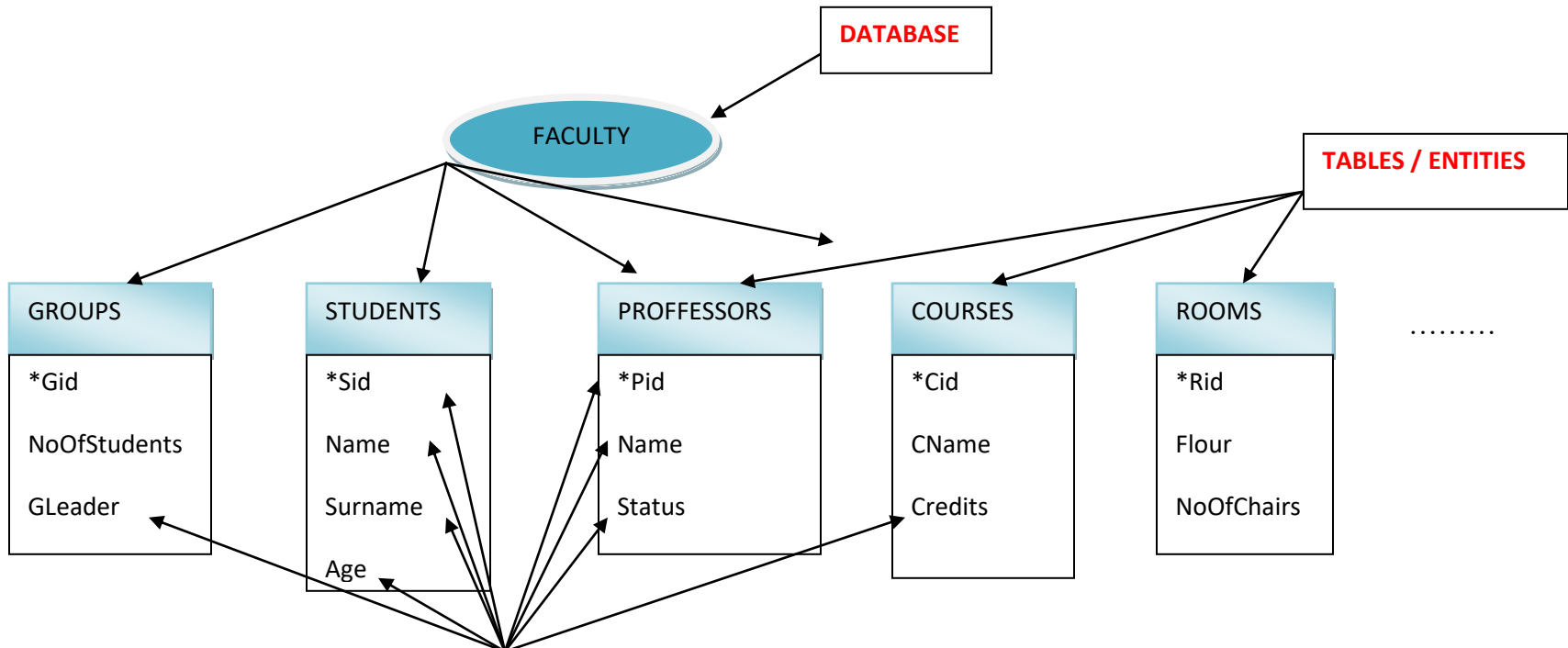


DATABASES

A database example



Each table has one or more

ATTRIBUTES / FIELDS / COLUMNS (in the table ...)

(Gid, NoOfSTudents, GLeader, Sid, Name, ...)

PRIMARY KEY = the field from a table that has UNIQUE values for each RECORD and it is NOT NULL. (Gid, Sid, ...)

GROUPS

Gid	NoOfStudents	GLeader
921	28	Paul
923	27	Alin
223	29	Cristina
...		

RECORDS / LINES / ROWS (in the table ...)

FOREIGN KEY = the PRIMARY KEY from the table used in the RELATIONSHIP with which establish the properties:

- have the SAME TYPE
- have the SAME VALUES (not necessarily unique, a value can appear for zero, one or more times)
- NOT NULL

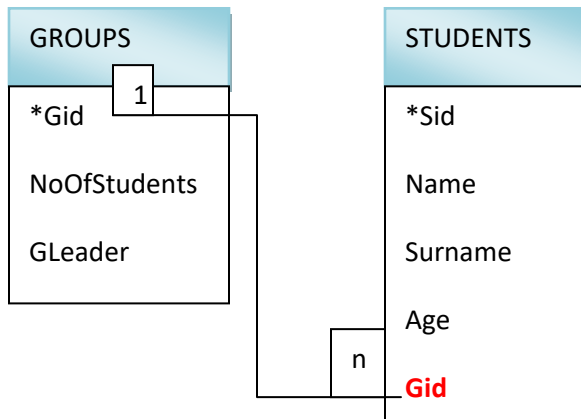
All – database, table, attribute, primary key, foreign key – SHOULD have only ONE WORD (even if are like NumberOfStudents), to access them easy.

RELATIONSHIP (2 tables involved)

- 1 – 1 : Students - DrivingLicences, Students - IdentityCards, Groups - LeaderGroups, Groups - Tutors, ...
- 1 – n : Groups – Students, ...
- m – n : Students – Courses, Professors – Students, ...

Relationship 1-n (one to many)

- The ORDER of the tables is IMPORTANT
- 1 – n - Groups – Students - **In a Group** there are **one or more Students** and a **Student** is part of **only one Group**
- n – 1 – Students – Groups – **A Student** is from **only one Group** and **in a Group** there are **one or more Students**
- The PRIMARY KEY is taken from the table that is in the part 1 of the relationship and becomes FOREIGN KEY in the table that is in the part n of the relationship
- The first table that will be created is the table with the Primary key (the one from the part 1 of the relationship), so that one can extract the values from this one and use them as values for the foreign key



- Gid – can have any name that one wants in the table Students
- Gid from Groups and Gid from Students must have the same type (INT, VARCHAR(30), DATE, ...)
- Gid from Students must have the same values as Gid from Groups, otherwise, error
- NULL cannot be inserted in Gid from Groups and also in Students

Gid	NoOfStudents	GLeader
921	28	Paul
923	27	Alin
223	29	Cristina
...		
...		

GROUPS (table)

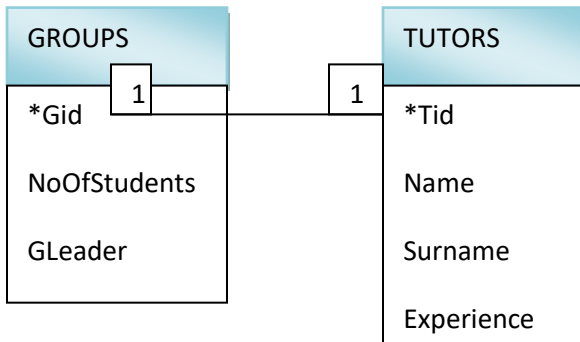
Sid	Name	Surname	Age	Gid
1	A	G	20	921
2	B	T	20	921
3	C	H	21	921
4	D	D	22	223
5	E	H	19	223
6	F	Y	20	225

Students (table)

No possible, because in Groups table the Gid value 225 does not exist. An error message will be received.

Relationship 1 – 1 (one to one)

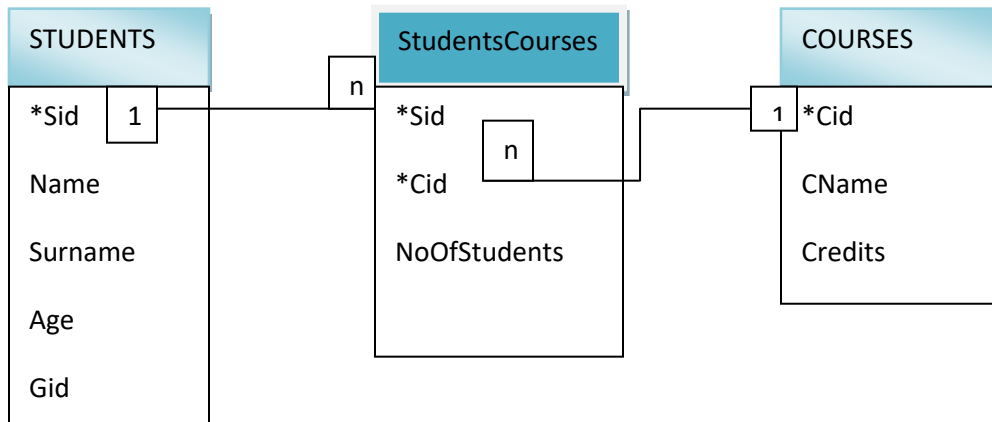
- A particular case of the relationship 1-n established between 2 primary keys – one it is also foreign key
- The first table created will have only primary key
- The second table created will have the primary key set it also as foreign key
- The order is not important (Groups – Tutors or Tutors-Groups)



- if the Groups is the first created, then the primary key is Gid in table Groups and
- Tid from Tutors will be set as Primary key with the same type as Gid from Groups and
- also Tid will be Foreign key in Tutors for Gid from Groups

Relationship m-n (many to many)

- This relationship must contain an INTERMEDIATE Table – this table will contain both primary keys from the 2 tables involved set as FOREIGN KEY (and preferable also as Primary keys)
- Students - Courses – **one or more Students** can take part **in one or more Courses** / **A Student** can participate in **one or more Courses** and **in a Course** can participate **one or more Students**



- The intermediate table can be called as one wants (for example the name of both of the tables), or something specific (for example for tables Clients and Products can be called Orders/Commands)
- The intermediate table can contain other attributes, too
- This implementation with the primary keys set it as foreign key also, WON'T ALLOW DUPLICATE PAIRS

Sid	Name	Surname	Age	Gid
1	A	G	20	921
2	B	T	20	921

Students (table)

Cid	CName	Credits
11	Databases	6
22	MAP	6

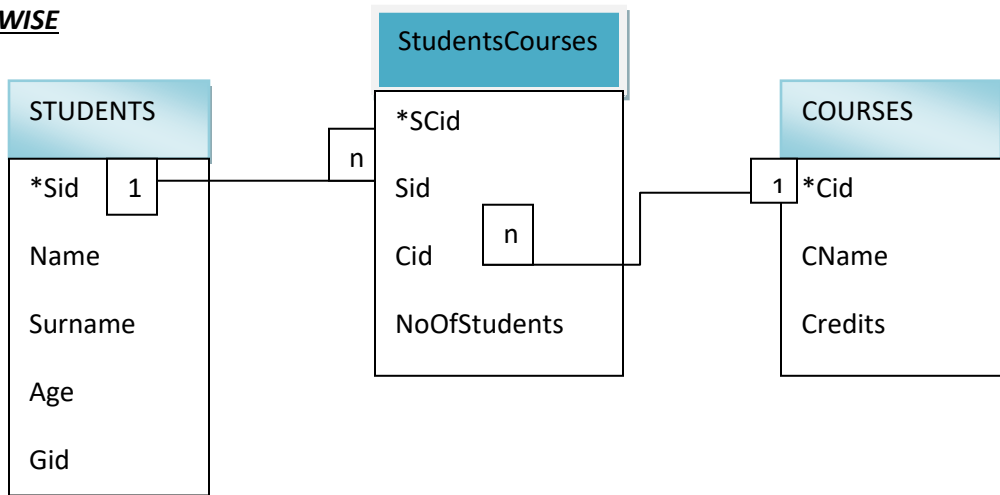
Courses (table)

All possible values will be (because duplicates in pairs will not be allowed):

Sid	Cid	NoOfStudents
1	11	21
1	22	21
2	11	21
2	22	23

StudentsCourses (table)

OTHERWISE



- The intermediate table can contain the same pairs for the foreign keys as many times as one wants

Sid	Name	Surname	Age	Gid
1	A	G	20	921
2	B	T	20	921

Students (table)

Cid	CName	Credits
11	Databases	6
22	MAP	6

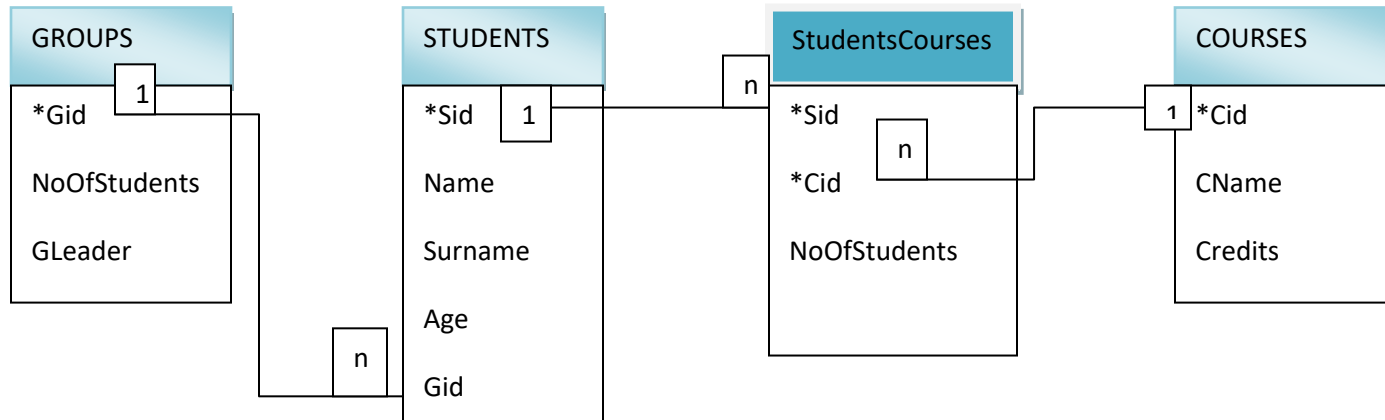
Courses (table)

SCid	Sid	Cid	NoOfStudents
1	1	11	21
2	1	22	21
3	1	11	21
4	1	11	23

StudentsCourses (table)

Relationships – one should not use CICLES.

- Do you need another relationship between Groups – Courses?
- NO, because there exist one – from Courses – to Students – to StudentsCourses – to Courses.
- Choose the 'principal relationships', the one you consider the most important and directly (for example Students are participating to Courses and Students are organized in Groups, and definitely Groups will be also related to Coursed, due to the Students) (other view can be Groups participate to Courses and the Students are organized in Groups, and so Students are related to Courses, due to Groups)



Observations

- In every table MUST EXIST at least ONE PRIMARY KEY
- In a database all tables must be related = each table must have at least one relationship with another table, such that one can extract data from each table using the relationships (please avoid cycles)
- A relationship can be of ONLY ONE TYPE (e.g. 1-n OR m-n) – Courses - Students has to be m-n OR 1-n, but not both of them! (ONLY black line OR red line) – otherwise it is a cicle

