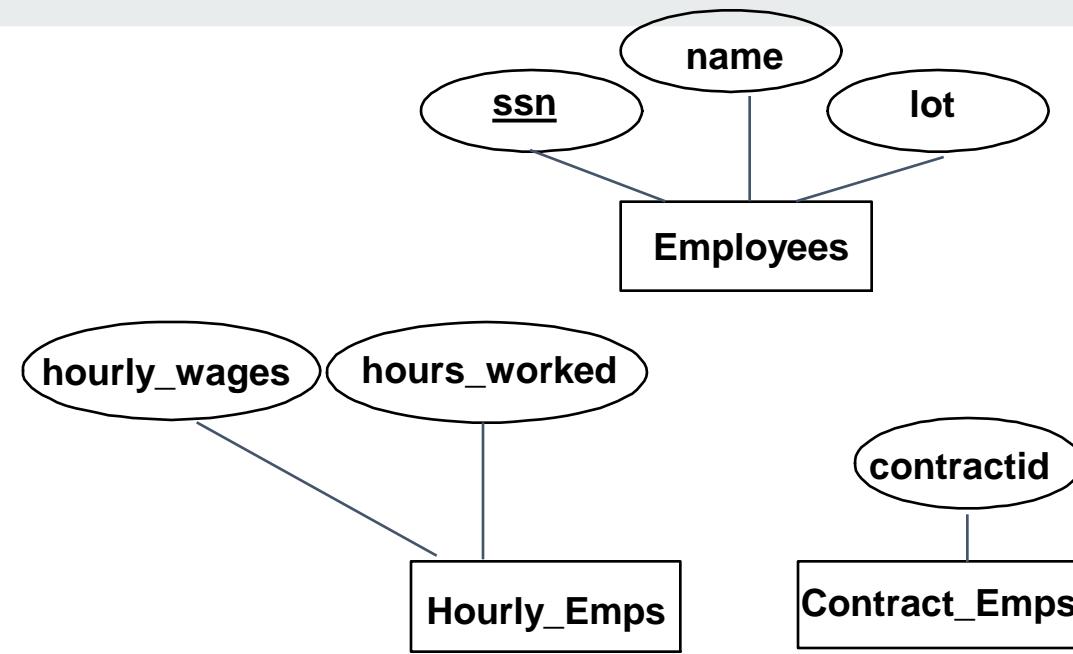
PName	<u>PID</u>
Alex	1
Mark	2
Claudia	3
Summer	4

Payment Amount	Date
1121	01/04/1982
12312	01/04/1982
1121	01/04/1958
1121	10/02/1994

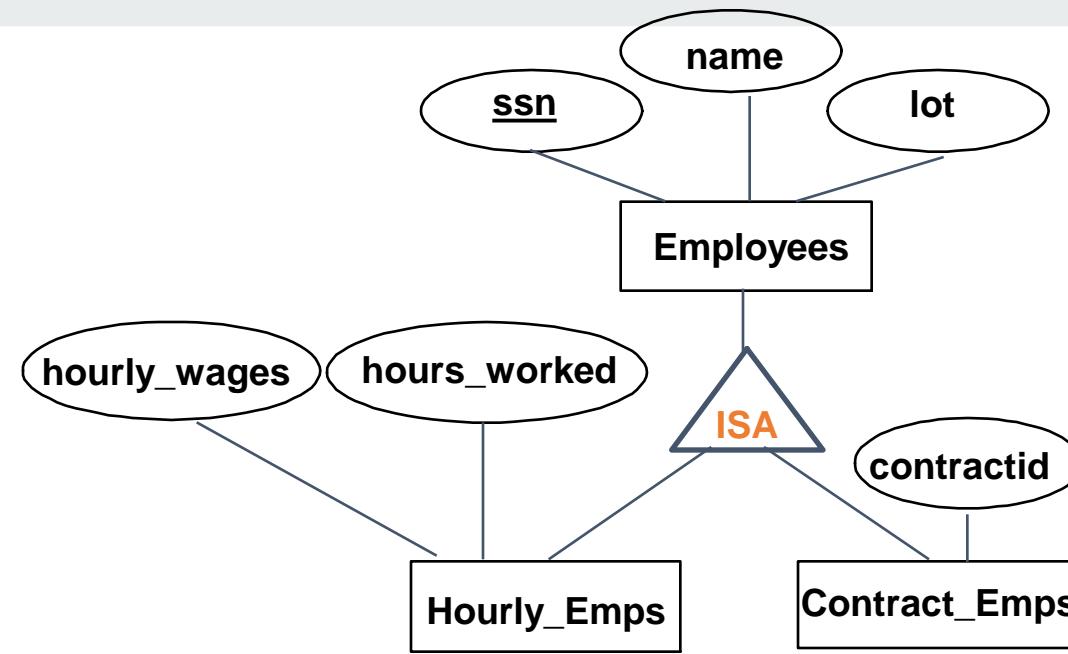
Massively multiplayer online role-playing game



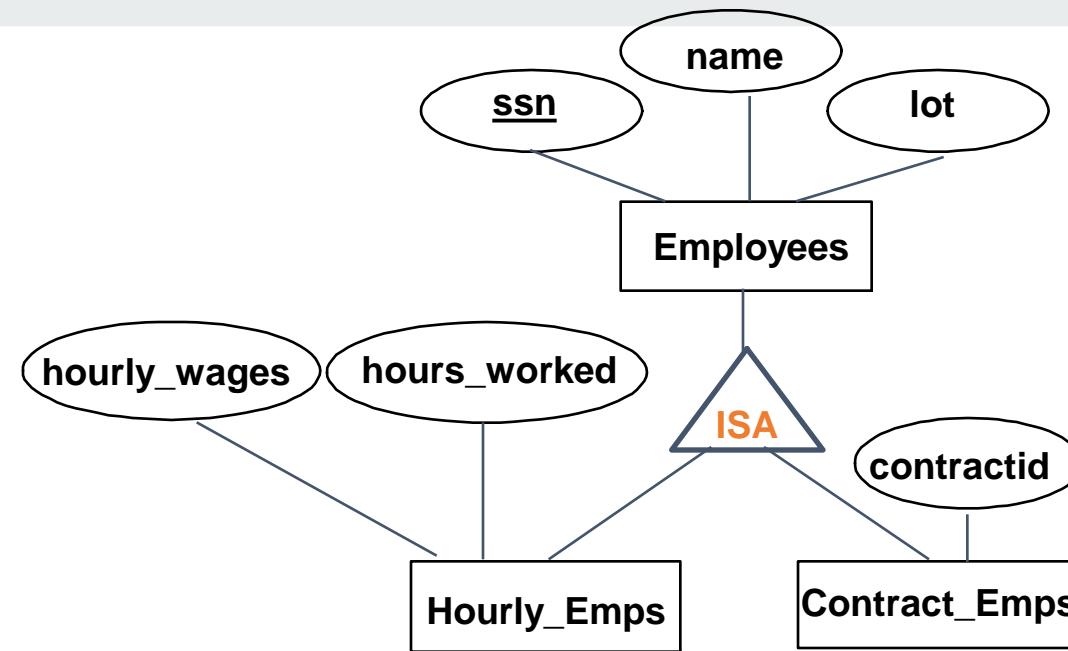
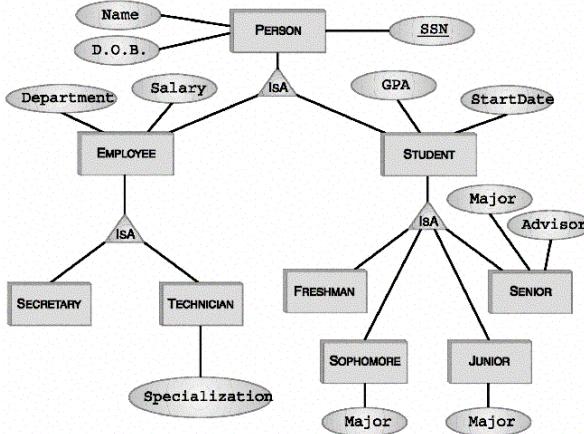
Extended ER ISA ('is a') Hierarchies



Extended ER ISA ('is a') Hierarchies



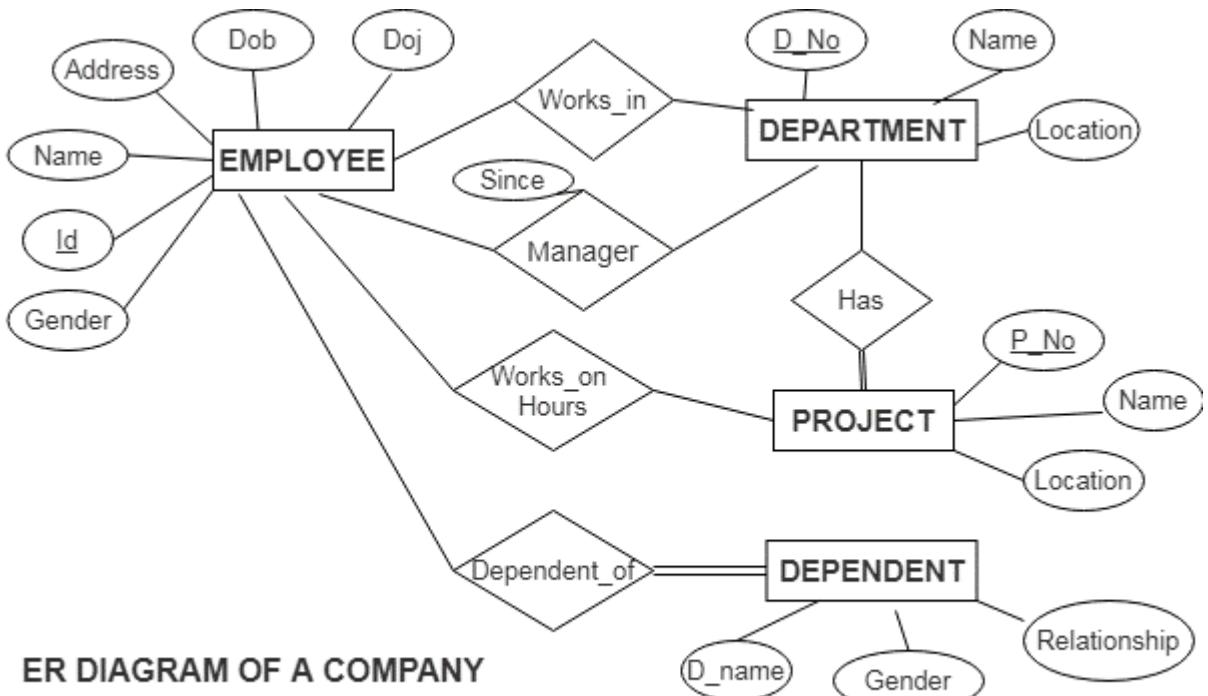
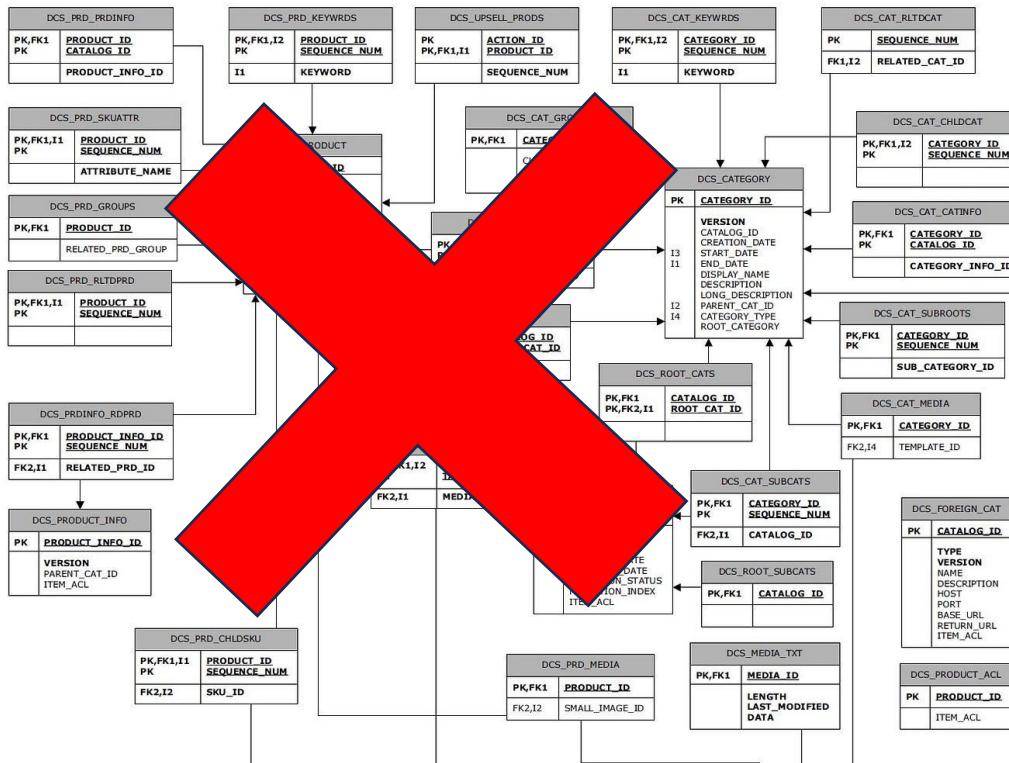
Extended ER ISA ('is a') Hierarchies



- *Overlap constraints:* Can Uraz be an Hourly Employee as well as a Contract Employee?
- *Covering constraints:* Does every Employee also have to be an Hourly Employee or a Contract Employee?
- Reasons for using ISA:
 - To add descriptive attributes specific to a subclass.
 - To identify entities that participate in a relationship.

Recall: please use the symbolism we use in the lectures.

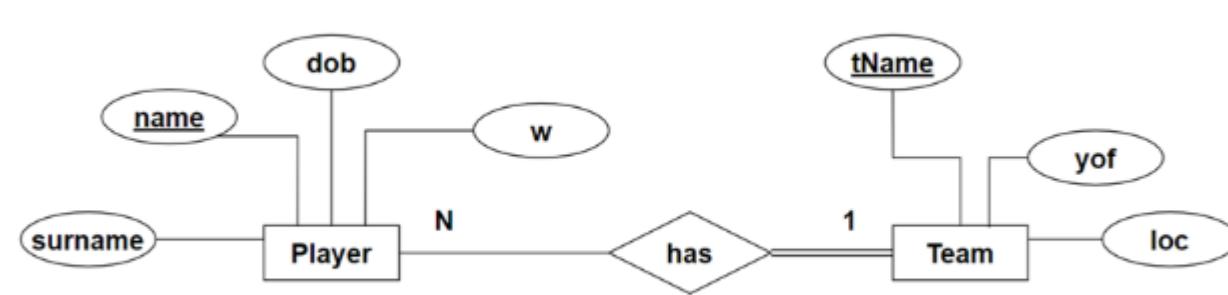
ATG Commerce Product Catalog Tables



Previous year Exam Question



Please examine the Erd given below:



Describe in words what the Erd shown represents.

A team must have at least one player.

A player may belong to at most one team.



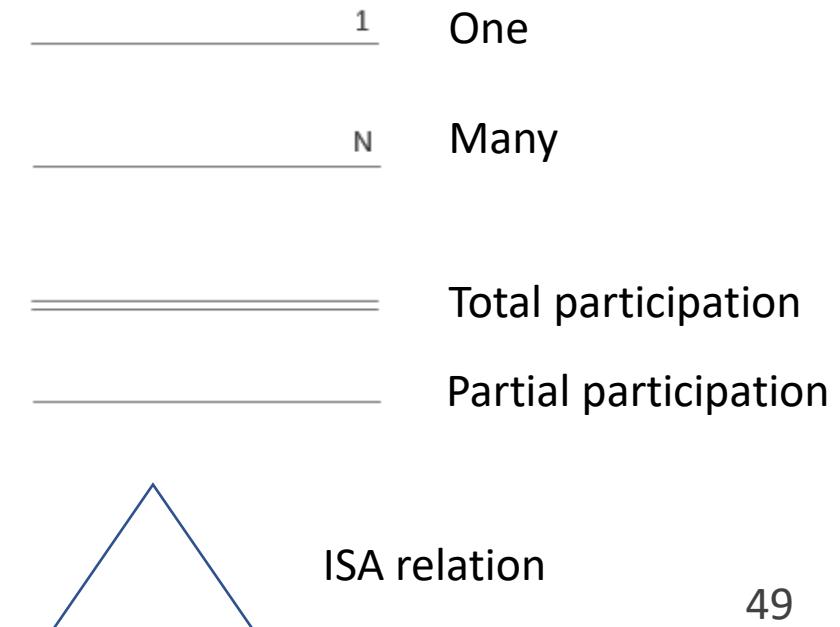
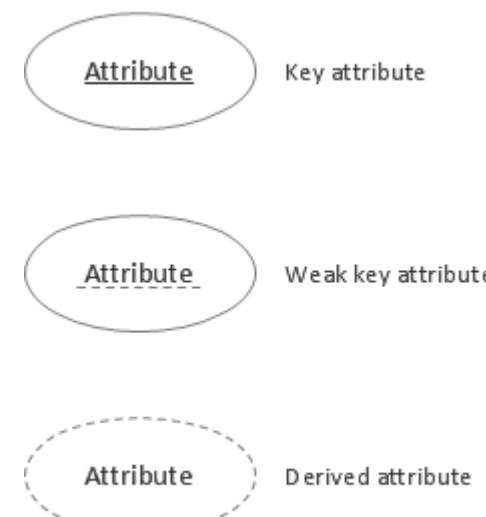
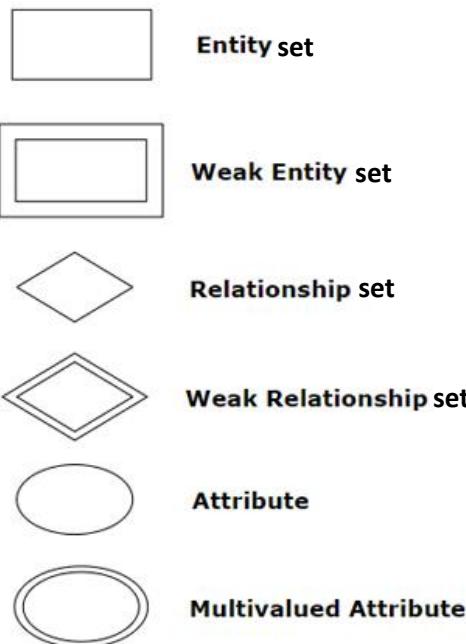
SCC.221 Data Engineering

2024 - Week 2 – Relational Database.
Uraz C Turker

What have we learnt so far?



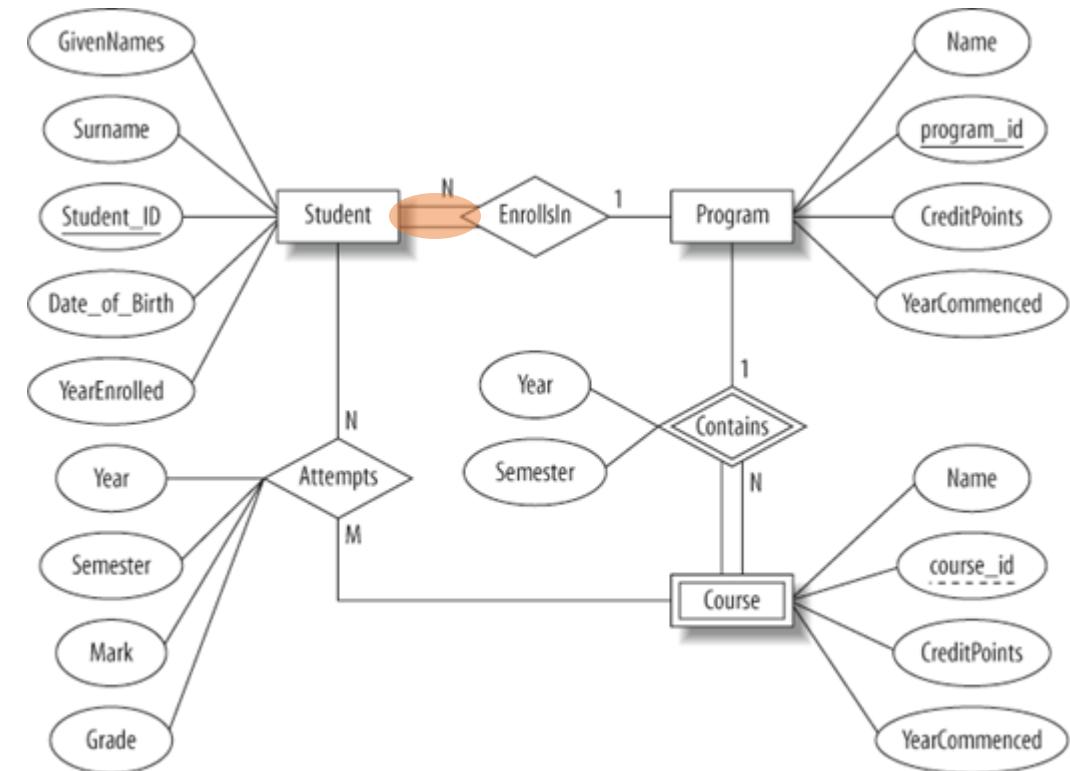
- We learn key ER concepts which allows us to design relational databases.



What have we learnt so far?



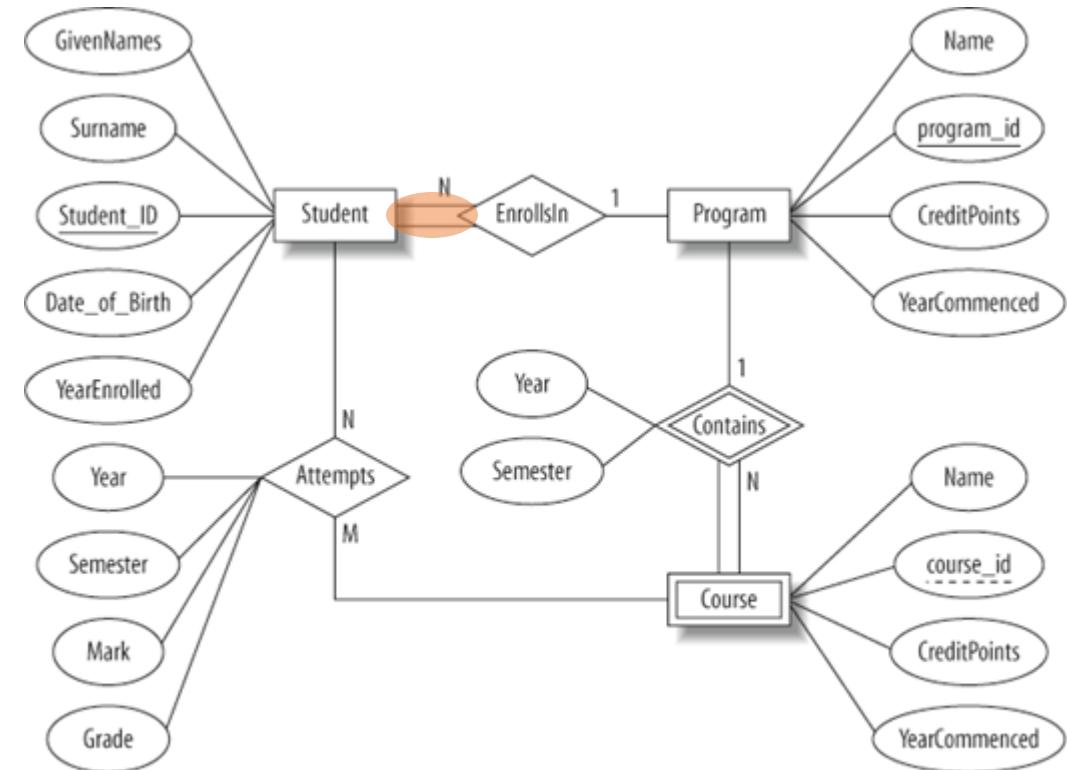
- A student enrol in program.



What have we learnt so far?



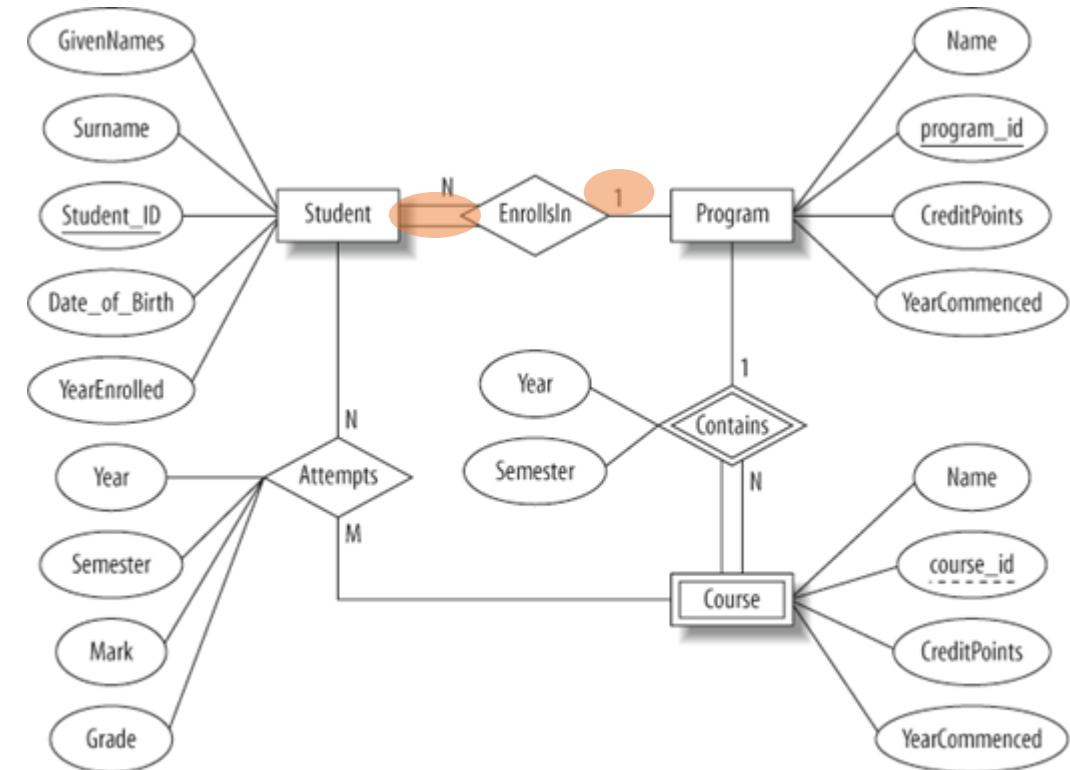
- A student **must** enrol in program.



What have we learnt so far?



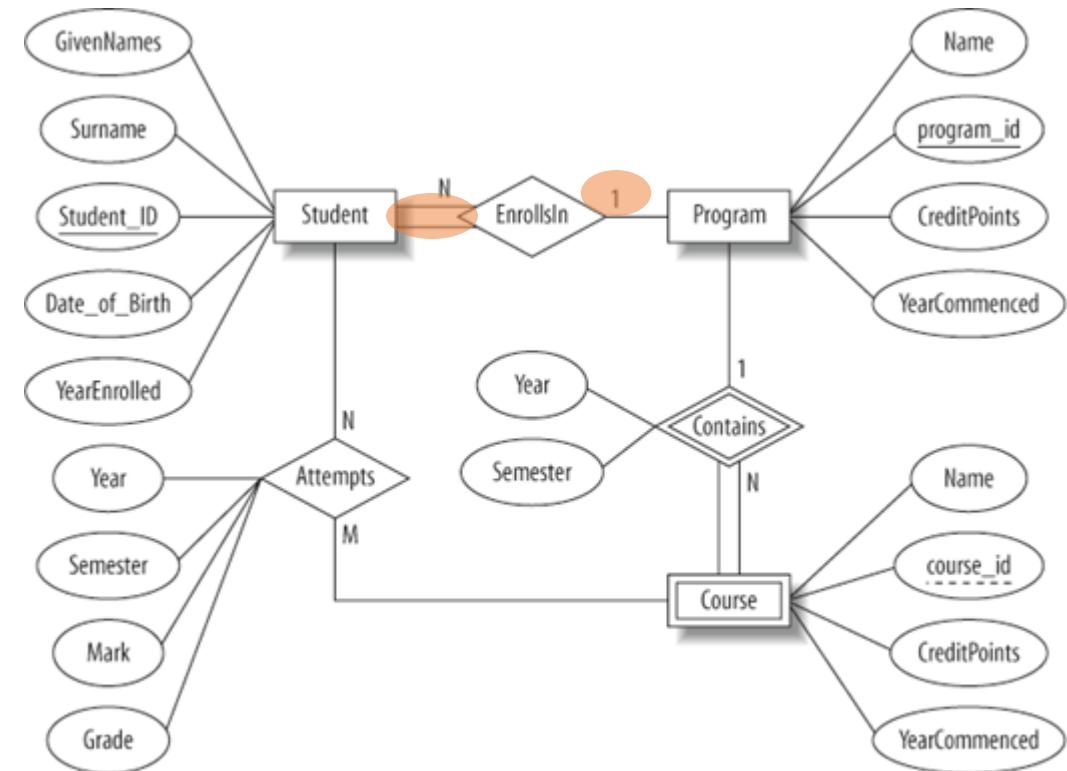
- A student **must** enrol in program.



What have we learnt so far?

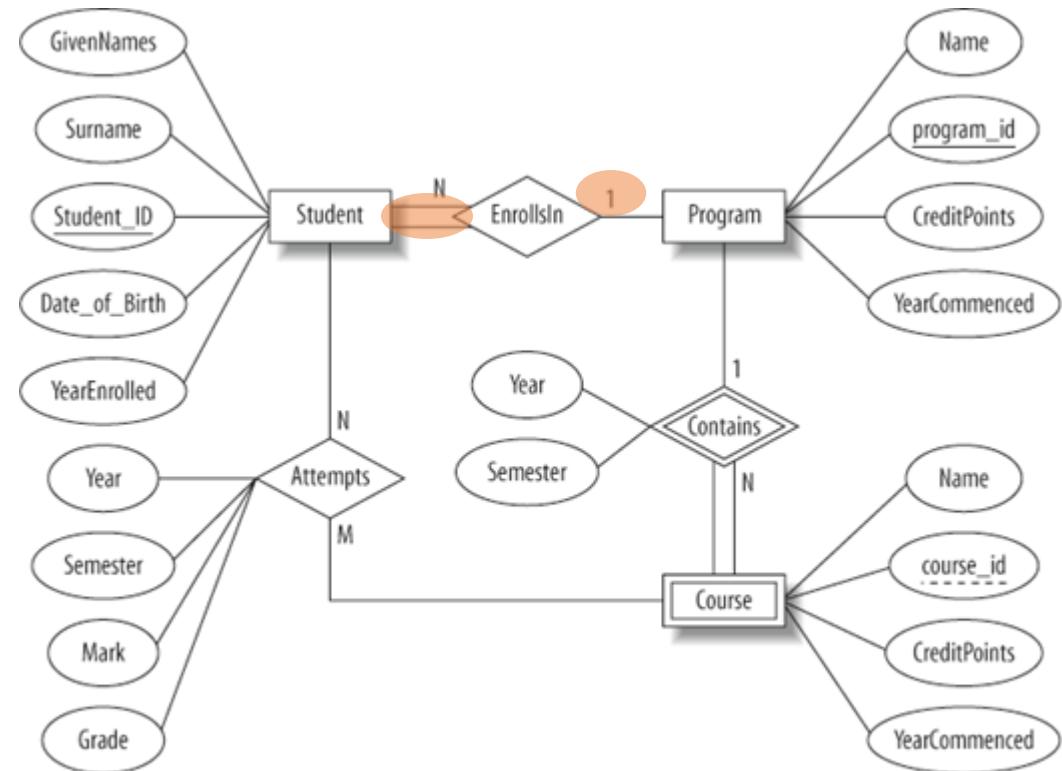


- A student **must** enrol in **one** program.



What have we learnt so far?

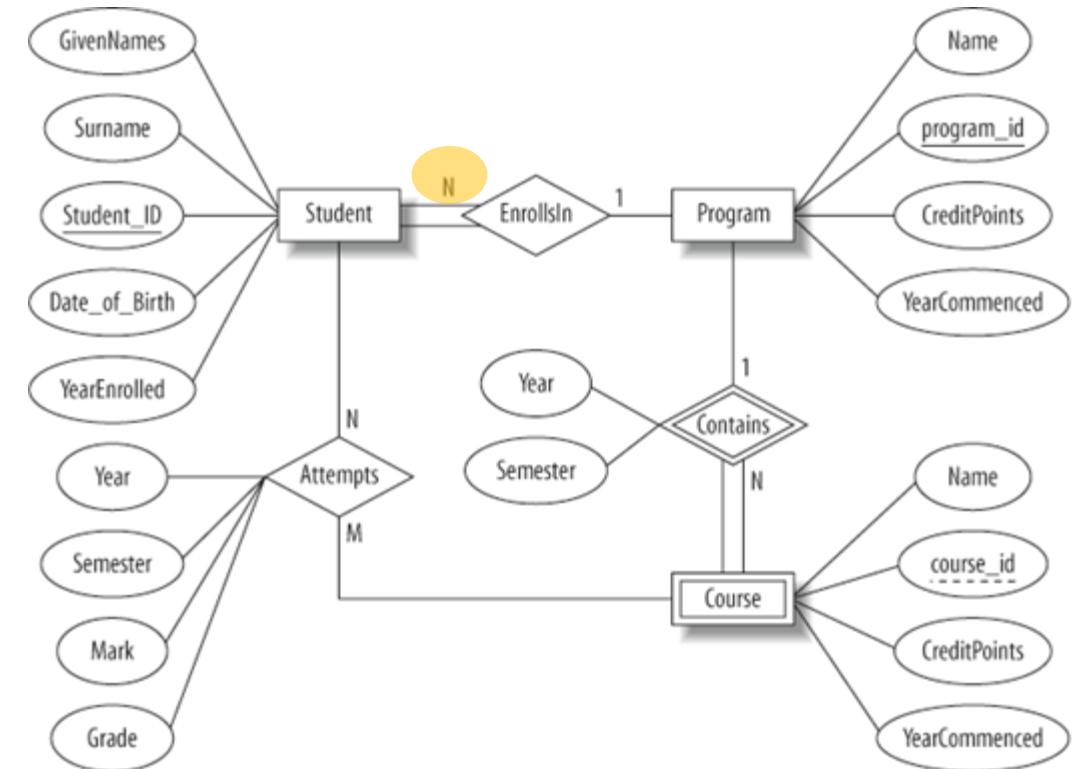
- A student **must** enrol in **one** program.
 - It has GivenNames, Surname, StudentID, Date_of_Birth, YearEnrolled attributes, where StudentID is a primary key.



What have we learnt so far?



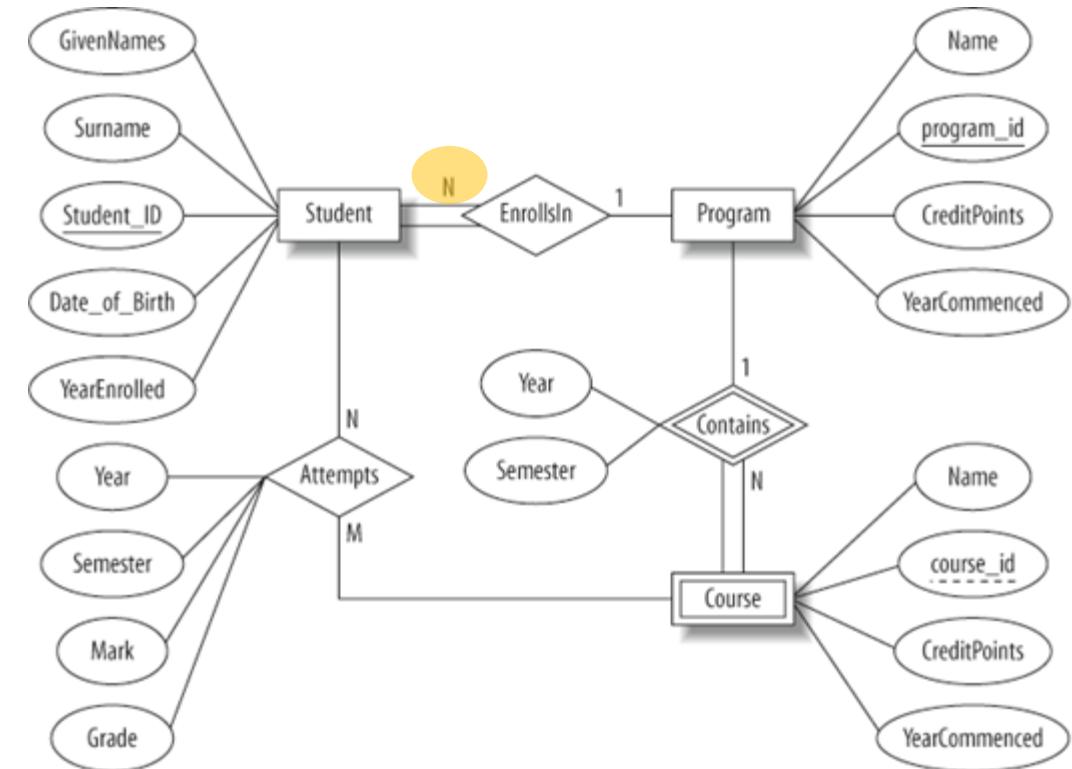
- Program have students.



What have we learnt so far?



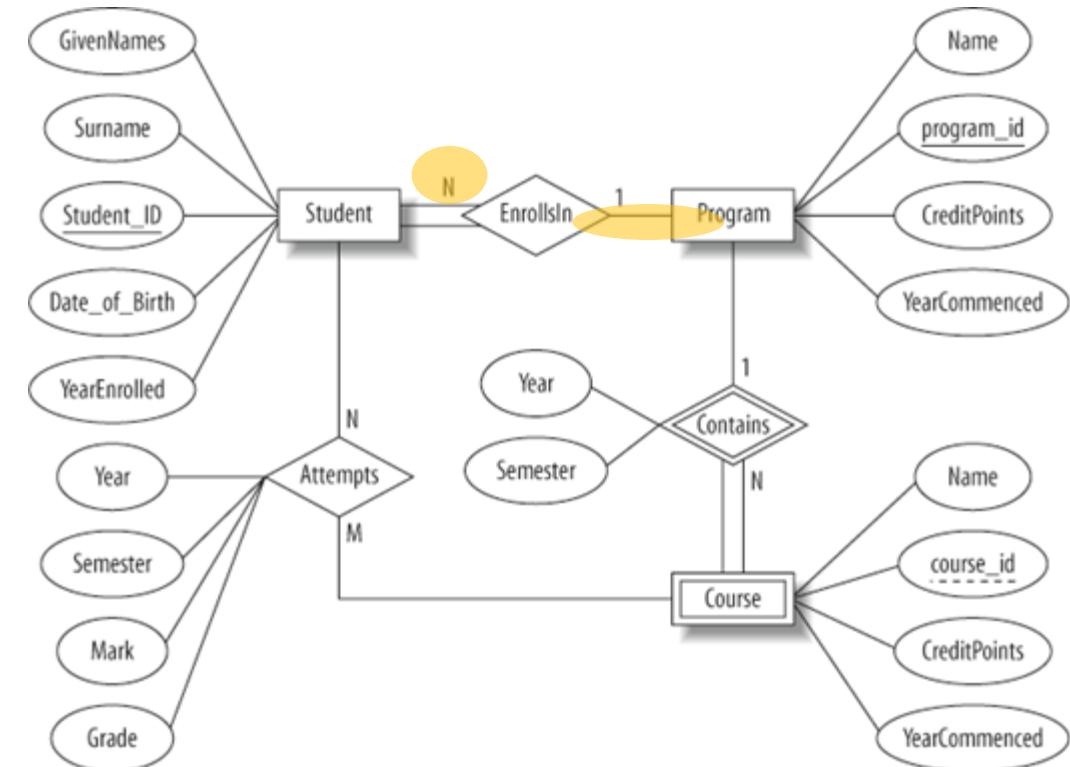
- Program have **many** students.



What have we learnt so far?



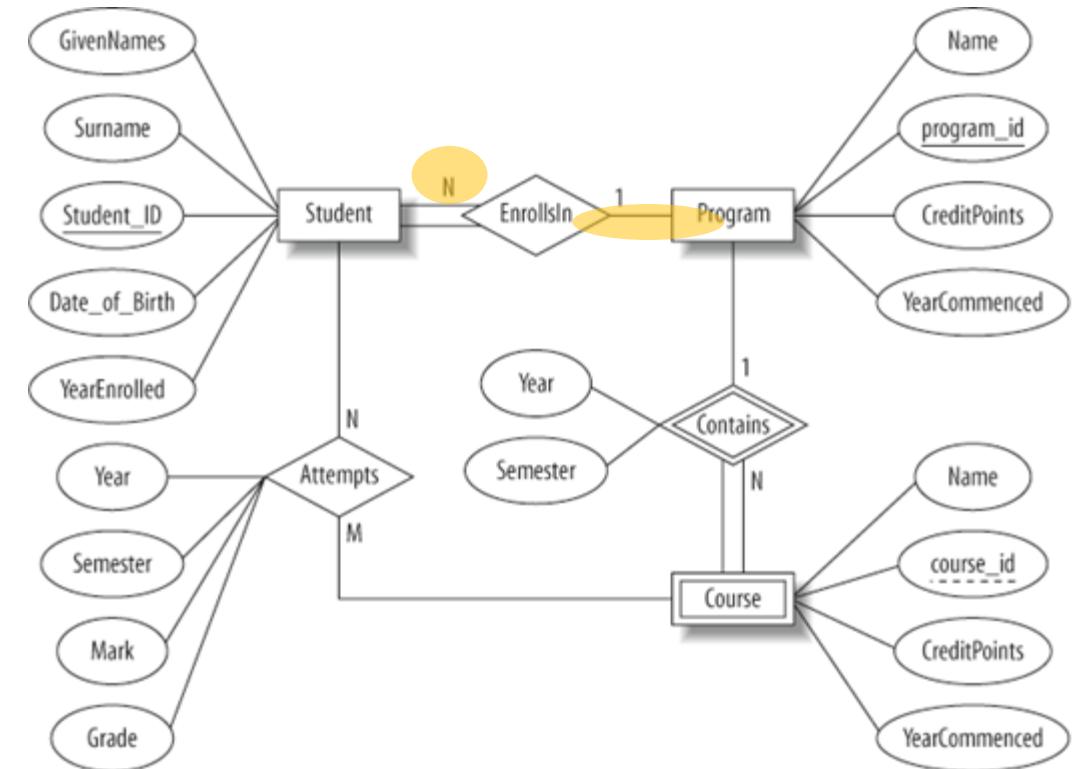
- Program have **many** students.



What have we learnt so far?



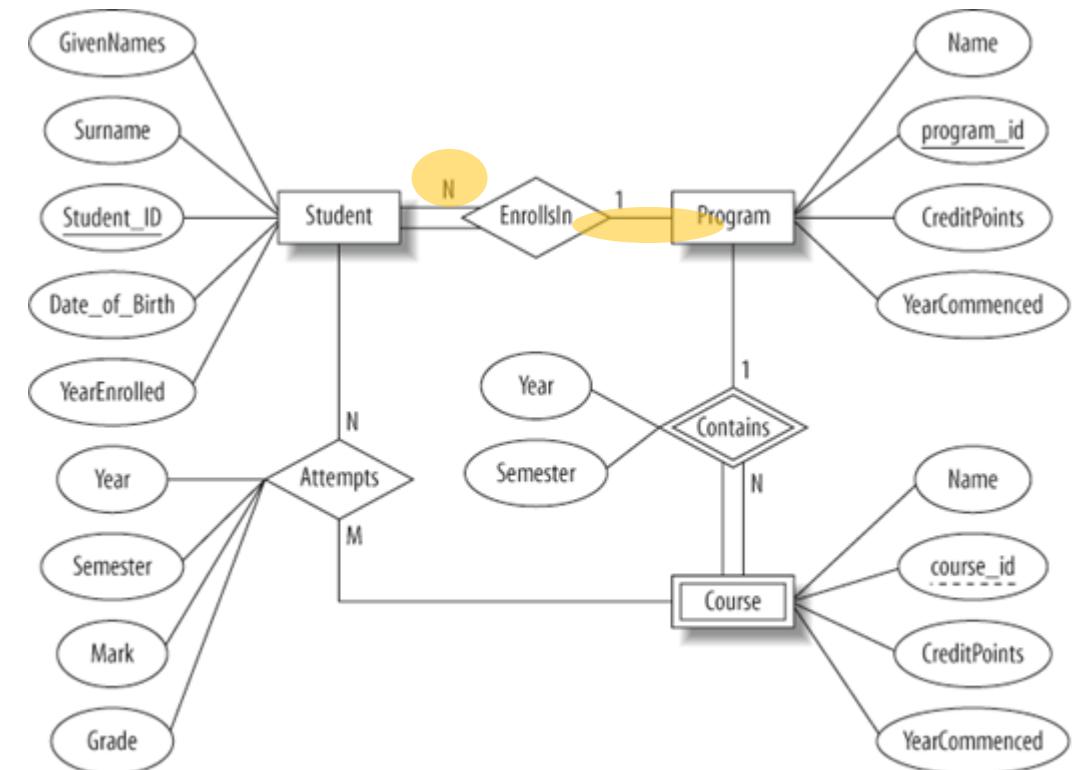
- Program **may** have **many** students.



What have we learnt so far?



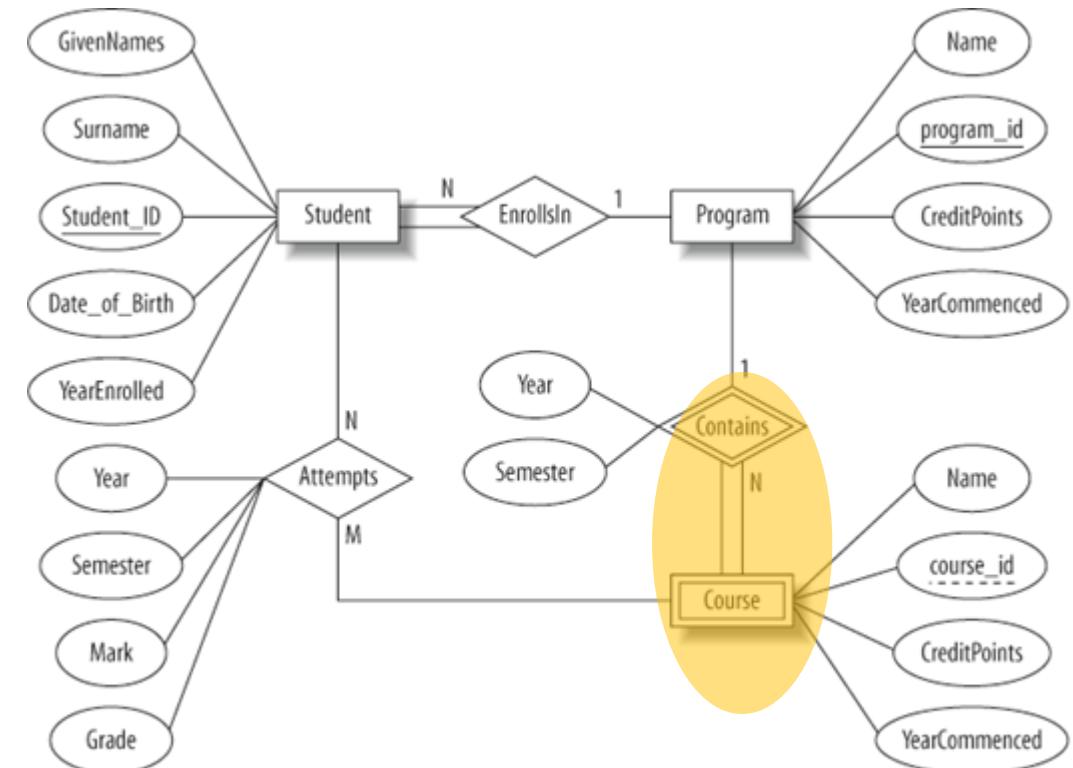
- Program **may** have **many** students.
 - It has name, program_id, CreditPoints, YearCommenced where program_id is a primary key.



What have we learnt so far?

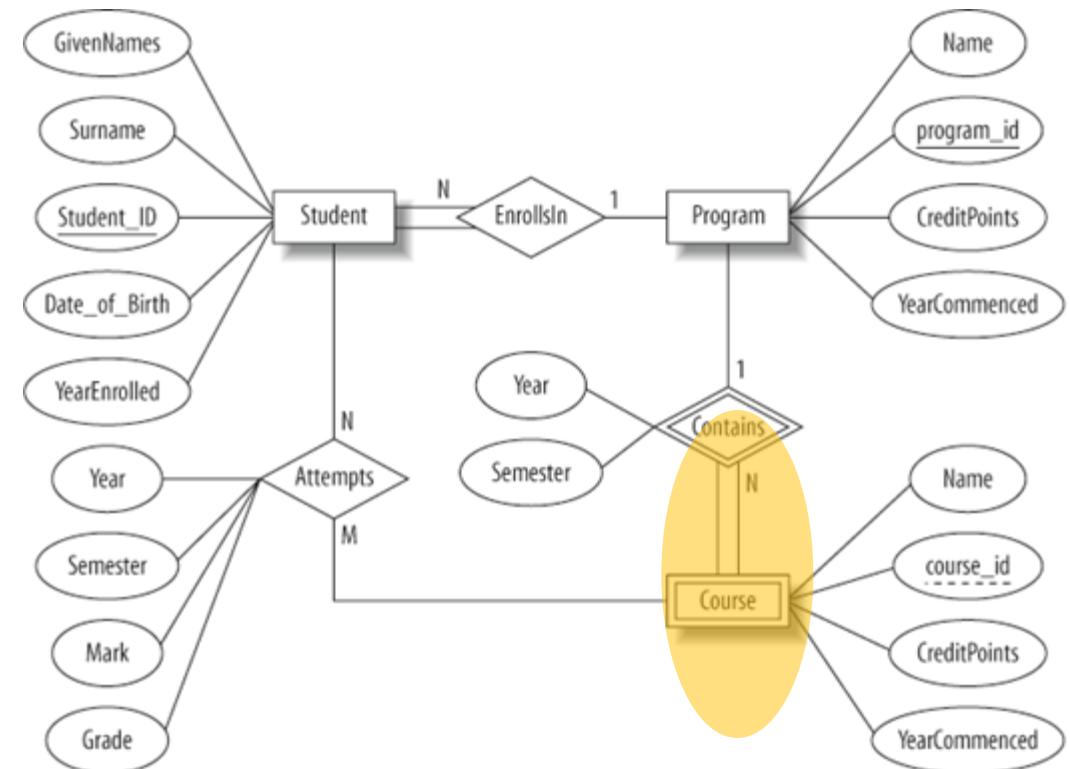


- When a program is deleted, the DBMS automatically deletes all related courses. (weak entity)



What have we learnt so far?

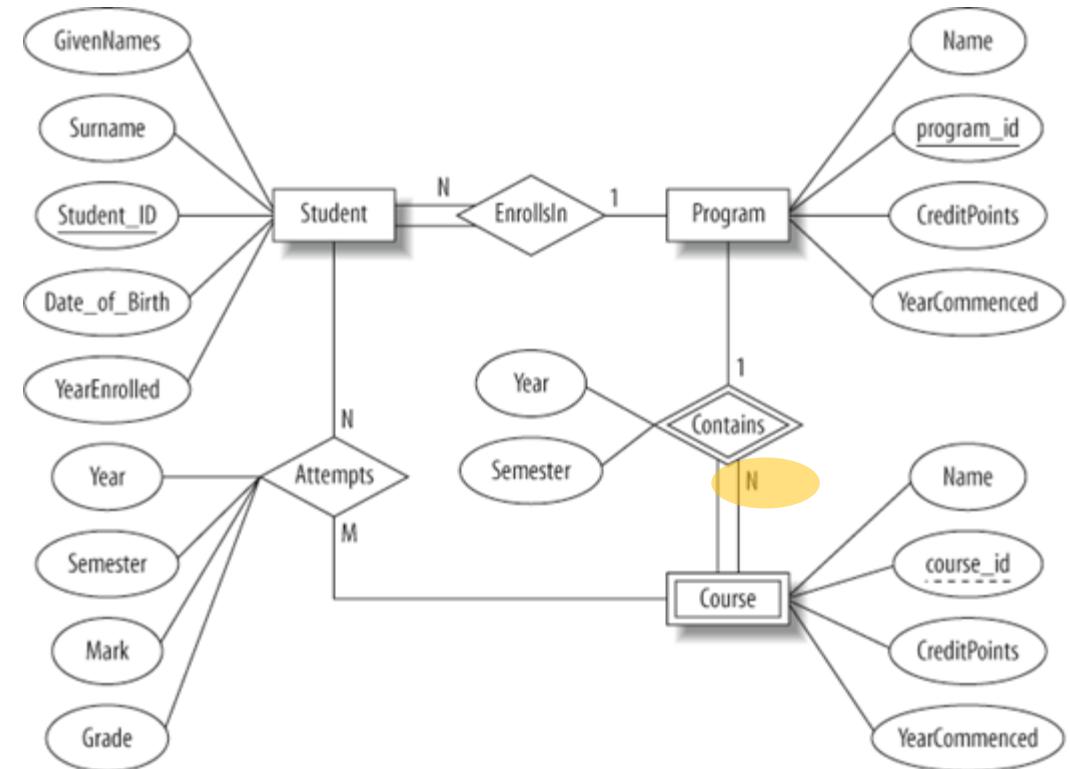
- When a program is deleted, the DBMS automatically deletes all related courses. (weak entity)
 - It has Name, course_id, CreditPoints, YearCommenced where course_id is the weak key.



What have we learnt so far?



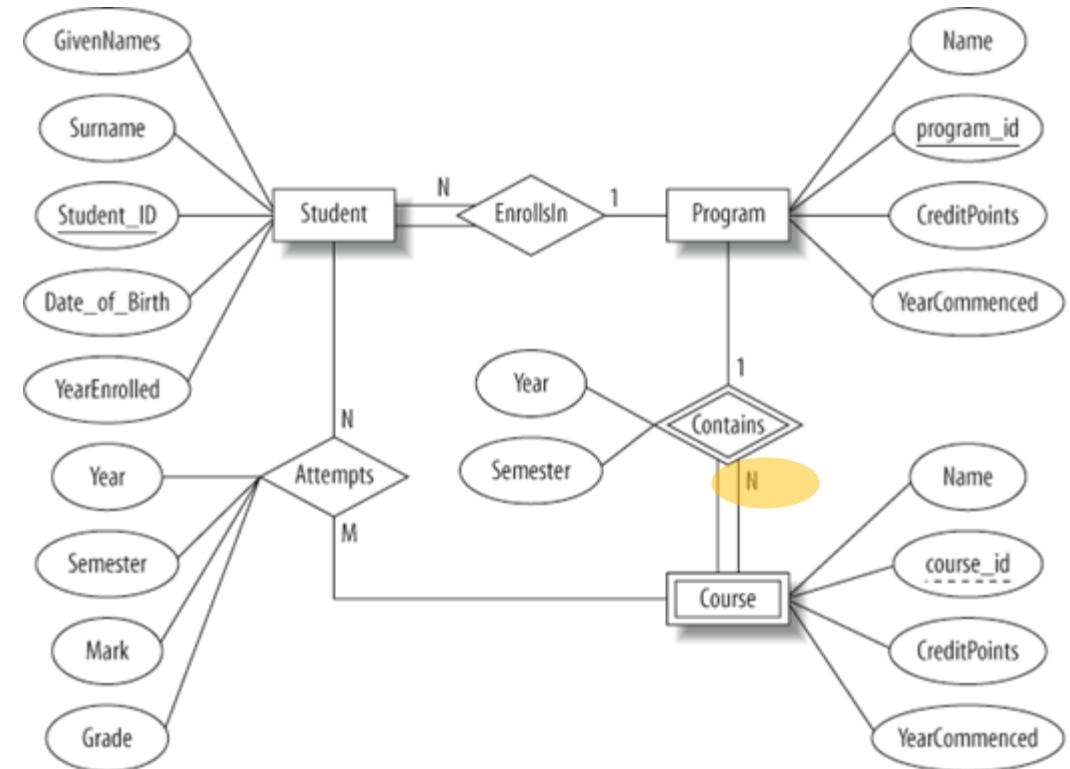
- Program have courses



What have we learnt so far?



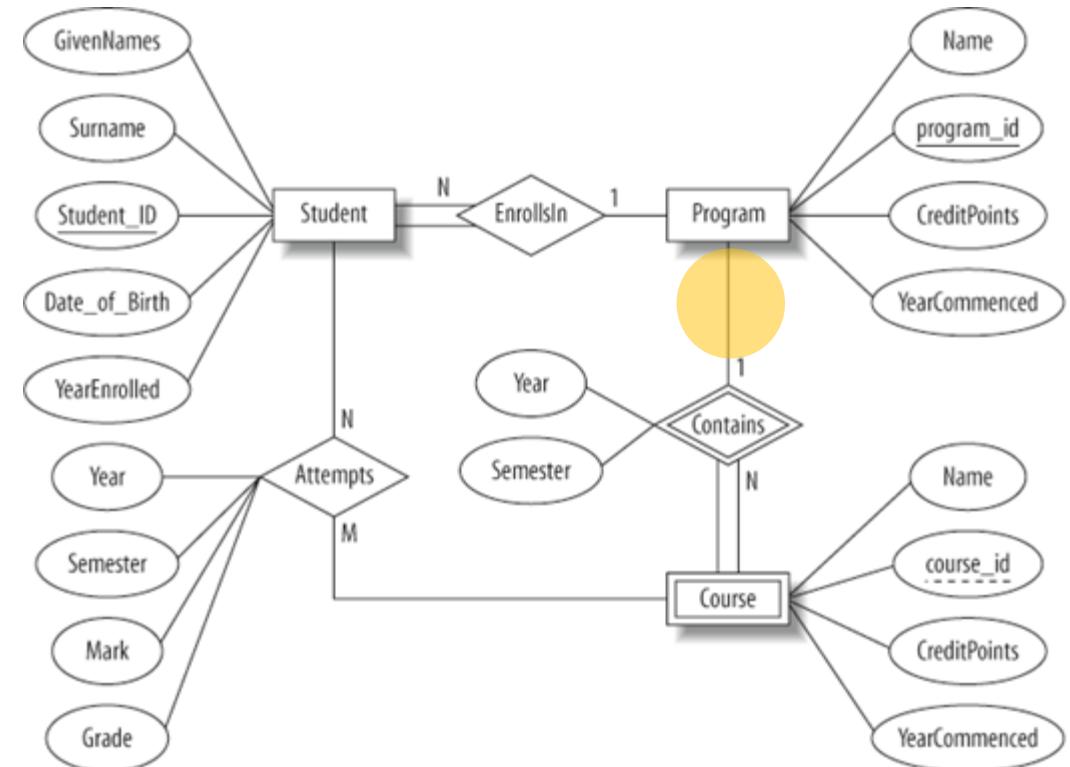
- Program have **many** courses



What have we learnt so far?



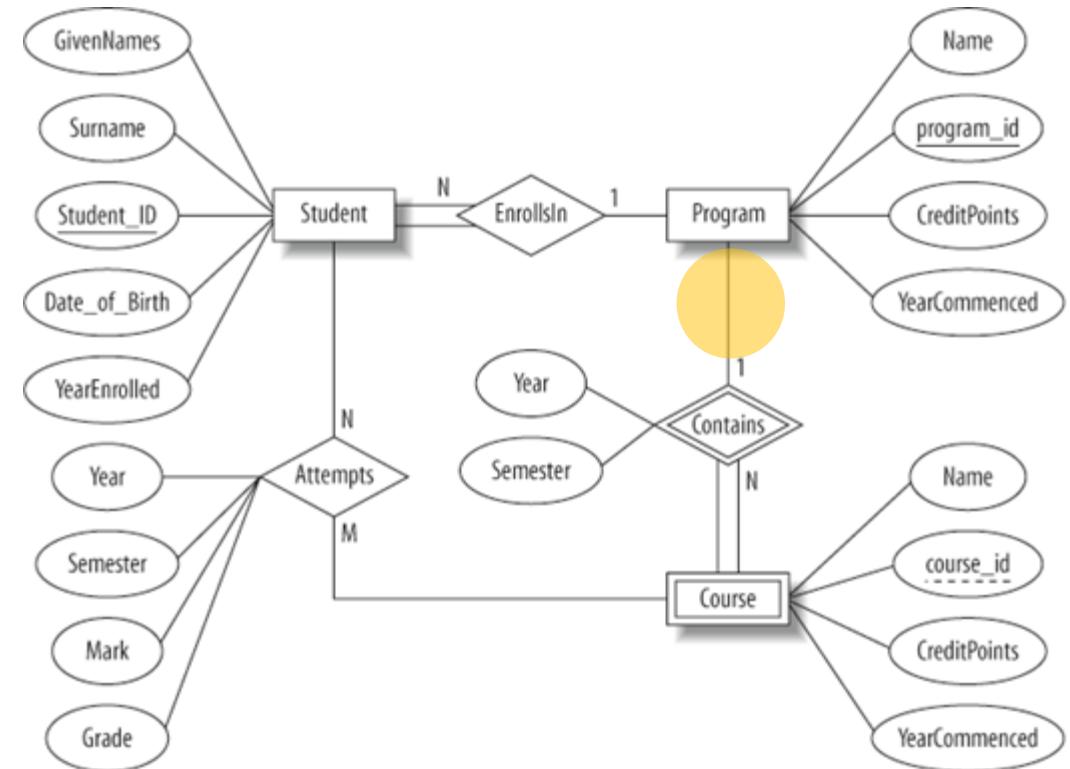
- Program have **many** courses



What have we learnt so far?

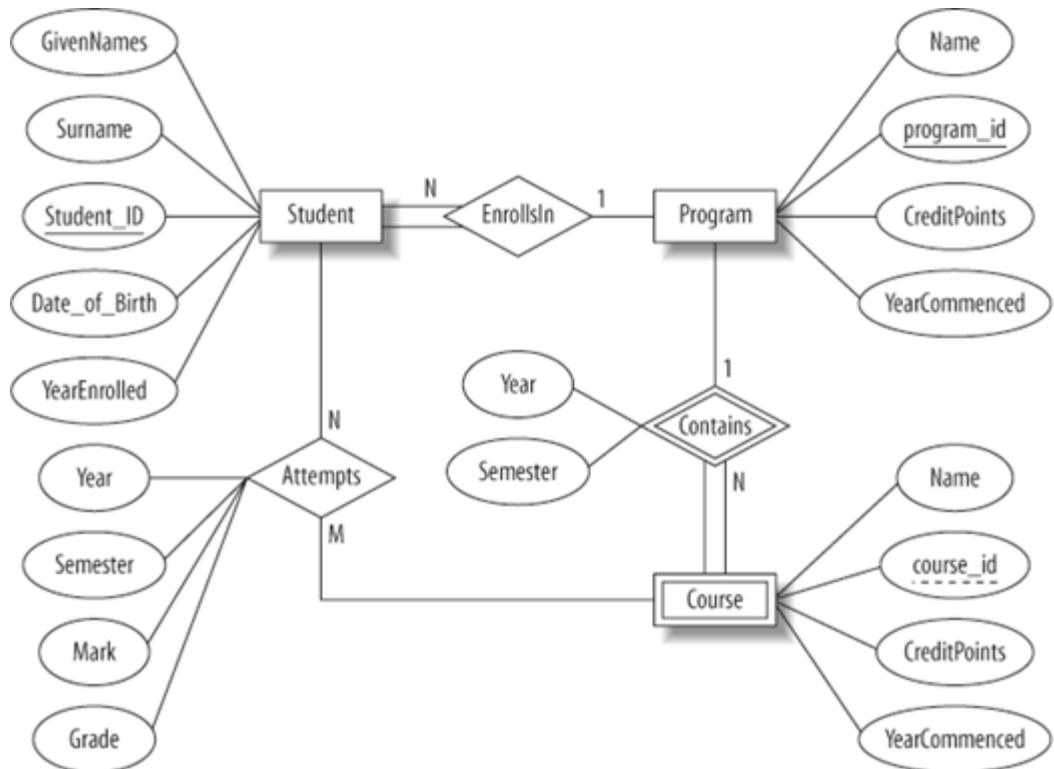


- Program **may** have **many** courses



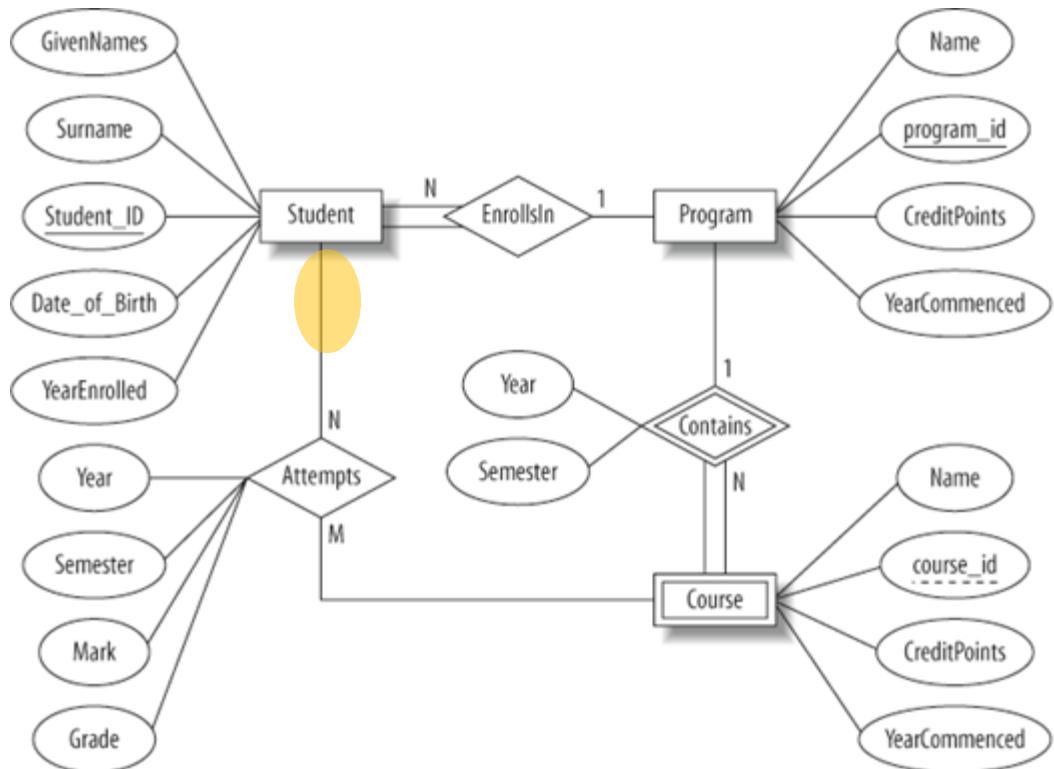
What have we learnt so far?

- A student ... **attempt** ... courses.
- A course .. **be attempted by** students.



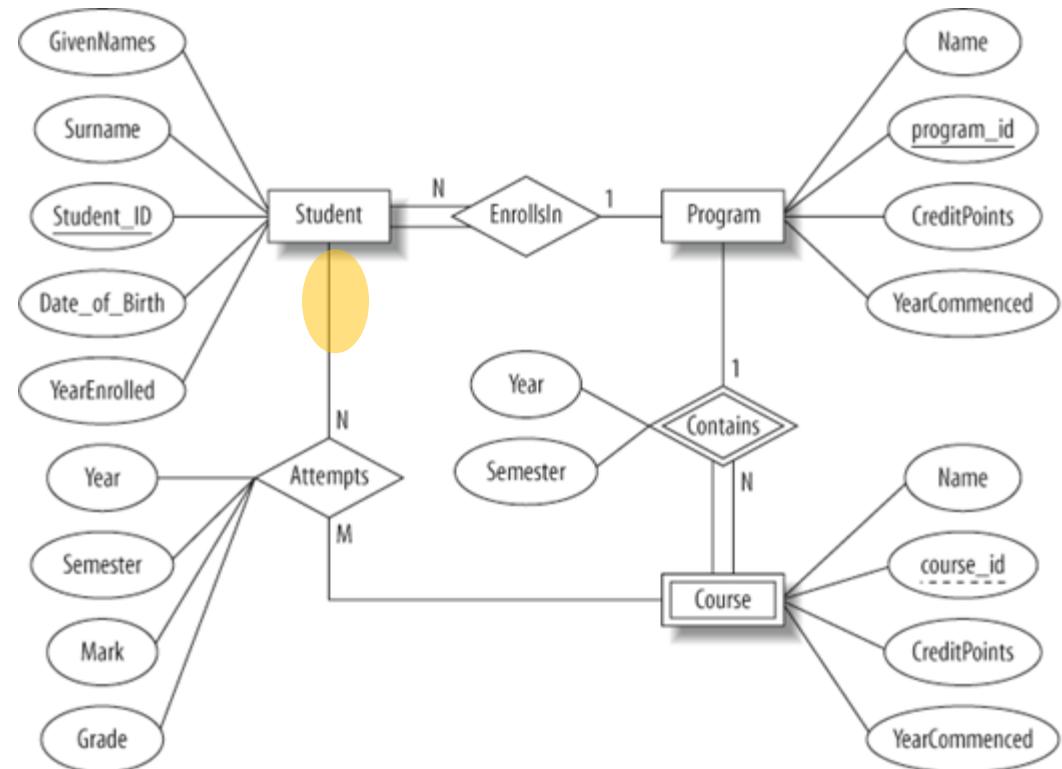
What have we learnt so far?

- A student ... **attempt** ... courses.
- A course .. **be attempted by** students.



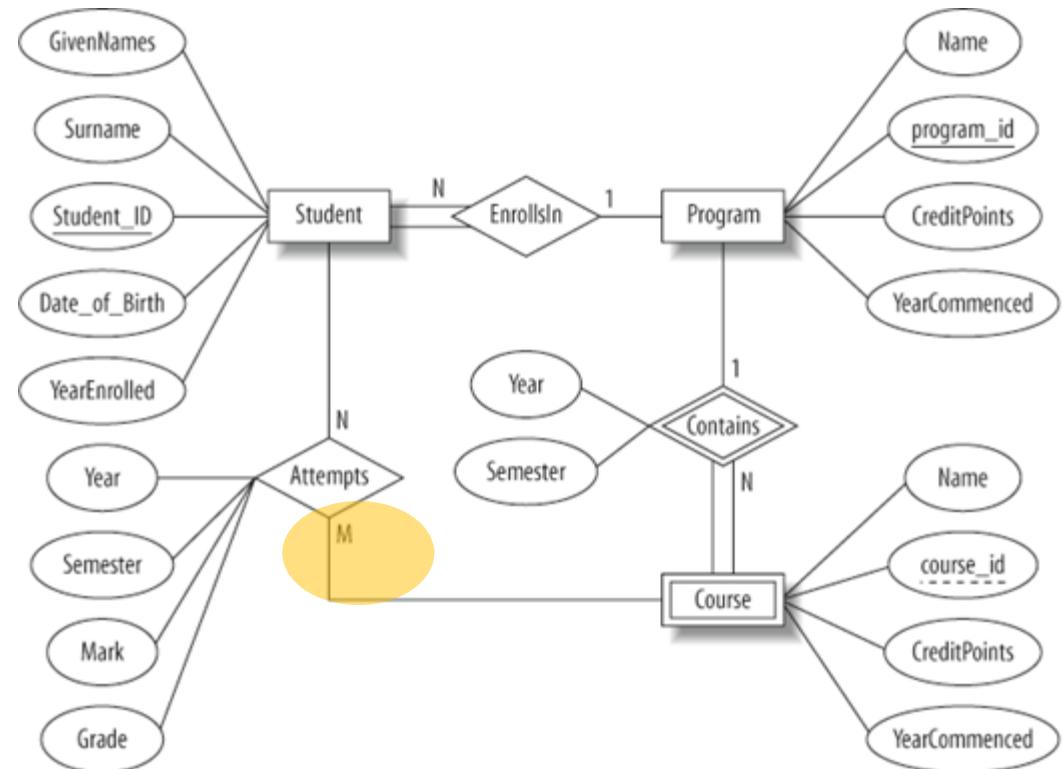
What have we learnt so far?

- A student **may attempt** ... courses.
- A course .. **be attempted by** students.



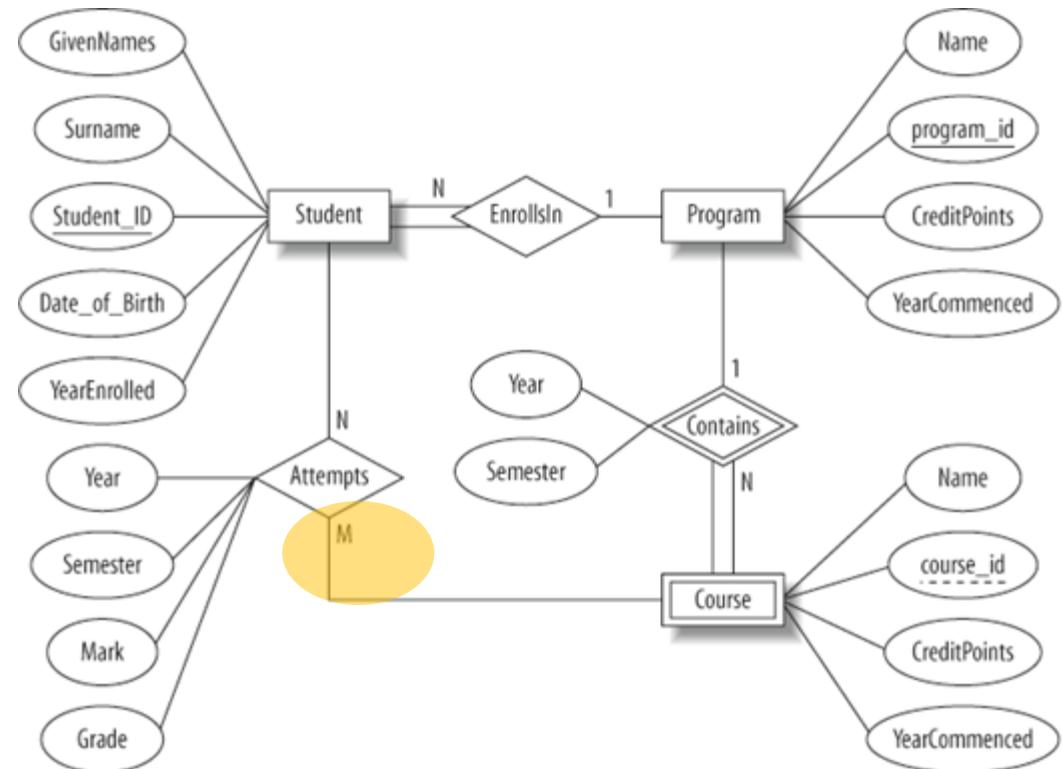
What have we learnt so far?

- A student **may attempt** ... courses.
- A course ... **be attempted by** ... students.



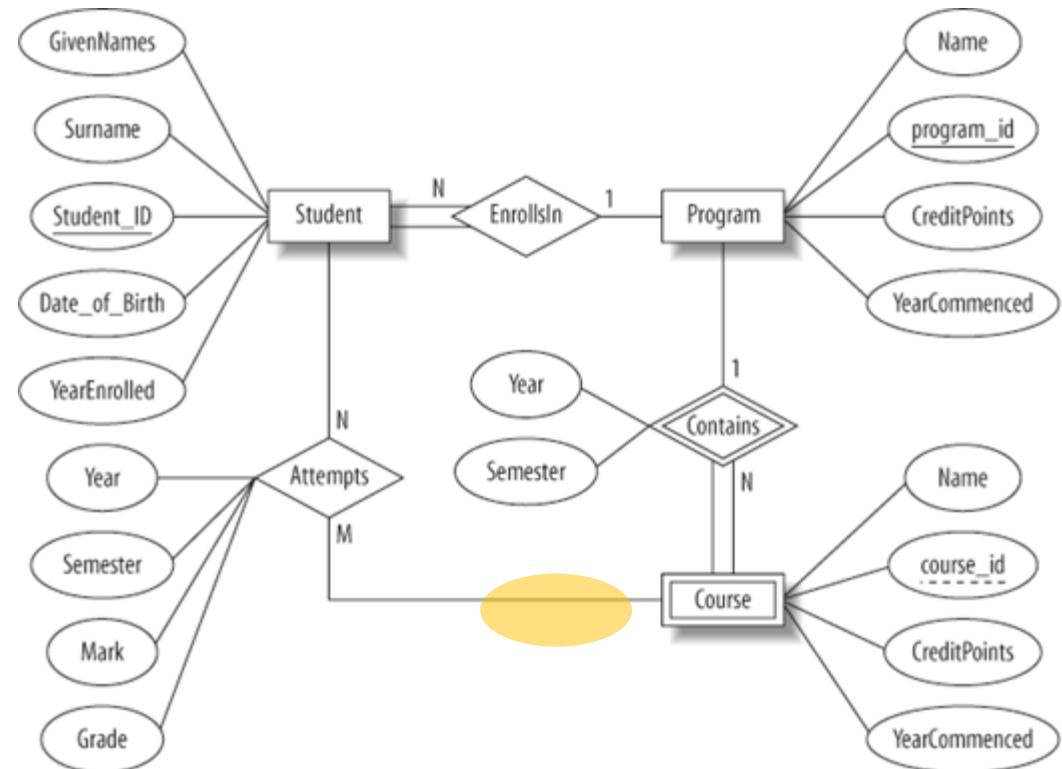
What have we learnt so far?

- A student **may attempt many** courses.
- A course ... **be attempted by ...** students.



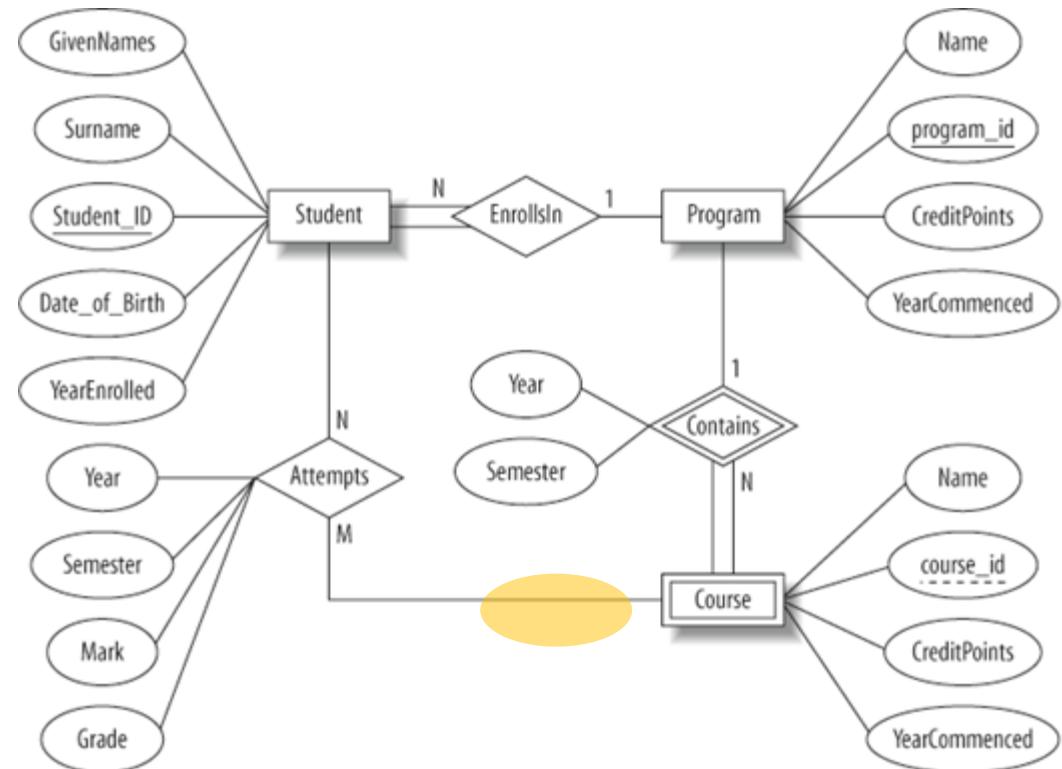
What have we learnt so far?

- A student **may attempt many** courses.
- A course **be attempted by** students.



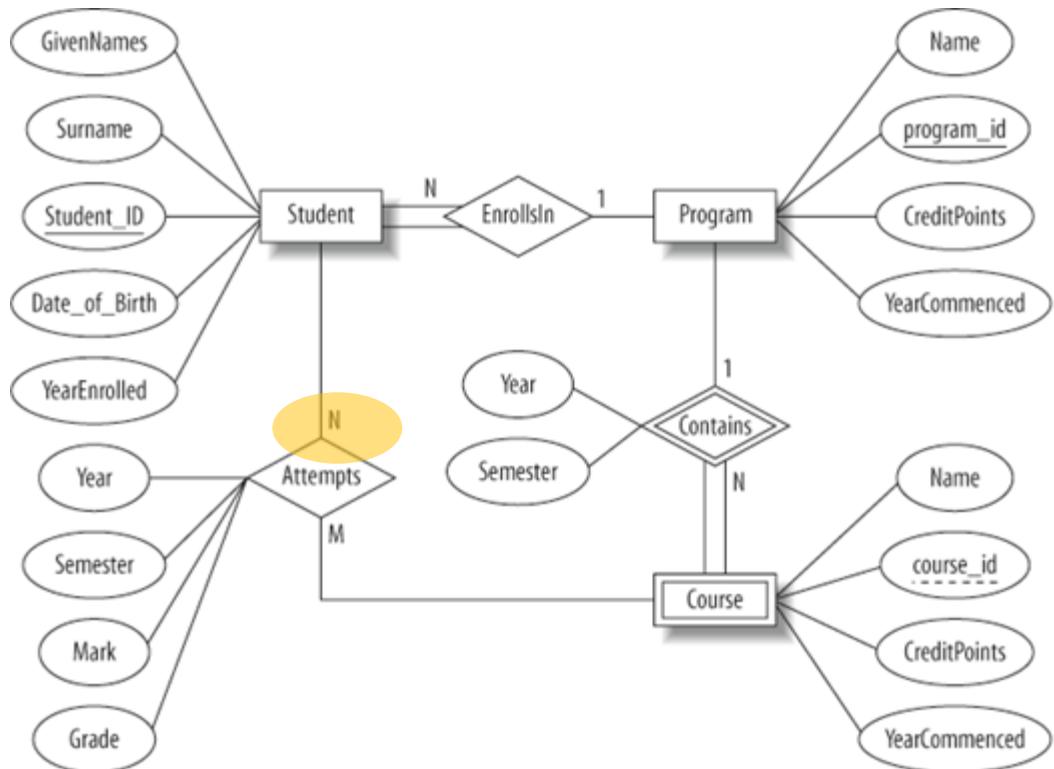
What have we learnt so far?

- A student **may attempt many** courses.
- A course **may be attempted by** students.



What have we learnt so far?

- A student **may attempt many** courses.
- A course **may be attempted by many** students.





Learning Outcomes



- You will learn concepts regarding the relational model, such as:
 - Integrity constraints, key constraints, schema, tuple, domain.
- After this week, you will learn how to
 - Convert a given ERD into the relational model.

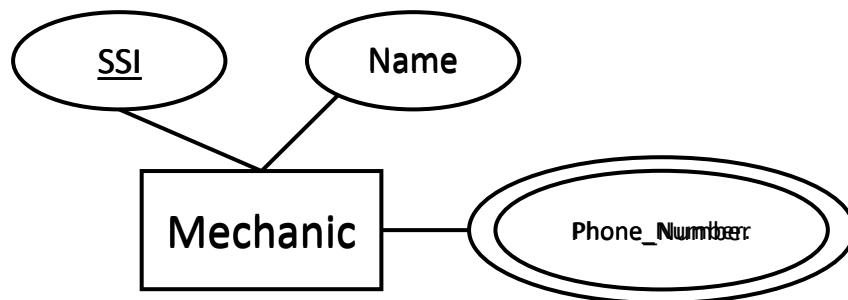
This lecture

- We will start learning the **Relational Model.**
 - Schema and definition.
 - Integrity constraints.
 - Key constraints
 - Referential Integrity

That is, we will start learning correspondence between ERD and Relational database.



- ER diagram
- Relation.

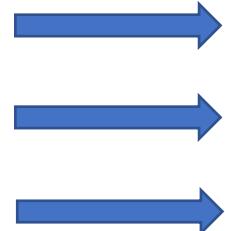


SSI	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237



Relational Database: Definitions

- Instead of Entity Sets and Entities
- We use **relation** as a *set* of rows or *tuples* (i.e., all rows are distinct) with description.



sid	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Shero	shero@cs	18	3.2
53650	Shero	shero@math	19	3.8

Rows/Tuples

Relational Database: Definitions

- *Relation is a mathematical statement*
- *Relation*: made up of 2 parts:
Instance: a *table* with rows and columns.

#Rows = cardinality, #fields/attributes = degree / arity.

sid	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Shero	shero@cs	18	3.2
53650	Shero	shero@math	19	3.8

Relation Schema (RS): specifies the relation's name and **type (domain)** of each **column**.

Write the Relation Schema for the given Relation Instance.

Students(*sid*: INT, *name*: TEXT, *login*: TEXT, *age*: INT, *gpa*: DOUBLE).

Name of the attribute: Domain of the attribute

MySQL DATA TYPES

DATE TYPE	SPEC	DATA TYPE	SPEC
CHAR	String (0 - 255)	INT	Integer (-2147483648 to 214748-3647)
VARCHAR	String (0 - 255)	BIGINT	Integer (-9223372036854775808 to 9223372036854775807)
TINYTEXT	String (0 - 255)	FLOAT	Decimal (precise to 23 digits)
TEXT	String (0 - 65535)	DOUBLE	Decimal (24 to 53 digits)
BLOB	String (0 - 65535)	DECIMAL	"DOUBLE" stored as string
MEDIUMTEXT	String (0 - 16777215)	DATE	YYYY-MM-DD
MEDIUMBLOB	String (0 - 16777215)	DATETIME	YYYY-MM-DD HH:MM:SS
LONGTEXT	String (0 - 4294967295)	TIMESTAMP	YYYYMMDDHHMMSS
LONGBLOB	String (0 - 4294967295)	TIME	HH:MM:SS
TINYINT	Integer (-128 to 127)	ENUM	One of preset options
SMALLINT	Integer (-32768 to 32767)	SET	Selection of preset options
MEDIUMINT	Integer (-8388608 to 8388607)	BOOLEAN	TINYINT(1)



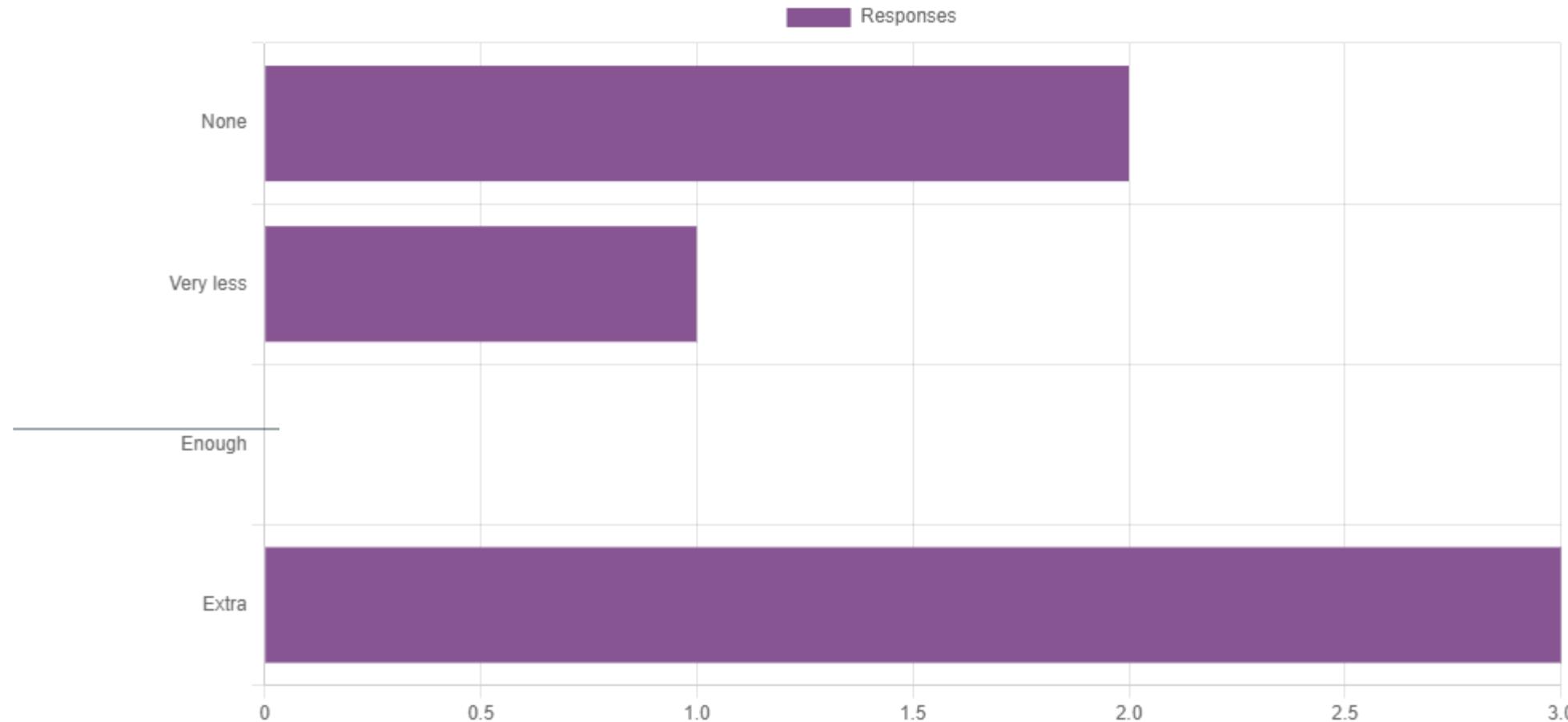
Week 2 ERD to Relational Model

We will begin at 09:00
By Uraz...

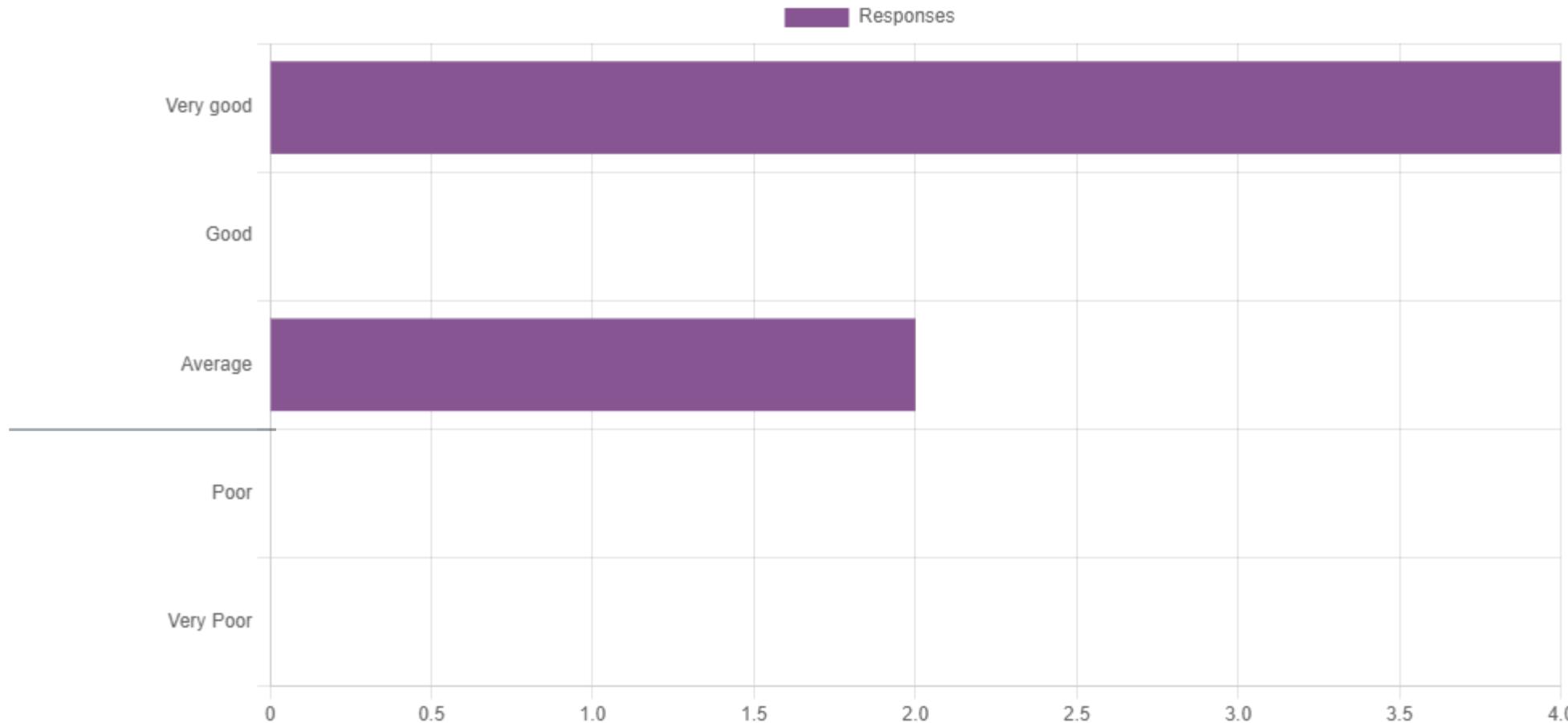


From you...

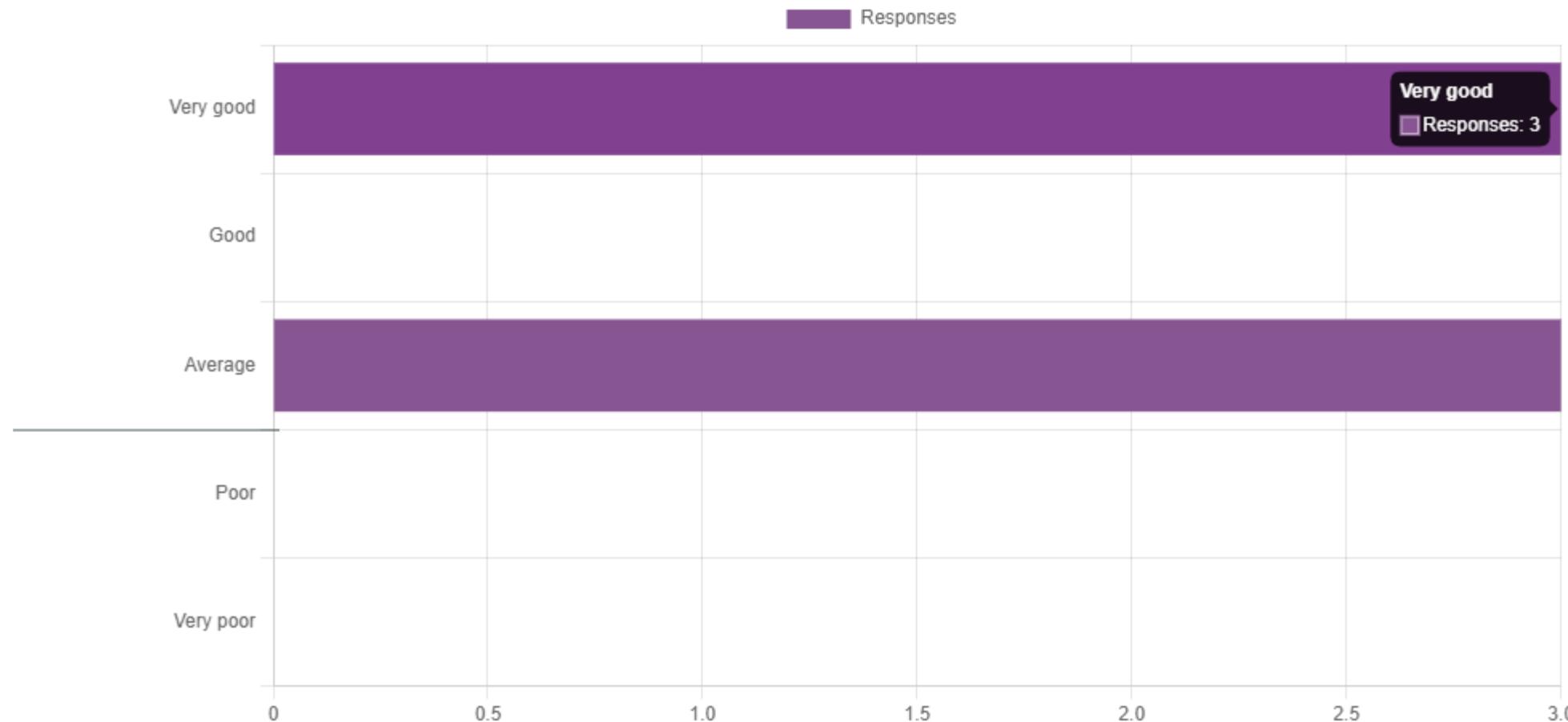
How was the support offered by module convenor?



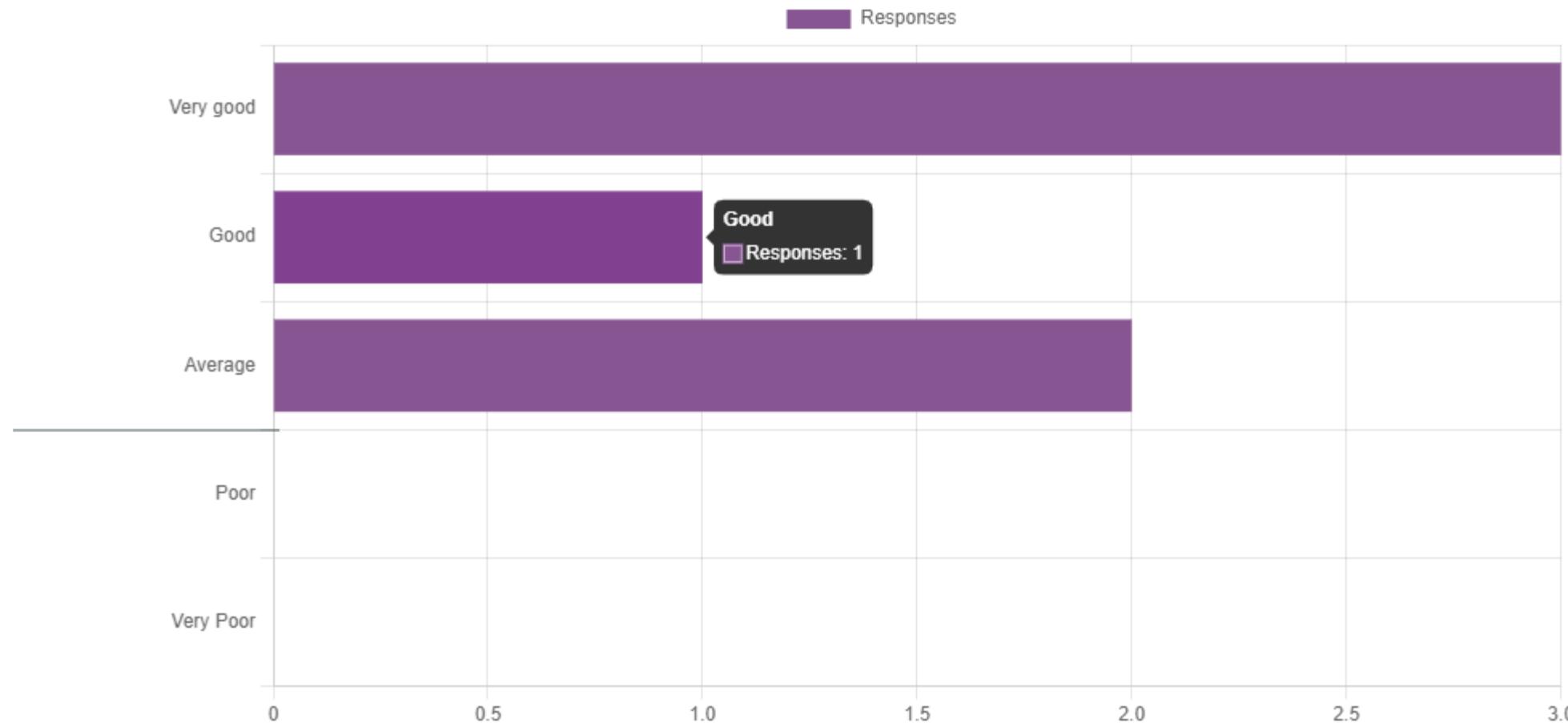
(slides) Helpfulness of the staff?



The module as a whole?



The quality of teaching?





What aspect of the module did you find most useful?

- Use of simple concepts like games to explain abstract ideas
-

What aspect of the module did you find most difficult?

- Notation used (i.e. double lines for total participation, the direction that cardinality is read)
-

What can be improved to help with the module?

- I would like a table with the different combinations & possibilities which make up the sentences for ERD relationships. Eg: "Partial participation constraints = 'at most one' / 'many' "
- Provide coursework examples with the assignment, not 2 weeks later.



What aspect of the module did you find most useful?

- Use of simple concepts like games to explain abstract ideas
-

What aspect of the module did you find most difficult?

- Notation used (i.e. double lines for total participation, the direction that cardinality is read)
-

What can be improved to help with the module?

- I would like a table with the different combinations & possibilities which make up the sentences for ERD relationships. Eg: "Partial participation constraints = 'at most one' / 'many' "
- Provide coursework examples with the assignment, not 2 weeks later.



What aspect of the module did you find most useful?

- Use of simple concepts like games to explain abstract ideas
-

What aspect of the module did you find most difficult?

- Notation used (i.e. double lines for total participation, the direction that cardinality is read)
-

What can be improved to help with the module?

- I would like a table with the different combinations & possibilities which make up the sentences for ERD relationships. Eg: "Partial participation constraints = 'at most one' / 'many' "
- Provide coursework examples with the assignment, not 2 weeks later.



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures. 2. (IMPORTANT) Include practice exercises based on slides for each lecture
 - Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
 - Answer emails whose subject is SCC221_, like the introduction lecture said.
-

Any other comments

-
- 20% of coursework graded on something we aren't being taught?
 - Thanks Uraz
 - Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures. 2. (IMPORTANT) Include practice exercises based on slides for each lecture
 - Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
 - Answer emails whose subject is SCC221_, like the introduction lecture said.
-

Any other comments

-
- 20% of coursework graded on something we aren't being taught?
 - Thanks Uraz
 - Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures. 2.
 - (IMPORTANT) Include practice exercises based on slides for each lecture
 - Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
 - Answer emails whose subject is SCC221_, like the introduction lecture said.
-

Any other comments

-
- 20% of coursework graded on something we aren't being taught?
 - Thanks Uraz
 - Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures. 2. (IMPORTANT) Include practice exercises based on slides for each lecture
- Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
- Answer emails whose subject is SCC221_, like the introduction lecture said.

Any other comments

- 20% of coursework graded on something we aren't being taught?
- Thanks Uraz
- Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures.
 - 2. (IMPORTANT) Include practice exercises based on slides for each lecture
 - Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
 - Answer emails whose subject is SCC221_, like the introduction lecture said.
-

Any other comments

-
- 20% of coursework graded on something we aren't being taught?
 - Thanks Uraz
 - Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures. 2. (IMPORTANT) Include practice exercises based on slides for each lecture
 - Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
 - Answer emails whose subject is SCC221_, like the introduction lecture said.
-

Any other comments

-
- 20% of coursework graded on something we aren't being taught?
 - Thanks Uraz
 - Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures. 2. (IMPORTANT) Include practice exercises based on slides for each lecture
 - Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
 - Answer emails whose subject is SCC221_, like the introduction lecture said.
-

Any other comments

-
- 20% of coursework graded on something we aren't being taught?
 - Thanks Uraz
 - Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures. 2. (IMPORTANT) Include practice exercises based on slides for each lecture
 - Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
 - Answer emails whose subject is SCC221_, like the introduction lecture said.
-

Any other comments

-
- 20% of coursework graded on something we aren't being taught?
 - Thanks Uraz
 - Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."



What more support can be provided?

- 1. Include Exercises done in lectures in the static slides so that they can be done again after the lectures.
 - 2. (IMPORTANT) Include practice exercises based on slides for each lecture
 - Can you make it clear on moodle what session is covering what slides? I find it really confusing since I need to review the content before the lecture
 - Answer emails whose subject is SCC221_, like the introduction lecture said.
-

Any other comments

-
- 20% of coursework graded on something we aren't being taught?
 - Thanks Uraz
 - Get rid of bonus points. Why is identifying errors on a ~100 slide power point worthy of extra credit?

Overall, best module this term :)

- The download for the week 2 slides pdf seems to be broken:
https://modules.lancaster.ac.uk/pluginfile.php/3945298/mod_folder/content/0/Week2.pdf?forcedownload=1
when opened gives the error "Failed to load PDF document."

Relational Database: Definitions

- Relation:** made up of 2 parts:
Instance: a *table* with rows and columns.

#Rows = *cardinality*, #fields/attributes = *degree / arity*.

Relation Schema (RS): specifies the relation's name and **type (domain)** of each **column**.

Write the Relation Schema for the given Relation Instance.

Weapons(*Weapon*: TEXT, *Damage*: INT, *AmmoType*: TEXT,
MaximumAmmo: INT, *FireMode*: TEXT,
RateofFire: INT, *MuzzleVelocity*: DOUBLE, *MaximumRange*: INT).

[https://quake.fandom.com/wiki/Weapon_\(Q1\)](https://quake.fandom.com/wiki/Weapon_(Q1))



Weapon	Damage	Ammo Type	Maximum Ammo	Fire Mode	Rate of Fire (rpm)	Muzzle Velocity (m/s)	Maximum Range (m)
Axe	20	None	∞	Single	120	-	-
Shotgun	24	Shells	100	Pump-action	120	380	78
Super Shotgun	56	Shells	100	Pump-action	85	380	78
Nailgun	9	Nails	200	Automatic	600	110	229
Super Nailgun	18	Nails	200	Automatic	600	158	229
Grenade Launcher	120	Rockets	100	Semi-auto	100	22.8	-

MySQL DATA TYPES

DATE TYPE	SPEC	DATA TYPE	SPEC
CHAR	String (0 - 255)	INT	Integer (-2147483648 to 2147483647)
VARCHAR	String (0 - 255)	BIGINT	Integer (-9223372036854775808 to 9223372036854775807)
TINYTEXT	String (0 - 255)	FLOAT	Decimal (precise to 23 digits)
TEXT	String (0 - 65535)	DOUBLE	Decimal (24 to 53 digits)
BLOB	String (0 - 65535)	DECIMAL	"DOUBLE" stored as string
MEDIUMTEXT	String (0 - 16777215)	DATE	YYYY-MM-DD
MEDIUMBLOB	String (0 - 16777215)	DATETIME	YYYY-MM-DD HH:MM:SS
LONGTEXT	String (0 - 4294967295)	TIMESTAMP	YYYYMMDDHHMMSS
LONGBLOB	String (0 - 4294967295)	TIME	HH:MM:SS
TINYINT	Integer (-128 to 127)	ENUM	One of preset options
SMALLINT	Integer (-32768 to 32767)	SET	Selection of preset options
MEDIUMINT	Integer (-8388608 to 8388607)	BOOLEAN	TINYINT(1)

Relational Database: Definitions

- **Relation:** made up of 2 parts:
Instance: a *table* with rows and columns.

#Rows = *cardinality*, #fields/attributes = *degree / arity*.

Team_ID	Wins	Loses	Name	Leader
223	1	43	Les Miserables	Uraz
14	16	0	Thanos	Thanos
1	23	3	Avengers	Iron Man

Relation Schema (RS): specifies the relation's name and **type (domain)** of each **column**.

Write the Relation Schema for the given Relation Instance.

Teams(*Team_ID*: INT, *Wins*: INT, *Loses*: INT, *Name*: TEXT, *Leader*: TEXT).

MySQL DATA TYPES

DATE TYPE	SPEC	DATA TYPE	SPEC
CHAR	String (0 - 255)	INT	Integer (-2147483648 to 214748-3647)
VARCHAR	String (0 - 255)	BIGINT	Integer (-9223372036854775808 to 9223372036854775807)
TINYTEXT	String (0 - 255)	FLOAT	Decimal (precise to 23 digits)
TEXT	String (0 - 65535)	DOUBLE	Decimal (24 to 53 digits)
BLOB	String (0 - 65535)	DECIMAL	"DOUBLE" stored as string
MEDIUMTEXT	String (0 - 16777215)	DATE	YYYY-MM-DD
MEDIUMBLOB	String (0 - 16777215)	DATETIME	YYYY-MM-DD HH:MM:SS
LONGTEXT	String (0 - 4294967295)	TIMESTAMP	YYYYMMDDHHMMSS
LONGBLOB	String (0 - 4294967295)	TIME	HH:MM:SS
TINYINT	Integer (-128 to 127)	ENUM	One of preset options
SMALLINT	Integer (-32768 to 32767)	SET	Selection of preset options
MEDIUMINT	Integer (-8388608 to 8388607)	BOOLEAN	TINYINT(1)

Exercise...



Relational Database: Definitions

- **Relation:** made up of 2 parts:

Instance: a *table* with rows and columns.

#Rows = *cardinality*, #fields/attributes = *degree / arity*.

Team_ID	Wins	Loses	Name	Leader
223	1	43	Les Miserables	Uraz
14	16	0	Thanos	Thanos
1	23	3	Avengers	Iron Man

Relation Schema (RS): specifies the relation's name and **type (domain)** of each **column**.

Write the Relation Schema for the given Relation Instance.

Teams(*Team_ID*: INT, *Wins*: INT, *Loses*: INT, *Name*: TEXT, *Leader*: TEXT).

What would happen if I made a mistake and entered the following row of data?

<Uraz, 34, 55, A poor lecturer, Uraz>

Relational model (Syntax vs Semantics)



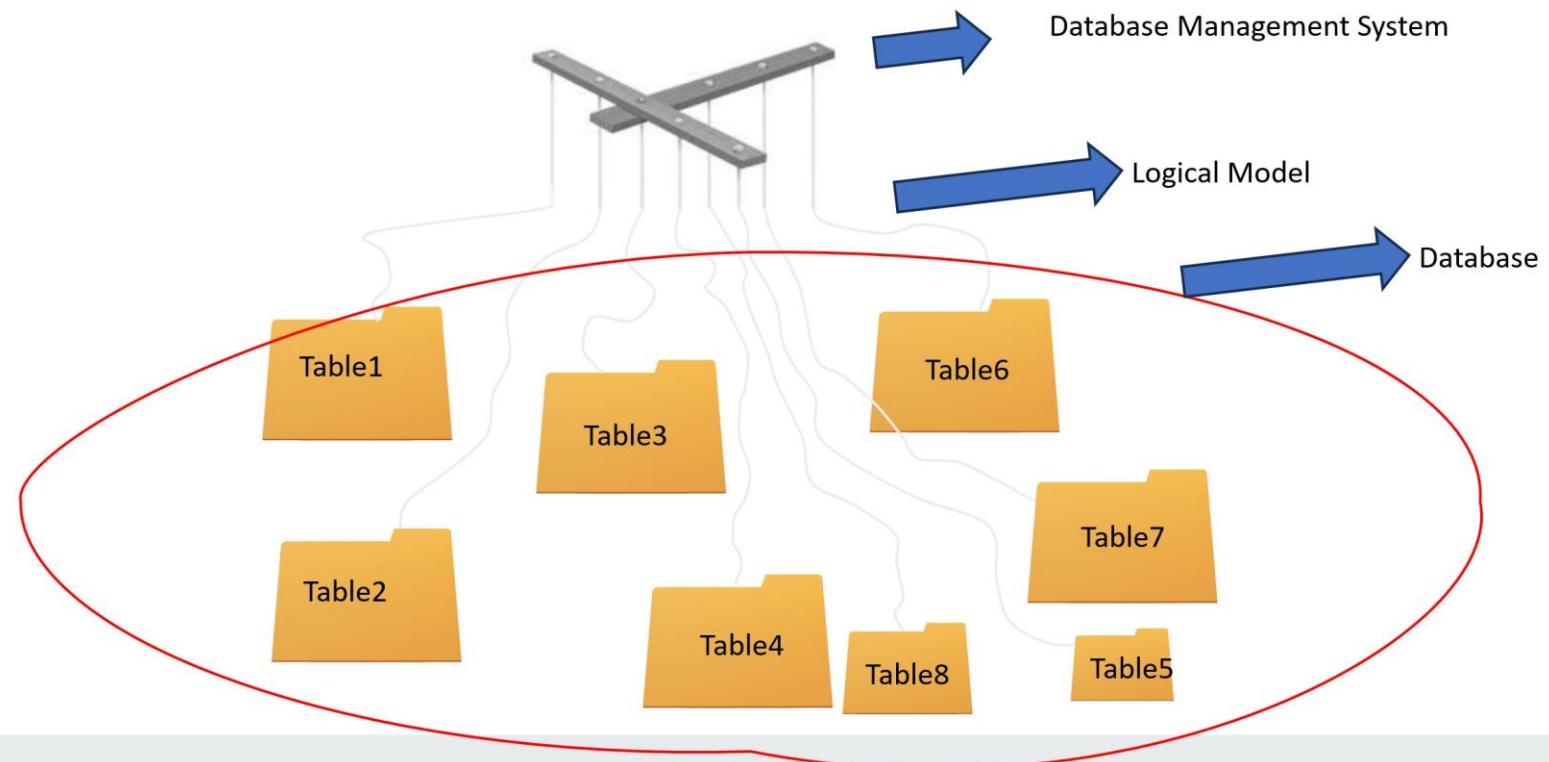
What else can go wrong?

- What would happen if I erase a key value?
- Data kept by a DBMS **must obey some rules!**
- To prevent these, relational databases are built upon logical rules (**constraints**) set by
 - Real-world rules (Context), designers' choices, functions, etc., using DDL.
- These rules establish what we call the **integrity** of the data
- Integrity of the data is protected by DBMS if **INTEGRITY CONSTRAINTS ARE GIVEN.**

Model	Weight	ChassisN	Max_Speed
	1400	12h37	200
Toyota_Corolla	1300	84t34	200
Hyundai E.GLS	1400	43j5h2	210

Relational Database: Definitions

Integrity Constraints are a part of the Logical Model and are provided to the DBMS while tables are created.



Definitions : Integrity Constraints (ICs)



- **Integrity constraint (IC):** a condition that must be true for *any* database instance.
- A *legal* instance of a relation satisfies all specified ICs.
 - DBMS should not allow illegal instances.
- If the DBMS checks ICs, stored data is more faithful to real-world meaning.
 - Avoids data entry errors, too!
- 1) **Domain constraint:** In any database instance, a value of the attribute must be an element of the attribute's domain (*gpa* is a DOUBLE-valued attribute, so we can only store DOUBLE values in related cells) as set in RS.

Students(*sid*: INT, *name*: TEXT, *login*: TEXT, *age*: INT, *gpa*: DOUBLE).

sid	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Shero	shero@cs	18	3.2
53650	Shero	shero@math	19	3.8
"1177"	Uraz	<u>u.turker@cs</u>	39	X

Domain constraint breached
(*gpa* is a DOUBLE value, Na is a TEXT)

Map	Vehicle	Occupants	Type
	Buggy	2	Land
Miramar	PG-117	5	Water
Vikendi	C-130	100	AIR

Definitions : Integrity Constraints (ICs)

2) **Entity Integrity constraint:** A key value cannot be duplicated or left empty in any instance.

- Once a DB Admin sets a key, the *DBMS must inspect every modification* on the data w.r.t the key value.
- DB Admin's task (using DDL) is to set the primary key for the data.
 - DBMS has built-in functions to protect key constraints; to activate those functions, DBA must identify them using DDL (week 4!).
- Write the Relational Schema for this table



PubG(*Map*: TEXT, *Vehicle*: Text, *Occupants*: INT, *Type*: Text, **PRIMARY KEY**: *Map*).

Relational Database: Definitions

- **Relation:** made up of 2 parts:

Instance: a *table* with rows and columns.

#Rows = *cardinality*, #fields/attributes = *degree / arity*.

Weight	cost	owner	explosive	Type
1kg	441	X82jxm	1	1
2kg	123	Fr29x9a	0	3
1kg	223	7ndj2qs	0	12

Relation Schema (RS): specifies the relation's name and **type (domain)** of each **column**.

Write the Relation Schema for the given Relation Instance.

Ammunition(*Weight*: TEXT, *cost*: INT, *owner*: TEXT, *explosive*: BOOLEAN, *Type*: INT, PRIMARY KEY: *owner*).

Relational Database: Definitions

- **Relation:** made up of 2 parts:

Instance: a *table* with rows and columns.

#Rows = *cardinality*, #fields/attributes = *degree / arity*.

Team_ID	Wins	Loses	Name	Leader
223	1	43	Les Miserables	Uraz
14	16	0	Thanos	Thanos
1	23	3	Avengers	Iron Man

Relation Schema (RS): specifies the relation's name and **type (domain)** of each **column**.

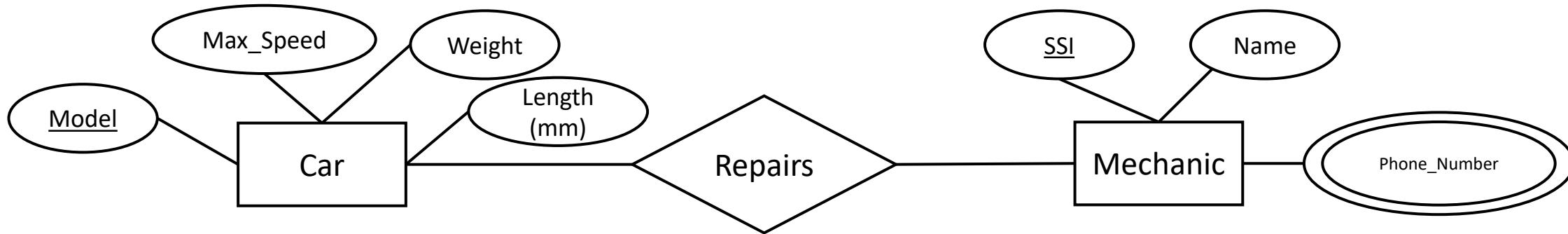
Write the Relation Schema for the given Relation Instance.

Teams(*Team_ID*: INT, *Wins*: INT, *Loses*: INT, *Name*: TEXT, *Leader*: TEXT, PRIMARY KEY: *Team_ID*).

Referential integrity: How about relationships?



- How do we set the integrity of relationships?



Relationship sets

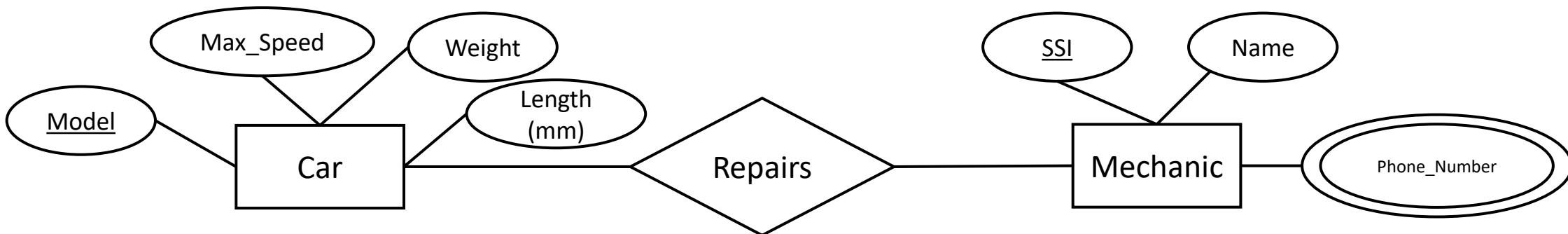
- We inform DBMS regarding the **referential integrity** for relationship sets while creating them using Foreign Keys (in DDL).

Referential integrity: Another key! Foreign keys

- **Foreign keys:** a peripheral attribute that establishes referential integrity between entity sets.

<u>Model</u>	Weight	Length (mm)	Max_Speed
BMW 3.21	1400	2501	200
Toyota_Corolla	1300	3321	200
Hyundai E.GLS	1400	3895	210

<u>SSI</u>	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

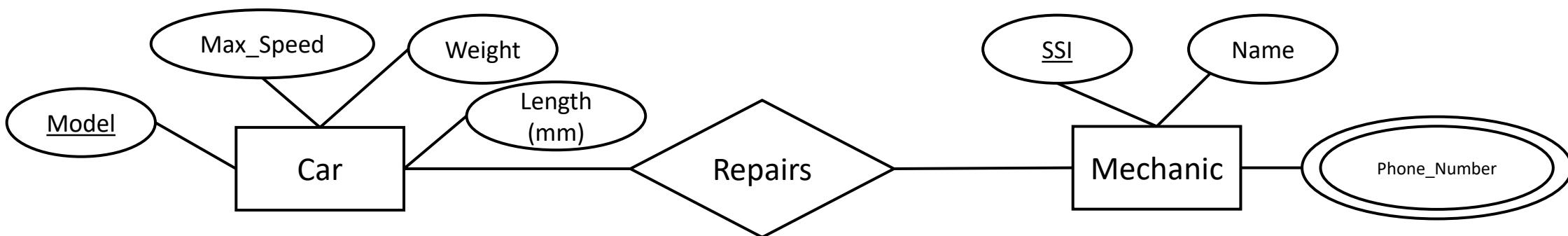


Referential integrity: Another key! Foreign keys

- **Foreign keys:** a peripheral attribute that establishes referential integrity between entity sets.
- It requires either i) importing the Primary Key attribute of one table to the other table

Model	Weight	Length (mm)	Max_Speed
BMW 3.21	1400	2501	200
Toyota_Corolla	1300	3321	200
Hyundai E.GLS	1400	3895	210

SSI	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237



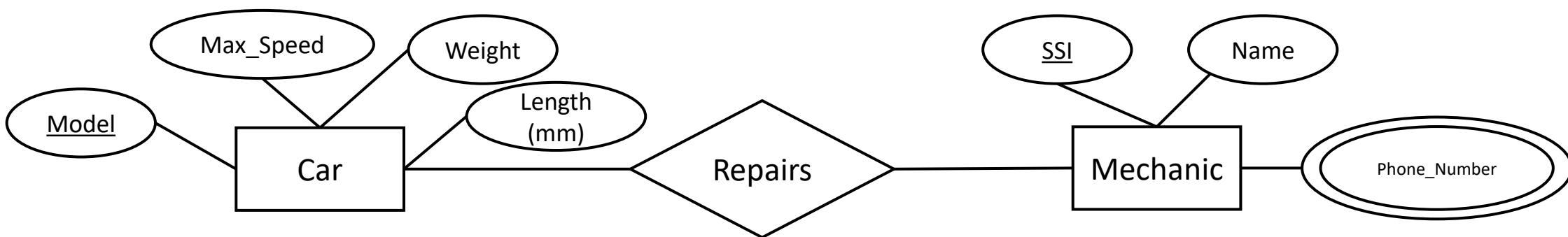
Referential integrity: Another key! Foreign keys

- **Foreign keys:** a peripheral attribute that establishes referential integrity between entity sets.
- It requires either i) importing the Primary Key attribute of one table to the other table



Model	Weight	Length (mm)	Max_Speed
BMW 3.21	1400	2501	200
Toyota_Corolla	1300	3321	200
Hyundai E.GLS	1400	3895	210

SSI	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

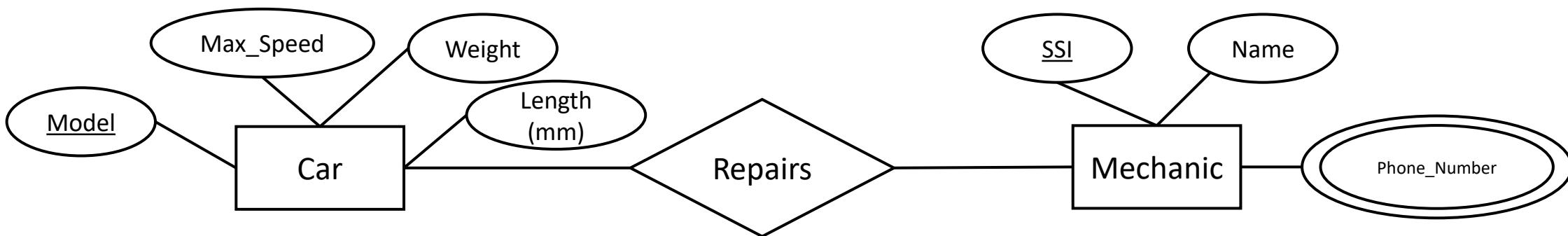


Referential integrity: Another key! Foreign keys

- **Foreign keys:** a peripheral attribute that establishes referential integrity between entity sets.
- It requires either i) importing the Primary Key attribute of one table to the other table

Model	Weight	Length (mm)	Max_Speed
BMW 3.21	1400	2501	200
Toyota_Corolla	1300	3321	200
Hyundai E.GLS	1400	3895	210

Model	SSI	Name	Phone_Number
BMW 3.21	87542702	Tom	75315567, 75315264
Toyota_Corolla	68201937	Uraz	75335521, 75334567
Hyundai E.GLS	23139827	Nick	75315544, 75315237

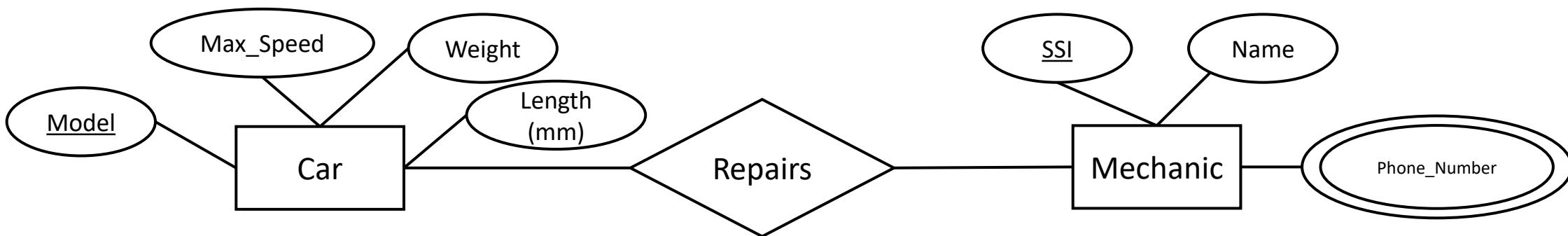


Referential integrity: Another key! Foreign keys

- Values may be in different order!

<u>Model</u>	Weight	Length (mm)	Max_Speed
BMW 3.21	1400	2501	200
Toyota_Corolla	1300	3321	200
Hyundai E.GLS	1400	3895	210

Model	<u>SSI</u>	Name	Phone_Number
Toyota_Corolla	87542702	Tom	75315567, 75315264
Hyundai E.GLS	68201937	Uraz	75335521, 75334567
BMW 3.21	23139827	Nick	75315544, 75315237



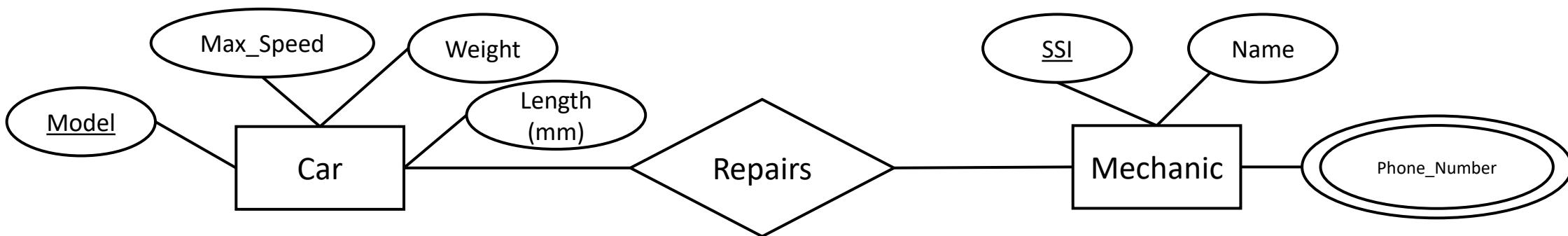
Referential integrity: Another key! Foreign keys

- **Foreign keys:** a peripheral attribute that establishes referential integrity between entity sets.
- It requires either i) importing the Primary Key attribute of one table to the other table or ii) creating a new table that holds the primary keys of the tables in relation.

<u>Model</u>	Weight	Length (mm)	Max_Speed
BMW 3.21	1400	2501	200
Toyota_Corolla	1300	3321	200
Hyundai E.GLS	1400	3895	210

<u>Model</u>	SIS
Toyota_Corolla	87542..
Hyundai E.GLS	68201..
BMW 3.21	2313..

SSI	Name	Phone_Number
87542702	Tom	75315567, 75315264
68201937	Uraz	75335521, 75334567
23139827	Nick	75315544, 75315237

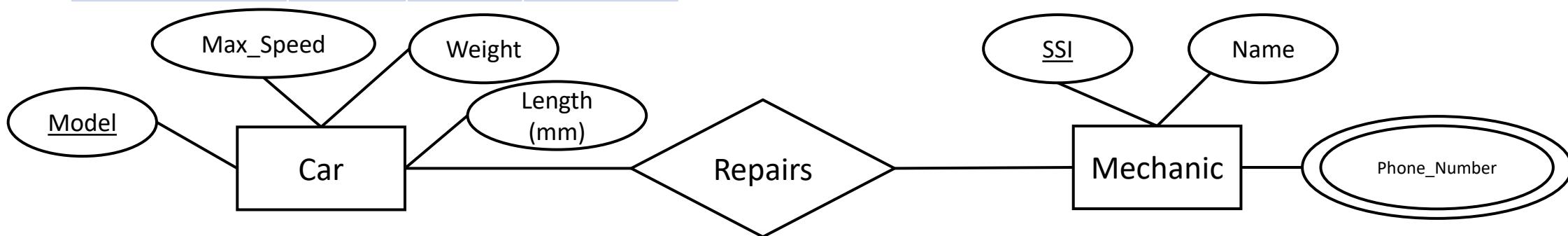


Referential integrity: Foreign keys

- **Foreign key:** ‘logical pointer’.

Model	Weight	Length (mm)	Max_Speed
BMW 3.21	1400	2501	200
Toyota_Corolla	1300	3321	200
Hyundai E.GLS	1400	3895	210

Model	SSI	Name	Phone_Number
Toyota_Corolla	87542702	Tom	75315567, 75315264
Hyundai E.GLS	68201937	Uraz	75335521, 75334567
BMW 3.21	23139827	Nick	75315544, 75315237



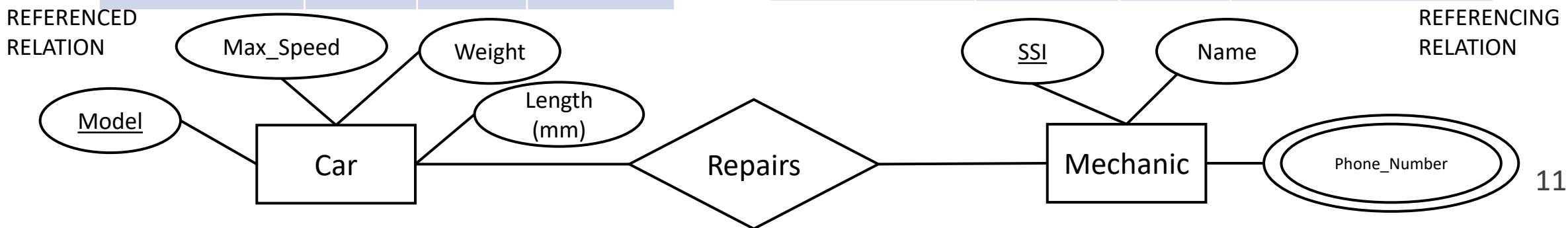
Properties of foreign keys

- Foreign key must:
 - Have the same name and domain/type as the referencing relation.
 - Related entities must have the same values.



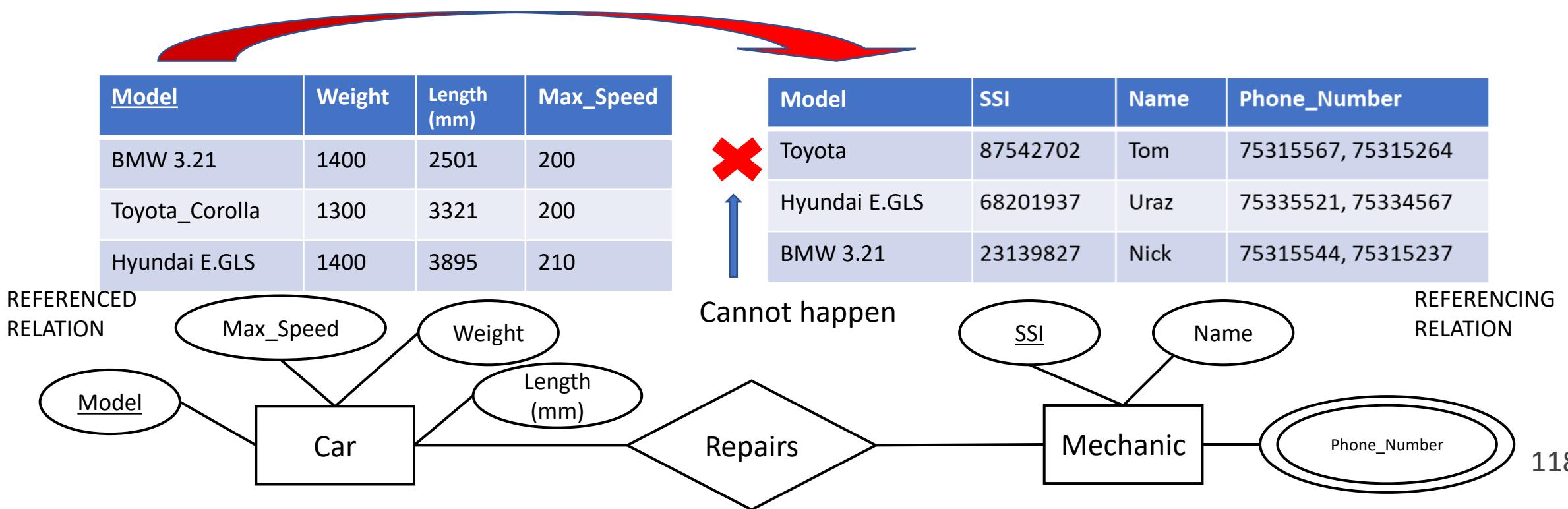
Model	Weight	Length (mm)	Max_Speed
BMW 3.21	1400	2501	200
Toyota_Corolla	1300	3321	200
Hyundai E.GLS	1400	3895	210

Model	SSI	Name	Phone_Number
Toyota_Corolla	87542702	Tom	75315567, 75315264
Hyundai E.GLS	68201937	Uraz	75335521, 75334567
BMW 3.21	23139827	Nick	75315544, 75315237



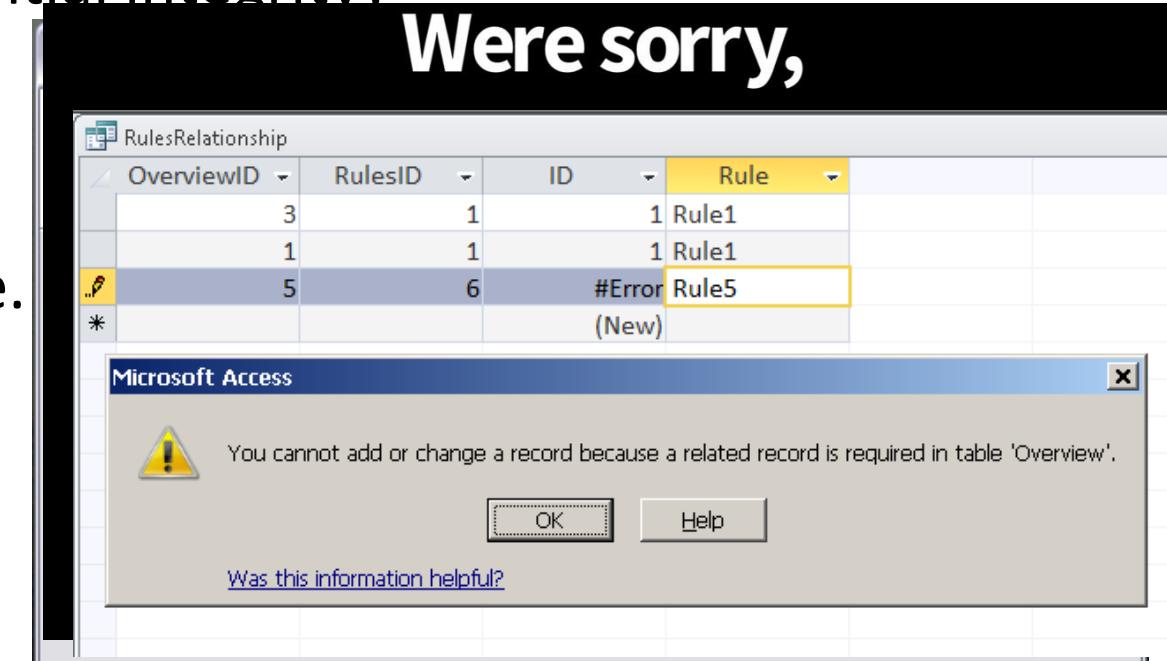
Properties of foreign keys

- Foreign key must:
 - Have the same name and domain/type as the referencing relation.
 - Related entities must have the same values.



Referential integrity: Foreign keys

- If all foreign key constraints are enforced, referential integrity is achieved, i.e., no dangling references, dissimilar values, etc.
- Can you name a data model w/o referential integrity?
- Links in HTML.
- Pointers in C++.
- Phone numbers recorded in your phone.
- In Database.



Foreign Key in action.

Find the total quantity and items for orders of customers living in “Pennsylvania”.

Customers	Customer#	Name	Street	City	Country
	AT01	Alan Turing	Maida Vale	London	UK
	JB01	Jean Bartik	Woodland Walk	Pennsylvania	USA
	MH01	Margaret Hamilton	300 E Street	Pennsylvania	USA
	AL01	Ada Lovelace	Hucknall Road	Nottingham	UK
	EC01	Edgar F. Codd	15 Parks Road	Oxford	UK

Process “Customers”: Find rows with “Pennsylvania” and Get the Key Values.

Process “Orders”: Find Item# and Quantity using keys.

Process “Items”: Find Descriptions using Item#.

Items	Item#	Description	Category
	0001	Hard Disk Drive	Internal Hardware
	0002	16GB RAM	Internal Hardware
	0003	Mechanical Keyboard	Peripherals
	0004	LCD 32" HD Monitor	Display
	0005	2200 RTX GPU 11GB	Internal Hardware

Orders	Order#	Item#	Customer#	Delivery_date	Quantity
	Or0022	0002	MH01	2020-02-10	2
	Or0023	0004	AL01	2020-01-30	1
	Or0024	0001	AT01	2020-02-05	1
	Or0025	0005	JB01	2020-02-06	1
	Or0026	0003	EC01	2020-02-01	3
	Or0027	0004	JB01	2020-02-03	6

->9, LCD32" Monitor, 2200 RTX GPU 11GB, 16GBram.

120

Can you draw the ER diagram for these relations? (May be next week!)

Strategies to enforce Referential Integrity

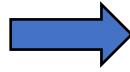
- Consider Students and Enrolled; *sid* in Enrolled is a foreign key that references Students.
- What should be done if an Enrolled tuple with a non-existent student id is inserted?
- ***Reject it!***

Students				
<u>sid</u>	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@eecs	18	3.2
53650	Smith	smith@math	19	3.8

Enrolled	<u>sid</u>	<u>cid</u>	grade
	53666	Carnatic101	C
	53666	Reggae203	B
	53650	Topology112	A
	53666	History105	B
	1177	SCC201	A

Strategies to enforce Referential Integrity

- What should be done if a Students tuple (say 53650) is deleted?



sid	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53650	Shero	shero@eecs	18	3.2
53689	Smith	smith@math	19	3.8

- Also delete all Enrolled tuples that refer to it.
- Disallow deletion of a Students tuple that is referred to.
- Set sid in Enrolled tuples that refer to it to a *default sid*.
- (In SQL, also: Set sid in Enrolled tuples that refer to it to a special value *null*, denoting ‘*unknown*’ or ‘*inapplicable*’.)



sid	cid	grade
53666	Carnatic101	C
53666	Reggae203	B
53650	Topology112	A
53666	History105	B

- What happens if the primary key of the Students tuple is updated (53650 to 00001)?

Referential Integrity in SQL/92

- SQL/92 supports all 4 options on deletes and updates.
 - Default is **NO ACTION** (*delete/update is rejected*)
 - **CASCADE** (also delete all tuples that refer to deleted tuple)
 - **SET NULL / SET DEFAULT** (sets foreign key value of referencing tuple)

Lets speculate about ICs. for the following tables.

Customers	Customer#	Name	Street	City	Country
	AT01	Alan Turing	Maida Vale	London	UK
	JB01	Jean Bartik	Woodland Walk	Pennsylvania	USA
	MH01	Margaret Hamilton	300 E Street	Pennsylvania	USA
	AL01	Ada Lovelace	Hucknall Road	Nottingham	UK
	EC01	Edgar F. Codd	15 Parks Road	Oxford	UK

Assume these tables.

Do tables obey key constraints?

Do tables obey domain constraints?

What are the foreign keys?

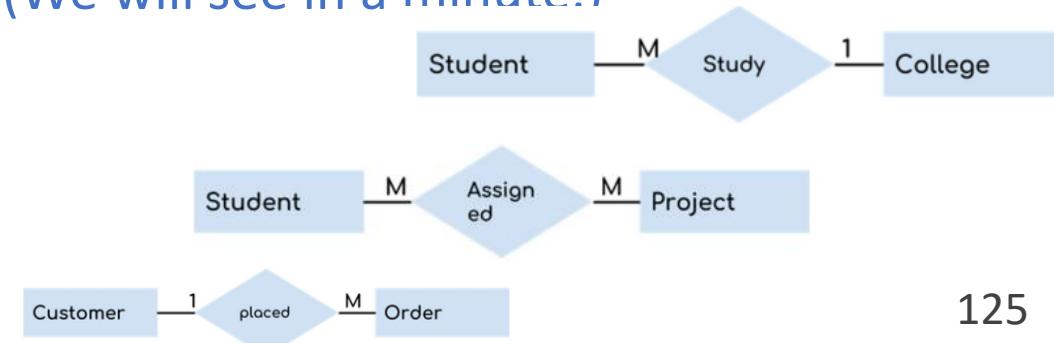
“Print the orders of Customers whose Customer number starts with A?”

Items	Item#	Description	Category
	0001	Hard Disk Drive	Internal Hardware
	0002	16GB RAM	Internal Hardware
	0003	Mechanical Keyboard	Peripherals
	0004	LCD 32" HD Monitor	Display
	0005	2200 RTX GPU 11GB	Internal Hardware

Orders	Order#	Item#	Customer#	Delivery_date	Quantity
	Or0022	0002	MH01	2020-02-10	2
	Or0023	0004	AL01	2020-01-30	1
	Or0024	0001	AT01	2020-02-05	1
	Or0025	0005	JB01	2020-02-06	1
	Or0026	0003	EC01	2020-02-01	3
	Or0027	0004	JB01	2020-02-03	6

Integrity Constraints for relational databases.

- **Integrity of data:** it is the state of data in which data obeys the constraints set by DBA.
- 1) Domain constraints.
 - Values in tuples should obey types of attributes. (You should not provide text to INT field)
- 2) Entity Integrity constraints.
 - Keys of a relation must be **unique**, **non-redundant**, and **not Null** (entity integrity constraint).
- 3) Referential integrity.
 - How are two relations related to each other? (We will see in a minute.)
 - The relation must be made by using keys,
 - The DBMS must preserve this relation.



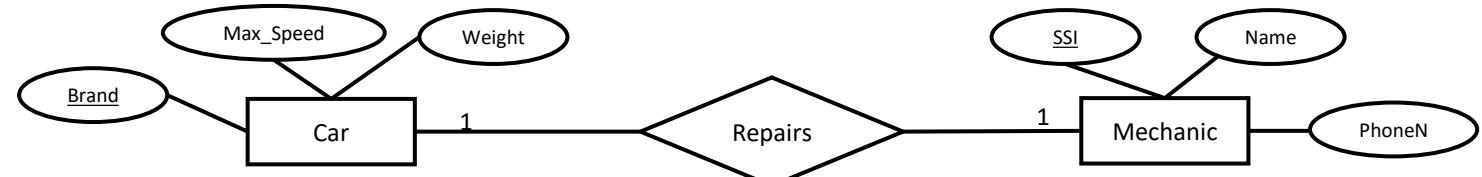


-
- EXTREMELY IMPORTANT CONTENT A HEAD!!!

How do we derive Foreign Keys and ICs for different relationship types?



"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."

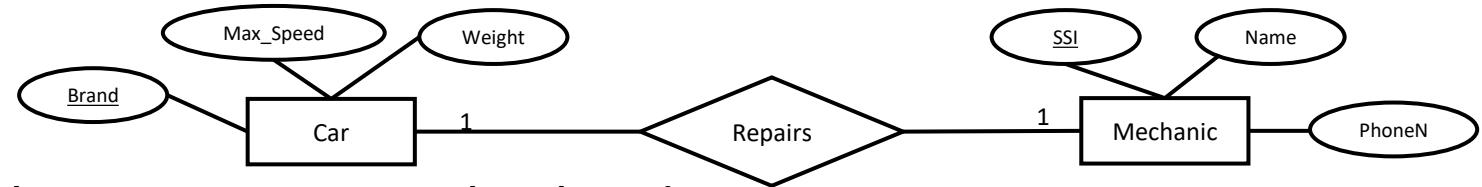


- 1 to 1 relation having partial participation on both sides:

How do we derive Foreign Keys and ICs for different relationship types?



"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."

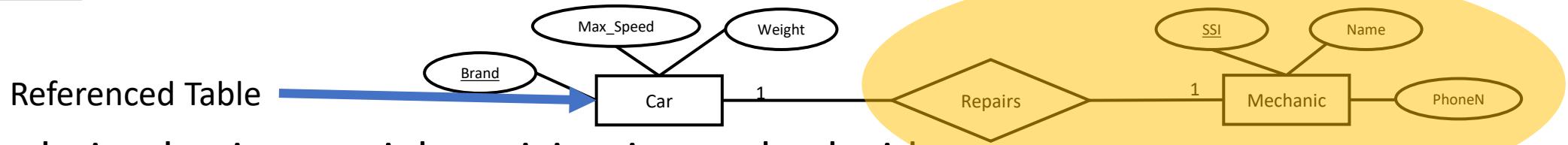


- 1 to 1 relation having partial participation on both sides:
- Select one table (randomly) as the referenced table and the other as the referencing table.
-
-
-
-
-

How do we derive Foreign Keys and ICs for different relationship types?



"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."

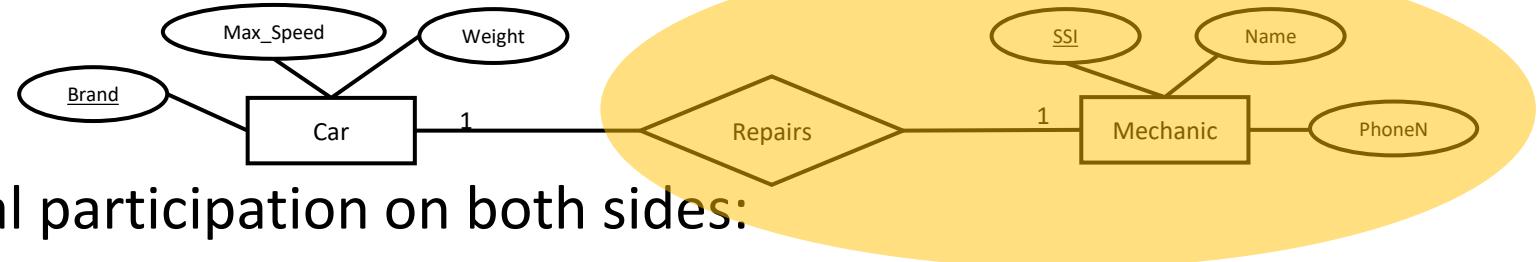


- 1 to 1 relation having partial participation on both sides:
- Select one table (randomly) as the referenced table and the other as the referencing table.
-
-
-
-
-

How do we derive Foreign Keys and ICs for different relationship types?



"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."

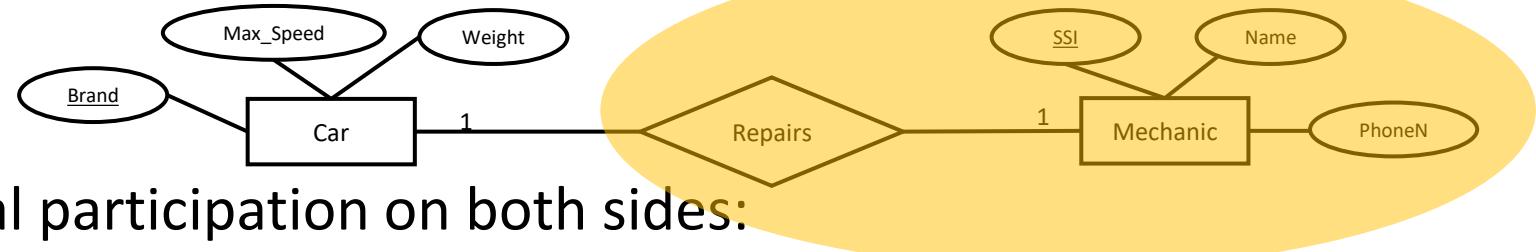


- 1 to 1 relation having partial participation on both sides:
- Select one table (randomly) as the referenced table and the other as the referencing table.
- Import the primary key of the referenced table to the referencing one.
-
-
-
-

How do we derive Foreign Keys and ICs for different relationship types?



"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."

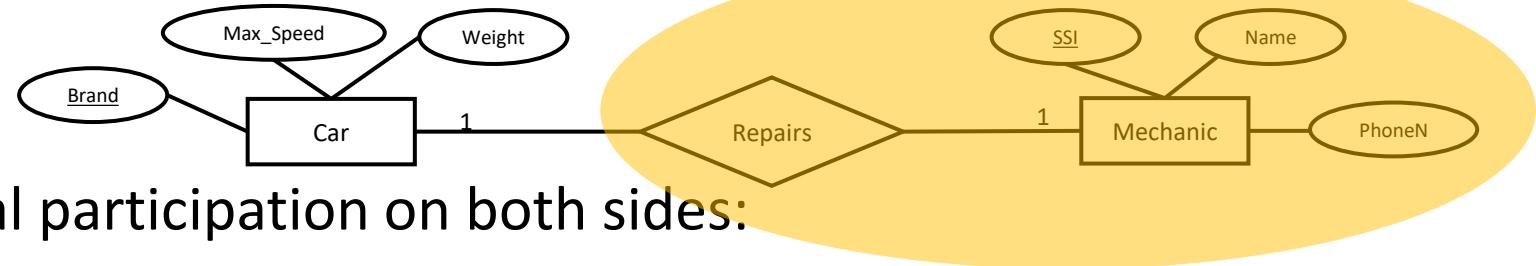


- 1 to 1 relation having partial participation on both sides:
- Select one table (randomly) as the referenced table and the other as the referencing table.
- Import the primary key of the referenced table to the referencing one.
- This key will be the foreign key and declare it its foreign key:
 -
 -
 -

How do we derive Foreign Keys and ICs for different relationship types?



"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."

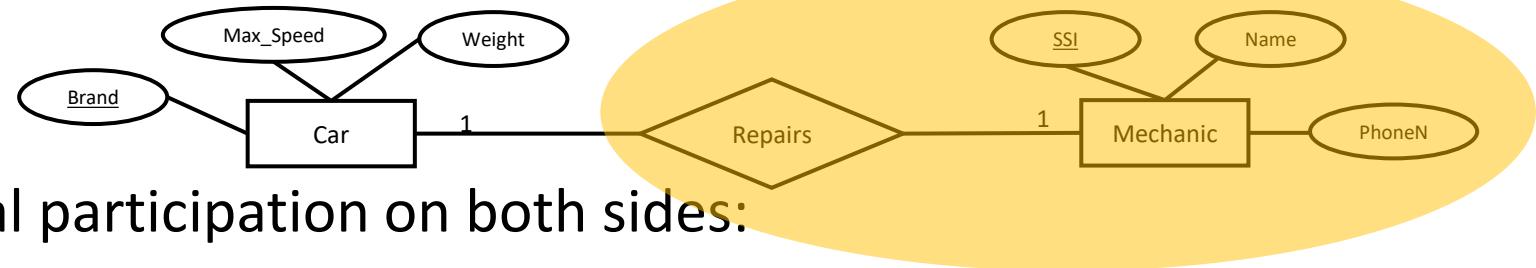


- 1 to 1 relation having partial participation on both sides:
- Select one table (randomly) as the referenced table and the other as the referencing table.
- Import the primary key of the referenced table to the referencing one.
- This key will be the foreign key and declare it its foreign key:
 - *Car(Brand:TEXT,Weight:INT,Length:DOUBLE,Max_Speed:INT, PRIMARY KEY:BRAND)*
 - *Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT, PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR)*
-
-

How do we derive Foreign Keys and ICs for different relationship types?



"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."

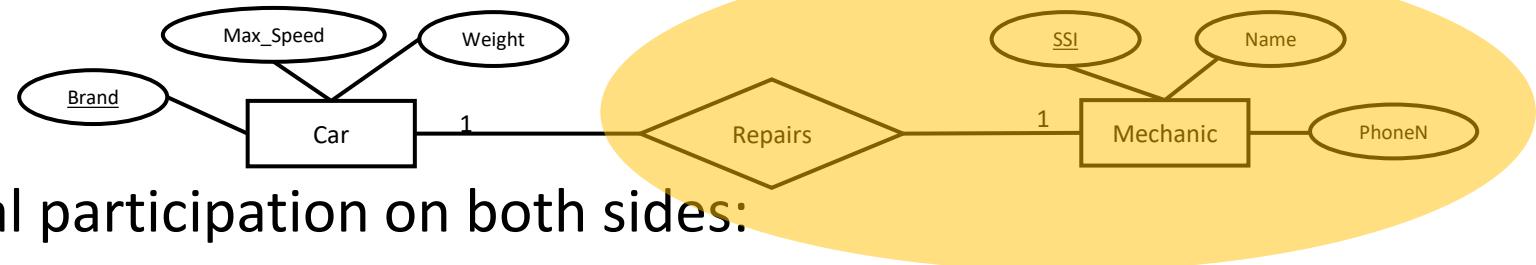


- 1 to 1 relation having partial participation on both sides:
- Select one table (randomly) as the referenced table and the other as the referencing table.
- Import the primary key of the referenced table to the referencing one.
- This key will be the foreign key and declare it its foreign key:
 - *Car(Brand:TEXT,Weight:INT,Length:DOUBLE,Max_Speed:INT, PRIMARY KEY:BRAND)*
 - *Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT, PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR)*
-
-

How do we derive Foreign Keys and ICs for different relationship types?



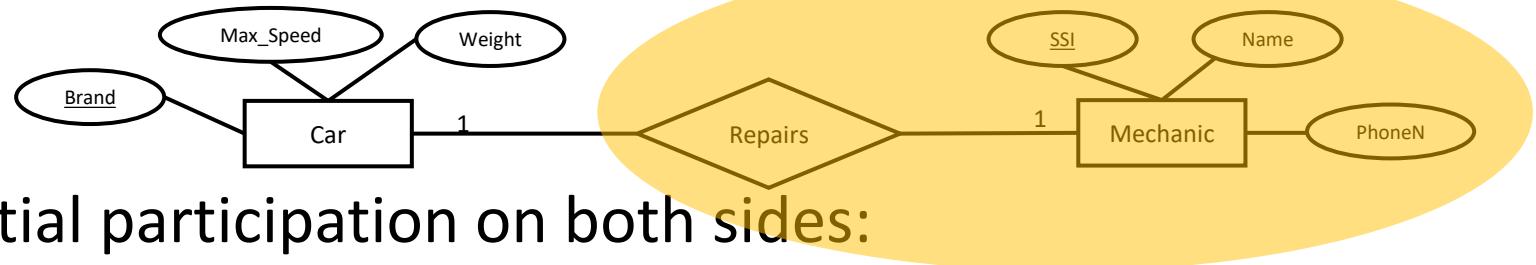
"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



- 1 to 1 relation having partial participation on both sides:
- Select one table (randomly) as the referenced table and the other as the referencing table.
- Import the primary key of the referenced table to the referencing one.
- This key will be the foreign key and declare it its foreign key:
 - *Car(Brand:TEXT,Weight:INT,Length:DOUBLE,Max_Speed:INT, PRIMARY KEY:BRAND)*
 - *Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT, PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR)*
- Do you think that this is enough?
 -

How do we derive Foreign Keys and ICs for different relationship types?

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



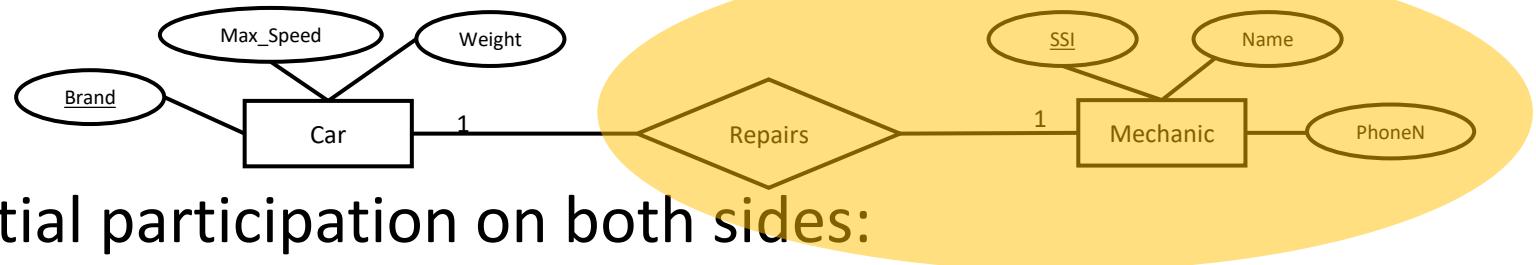
- 1 to 1 relation having partial participation on both sides:
 - *Car(Brand:TEXT,Weight:INT,Length:DOUBLE,Max_Speed:INT, PRIMARY KEY:BRAND)*
 - *Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT, PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR)*

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

SSI	Name	Phone_Number	Brand
87542702	Tom	75315567	Toyota_Corolla
68201937	Uraz	75335521	Hyundai E.GLS
23139827	Nick	75315544	BMW 3.21

How do we derive Foreign Keys and ICs for different relationship types?

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



- 1 to 1 relation having partial participation on both sides:
 - *Car(Brand:TEXT,Weight:INT,Length:DOUBLE,Max_Speed:INT, PRIMARY KEY:BRAND)*
 - *Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT, PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR)*

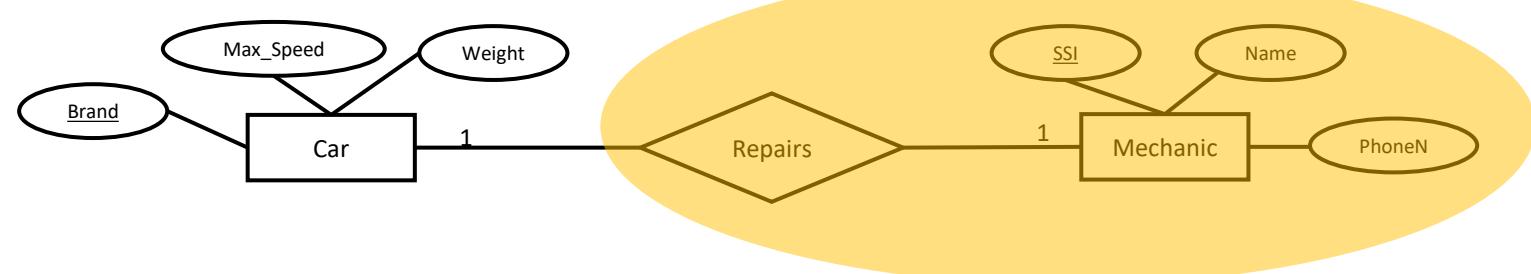
Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

SSI	Name	Phone_Number	Brand
87542702	Tom	75315567	Toyota_Corolla
68201937	Uraz	75335521	Hyundai E.GLS
23139827	Nick	75315544	BMW 3.21
43279823	BMW 3.21	75352362	Hyundai E.GLS

How do we derive Foreign Keys and ICs for different relationship types?

- 1 to 1 relations

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210



SSI	Name	Phone_Number	Brand
87542702	Tom	75315567	Toyota_Corolla
68201937	Uraz	75335521	Hyundai E.GLS
23139827	Nick	75315544	BMW 3.21

Car(Brand:TEXT,Weight:INT,Length:DOUBLE,
Max_Speed:INT, PRIMARY KEY:BRAND)

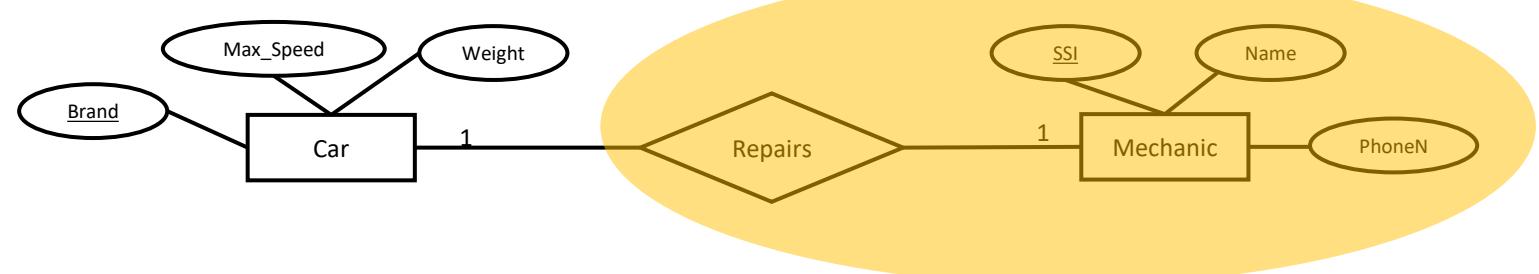
Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT,
PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR
)

The repairs relation is one-to-one. Therefore, for every SSI, there must exist one Brand.
Moreover, as Brand cannot repeat, we use the **UNIQUE** keyword.

How do we derive Foreign Keys and ICs for different relationship types?

- 1 to 1 relations

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210



SSI	Name	Phone_Number	Brand
87542702	Tom	75315567	Toyota_Corolla
68201937	Uraz	75335521	Hyundai E.GLS
23139827	Nick	75315544	BMW 3.21

Car(Brand:TEXT,Weight:INT,Length:DOUBLE,
Max_Speed:INT, PRIMARY KEY:BRAND)

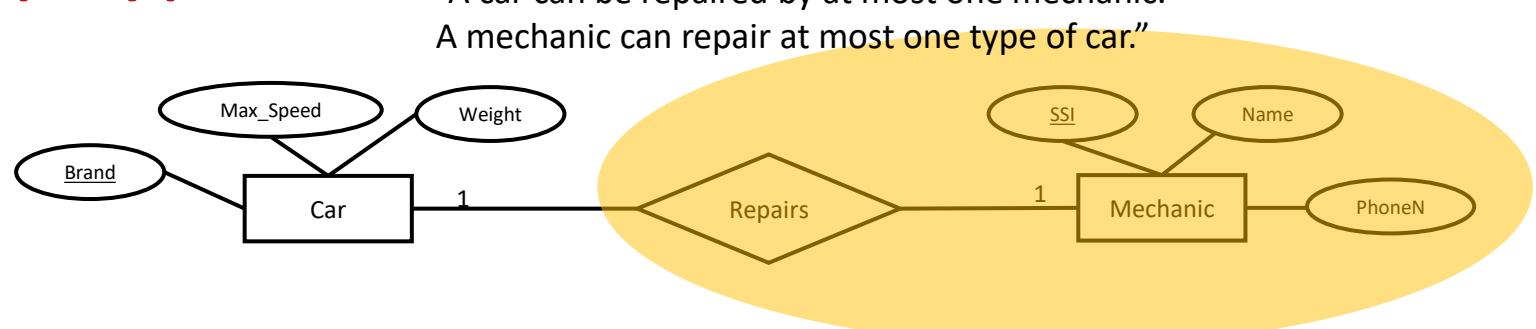
Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT,
PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR, **Brand
is UNIQUE**)

The repairs relation is one-to-one. Therefore, for every SSI, there must exist one Brand.
Moreover, as Brand cannot repeat, we use the **UNIQUE** keyword.

How do we derive Foreign Keys and ICs for different relationship types?

- 1 to 1 relations

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210



SSI	Name	Phone_Number	Brand
87542702	Tom	75315567	Toyota_Corolla
68201937	Uraz	75335521	Hyundai E.GLS
23139827	Nick	75315544	BMW 3.21

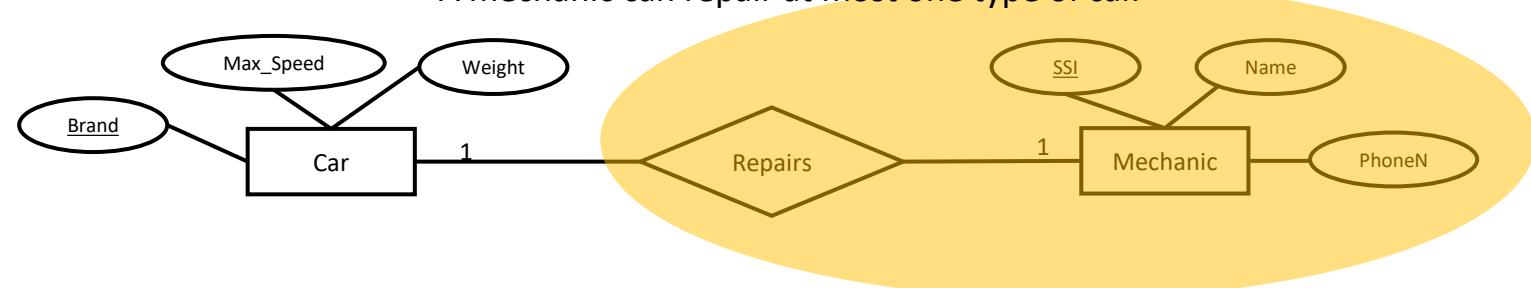
Car(Brand:TEXT,Weight:INT,Length:DOUBLE,
Max_Speed:INT, PRIMARY KEY:BRAND)

Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT,
PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR, Brand
is UNIQUE)

Assume I delete the tuple “BMW 3.21, 1400, 3.21, 200” from the CAR table. What value
should DBMS set for the Mechanic that can repair BMW 3.21?

How do we derive Foreign Keys and ICs for different relationship types?

- 1 to 1 relations



Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

SSI	Name	Phone_Number	Brand
87542702	Tom	75315567	Toyota_Corolla
68201937	Uraz	75335521	Hyundai E.GLS
23139827	Nick	75315544	BMW 3.21

Car(Brand:TEXT,Weight:INT,Length:DOUBLE,
Max_Speed:INT, PRIMARY KEY:BRAND)

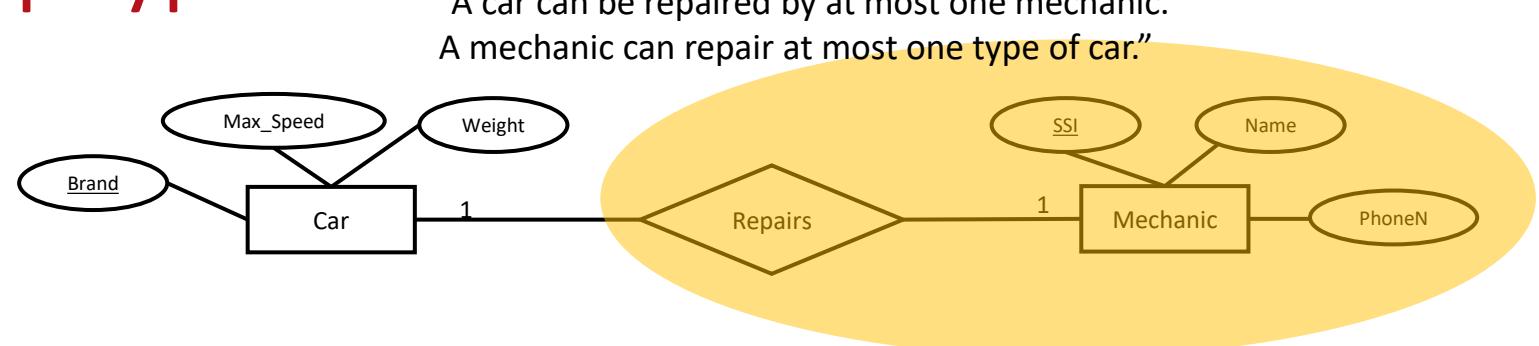
Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT,
PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR, Brand
is UNIQUE)

Assume I delete the tuple "BMW 3.21, 1400, 3.21, 200" from the CAR table. What value
should DBMS set for the Mechanic that can repair BMW 3.21?

Since the Repairs Relation **partially participates** in both ends, I can select **SET NULL** or **SET
DEFAULT**.

How do we derive Foreign Keys and ICs for different relationship types?

- 1 to 1 relations



Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

SSI	Name	Phone_Number	Brand
87542702	Tom	75315567	Toyota_Corolla
68201937	Uraz	75335521	Hyundai E.GLS
23139827	Nick	75315544	BMW 3.21

Car(Brand:TEXT,Weight:INT,Length:DOUBLE,
Max_Speed:INT, PRIMARY KEY:BRAND)

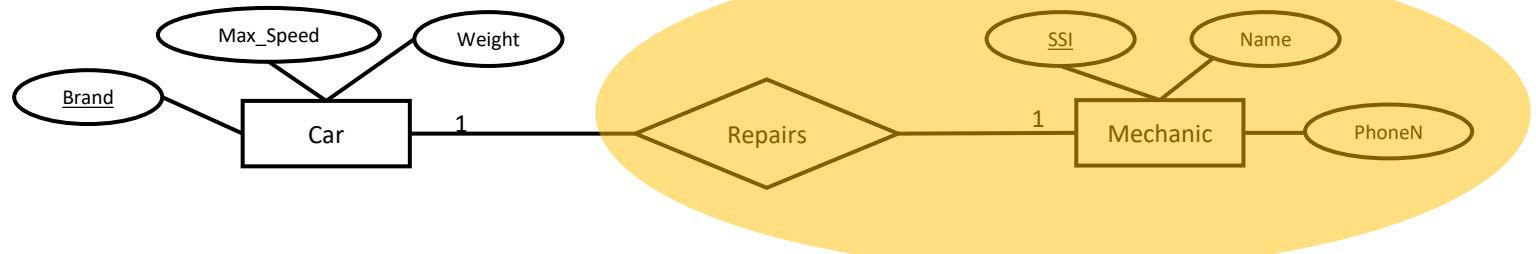
Mec_Rep(SSI:TEXT,Name:TEXT,Phone:TEXT,Brand:TEXT,
PRIMARY KEY:SSI, Foreign Key: Brand REFERENCING:CAR, Brand
is UNIQUE, **on Delete SET NULL/DEFAULT**)

Assume I delete the tuple “BMW 3.21, 1400, 3.21, 200” from the CAR table. What value
should DBMS set for the Mechanic that can repair BMW 3.21?

Since the Repairs Relation **partially participates** in both ends, I can select **SET NULL** or **SET
DEFAULT**.

Referential Integrity

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



CAR

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

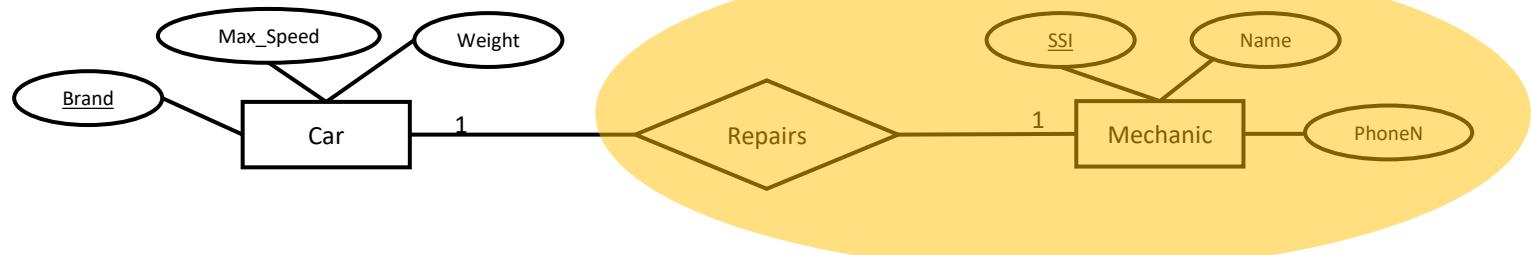
MEC_REPAIR

Brand	Price	SSI	Name	Phone_Number
BMW 3.21	10	87542702	Tom	75315567
Toyota_Corolla	23	68201937	Uraz	75335521
Hyundai E.GLS	12	23139827	Nick	75315544

- If a tuple (say 2nd tuple) is to be deleted from referenced table (CAR)
 -
 -
 -
 -

Referential Integrity

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



CAR

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

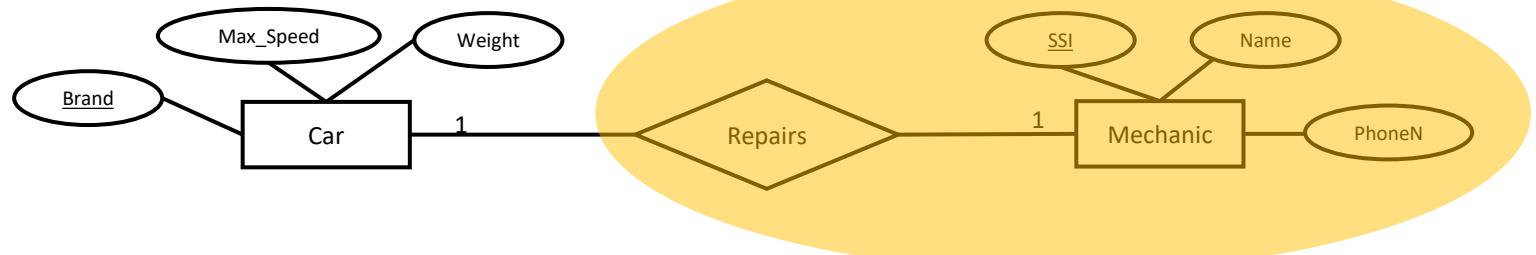
MEC_REPAIR

Brand	Price	SSI	Name	Phone_Number
BMW 3.21	10	87542702	Tom	75315567
Toyota_Corolla	23	68201937	Uraz	75335521
Hyundai E.GLS	12	23139827	Nick	75315544

- If a tuple (say 2nd tuple) is to be deleted from referenced table (CAR)
- Get the primary key value of the tuple (Toyota_Corolla).
-
-

Referential Integrity

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



CAR

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

MEC_REPAIR

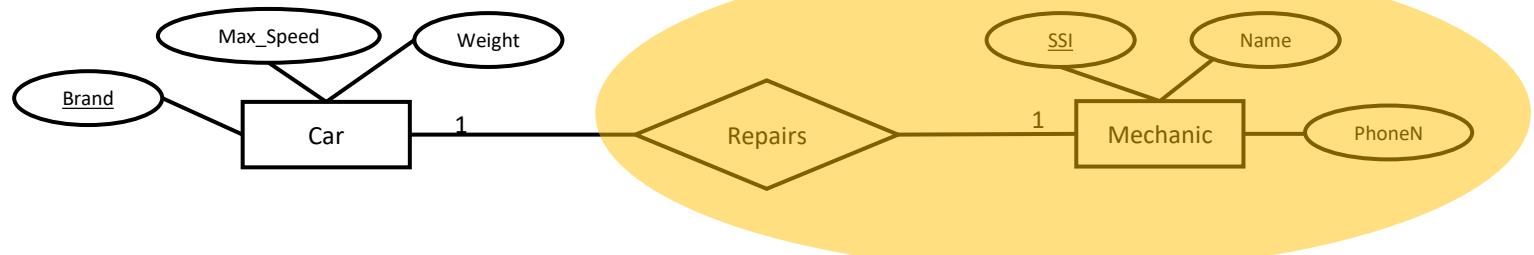
Brand	Price	SSI	Name	Phone_Number
BMW 3.21	10	87542702	Tom	75315567
Toyota_Corolla	23	68201937	Uraz	75335521
Hyundai E.GLS	12	23139827	Nick	75315544

- If a tuple (say 2nd tuple) is to be deleted from referenced table (CAR)
- Get the primary key value of the tuple (Toyota_Corolla).
- Find all the tuples with values (Toyota_Corolla) in the referencing table (MEC_REPAIR)

•

Referential Integrity

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



CAR

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

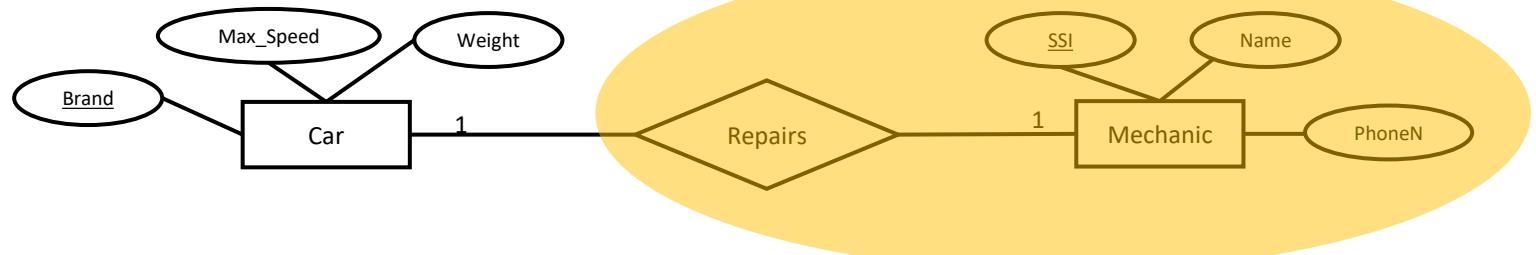
MEC_REPAIR

Brand	Price	SSI	Name	Phone_Number
BMW 3.21	10	87542702	Tom	75315567
DEFAULT	23	68201937	Uraz	75335521
Hyundai E.GLS	12	23139827	Nick	75315544

- If a tuple (say 2nd tuple) is to be deleted from referenced table (CAR)
- Get the primary key value of the tuple (Toyota_Corolla).
- Find all the tuples with values (Toyota_Corolla) in the referencing table (MEC_REPAIR)
 - If SET DEFAULT -> Select all such tuples in the referencing table (MEC_REPAIR) and set the foreign key value to a default value (you have to specify this) of these tuples in the referencing table (MEC_REPAIR).

Referential Integrity

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



CAR

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Hyundai E.GLS	1400	3.16	210

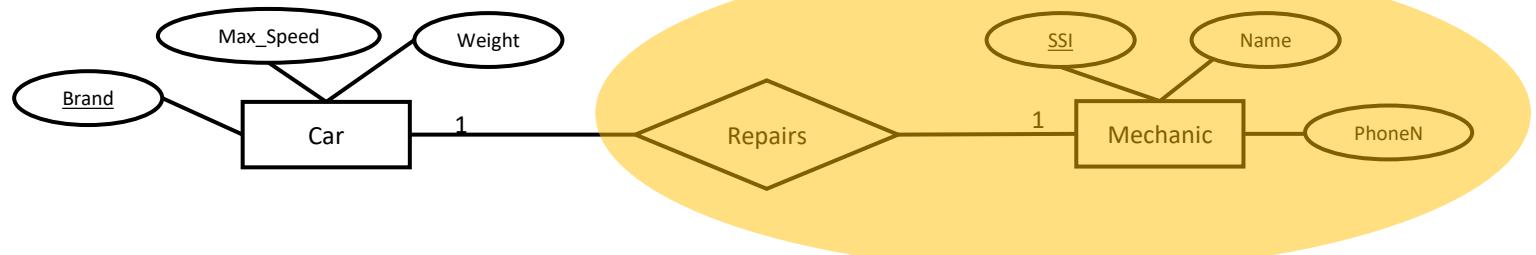
MEC_REPAIR

Brand	Price	SSI	Name	Phone_Number
BMW 3.21	10	87542702	Tom	75315567
DEFAULT	23	68201937	Uraz	75335521
Hyundai E.GLS	12	23139827	Nick	75315544

- If a tuple (say 2nd tuple) is to be deleted from referenced table (CAR)
- Get the primary key value of the tuple (Toyota_Corolla).
- Find all the tuples with values (Toyota_Corolla) in the referencing table (MEC_REPAIR)
 - If SET DEFAULT -> Select all such tuples in the referencing table (MEC_REPAIR) and set the foreign key value to a default value (you have to specify this) of these tuples in the referencing table (MEC_REPAIR). And delete tuples in CAR

Referential Integrity

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



CAR

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

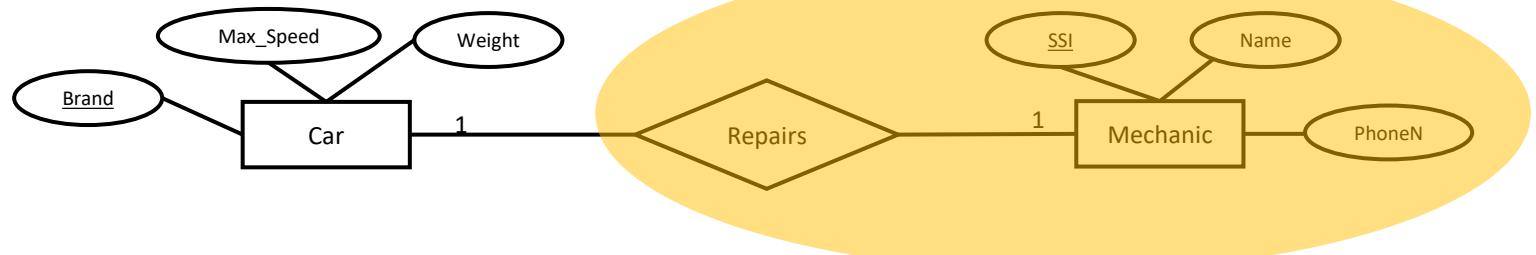
MEC_REPAIR

Brand	Price	SSI	Name	Phone_Number
BMW 3.21	10	87542702	Tom	75315567
Toyota_Corolla	23	68201937	Uraz	75335521
Hyundai E.GLS	12	23139827	Nick	75315544

- If a tuple (say 2nd tuple) is to be deleted from referenced table (CAR)
- Get the primary key value of the tuple (Toyota_Corolla).
- Find all the tuples with values (Toyota_Corolla) in the referencing table (MEC_REPAIR)
 - If SET NULL -> Select all these tuples in the referencing table (MEC_REPAIR)

Referential Integrity

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



CAR

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Toyota_Corolla	1300	3.18	200
Hyundai E.GLS	1400	3.16	210

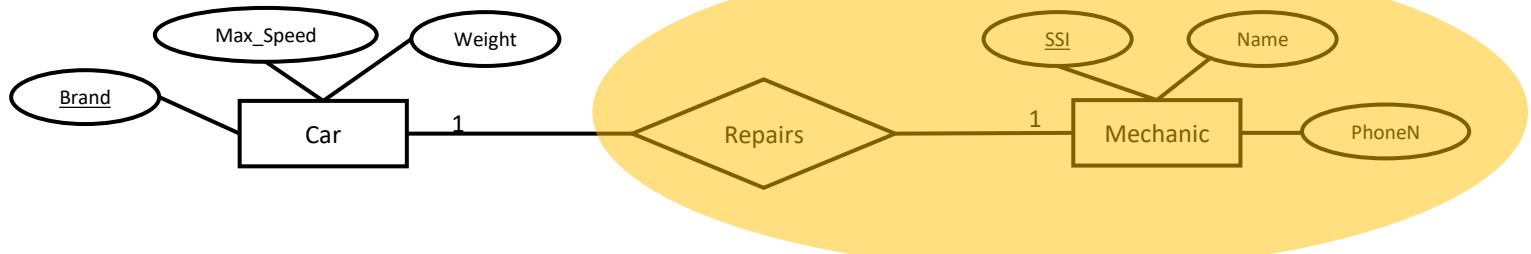
MEC_REPAIR

Brand	Price	SSI	Name	Phone_Number
BMW 3.21	10	87542702	Tom	75315567
NULL	23	68201937	Uraz	75335521
Hyundai E.GLS	12	23139827	Nick	75315544

- If a tuple (say 2nd tuple) is to be deleted from referenced table (CAR)
- Get the primary key value of the tuple (Toyota_Corolla).
- Find all the tuples with values (Toyota_Corolla) in the referencing table (MEC_REPAIR)
 - If SET NULL -> Select all these tuples in the referencing table (MEC_REPAIR) and set the foreign key value to a NULL value of these tuples in the referencing table (MEC_REPAIR).

Referential Integrity

"A car can be repaired by at most one mechanic.
A mechanic can repair at most one type of car."



CAR

Brand	Weight	Length	Max_Speed
BMW 3.21	1400	3.21	200
Hyundai E.GLS	1400	3.16	210

MEC_REPAIR

Brand	Price	SSI	Name	Phone_Number
BMW 3.21	10	87542702	Tom	75315567
NULL	23	68201937	Uraz	75335521
Hyundai E.GLS	12	23139827	Nick	75315544

- If a tuple (say 2nd tuple) is to be deleted from referenced table (CAR)
- Get the primary key value of the tuple (Toyota_Corolla).
- Find all the tuples with values (Toyota_Corolla) in the referencing table (MEC_REPAIR)
 - If SET NULL -> Select all these tuples in the referencing table (MEC_REPAIR) and set the foreign key value to a NULL value of these tuples in the referencing table (MEC_REPAIR).
 - And delete the tuples in the referenced table (CAR).