SQL - Relationships

Kodkod 4

Overview

- Understand what relationships are and why they are important.

- Identify and define Primary Keys (PK) and Foreign Keys (FK).

- Create connections between tables using FOREIGN KEY constraints.

- Understand the main relationship types:

- One-to-One, One-to-Many, and Many-to-Many.

- Maintain data integrity and prevent orphan records.

Tables as Entities

- In a relational database, each table represents a real-world entity - a person, object, event, or concept that we want to store information about.
- Each row (record) represents one instance of that entity, and each column (field) represents an attribute of it.
- Examples:

Table	Represents	Example Attributes
students	a student	id, full_name, identity_number, birth_date, class_id
courses	a course	id, course_name
lecturers	a lecturer	id, lecturer_name
classes	a class group	id, class_name, course_id
study_hours	a hours class	id, class_id, 1st_lec_id, 2nd_lec_id

Relationships

## Can We Store Everything in One Flat Table?
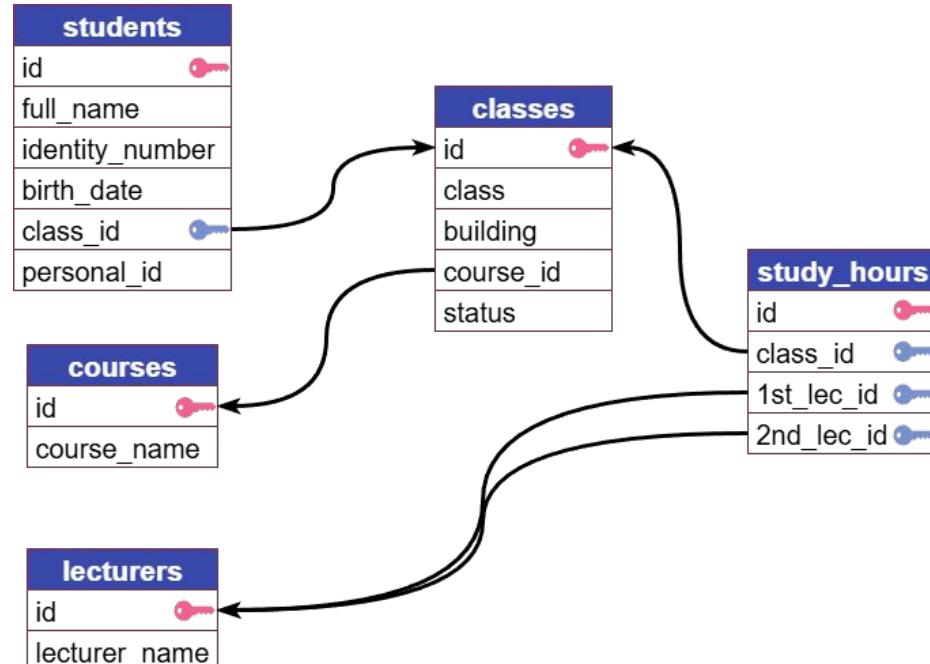
- What happens if we need to update a teacher's name?
- How many places do we need to change it?
- What happens if we delete a class? Will we lose all the students too?
- Do we see any repeated data here?

| student_id	student_name	class_name	course_name	teacher_name	building
1	Alice Green	Class A	Math	John Smith	A
2	David Levi	Class A	Math	John Smith	A
3	Sarah Cohen	Class A	Math	John Smith	A
4	Ben Azulay	Class B	History	Dana Weiss	B
5	Ruth Katz	Class B	History	Dana Weiss	B
6	Tom Amir	Class C	Physics	Ron Levi	C

Relationships

Why We Need Relationships
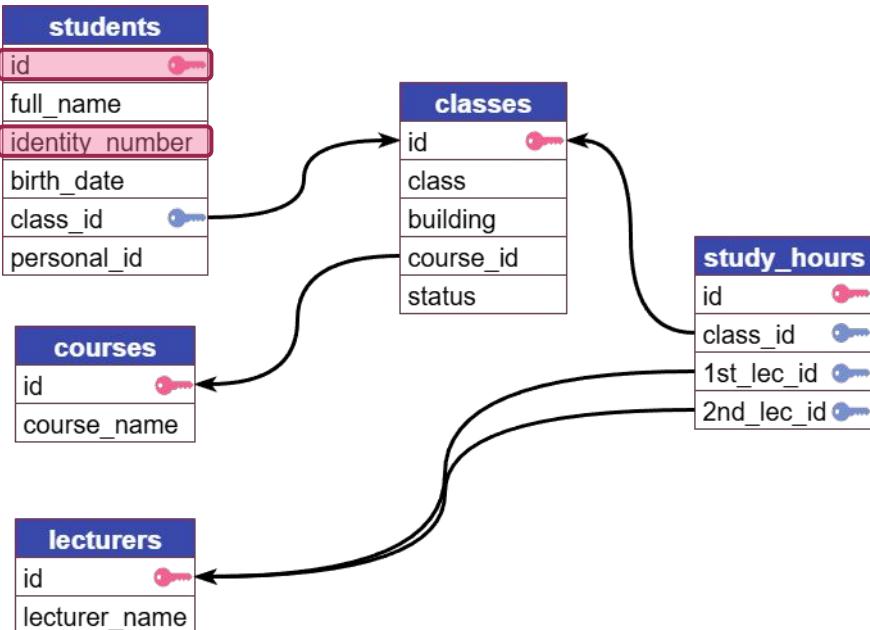
- Until now, we worked with each table individually.
- But in real life, data is **connected** - students, classes, teachers, and courses all depend on each other.
- The pink key 🔑 represents a primary key.
- The blue key 🔑 represents a foreign key.

students
id	🔑
full_name	
identity_number	
birth_date	
class_id	🔑
personal_id	

classes
id	🔑
class	
building	
course_id	
status	

study_hours
id	🔑
class_id	🔑
1st_lec_id	🔑
2nd_lec_id	🔑

courses
id	🔑
course_name	

lecturers
id	🔑
lecturer_name	

Relationships Types
One-to-One (1:1)

one student - one identity number
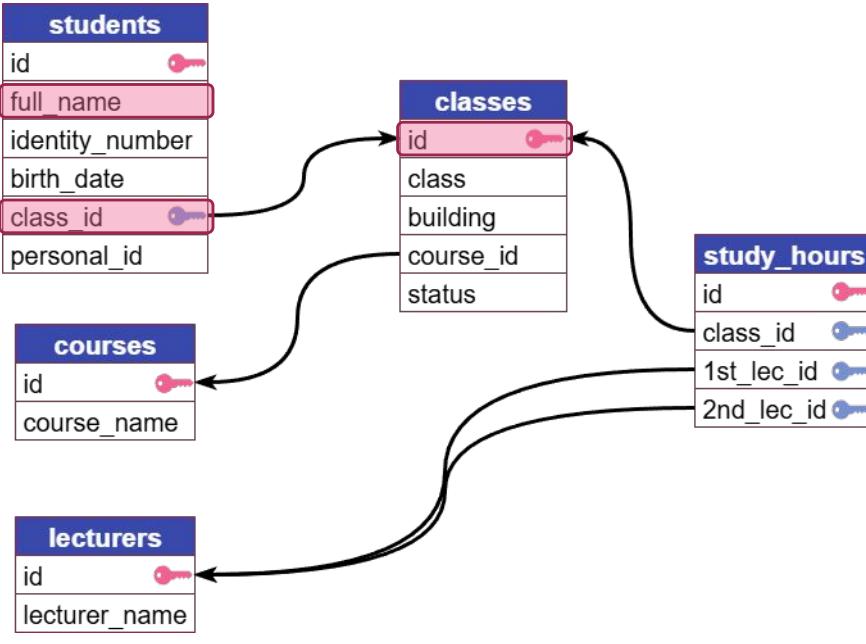
id	full_name	identity_number	birth_date	class_id
1	Jennifer Thomas	999412000000	04/07/2000	2
2	Linda Martinez	999289000000	30/12/1990	2
3	James Thomas	999449000000	28/07/1991	2
4	Barbara Rodriguez	999646000000	21/08/1992	2
5	Michael Clark	999495000000	18/09/1992	4
6	Laura Martinez	999773000000	13/10/1995	4

Relationships Types

One-to-Many (N:1)

one class - many students
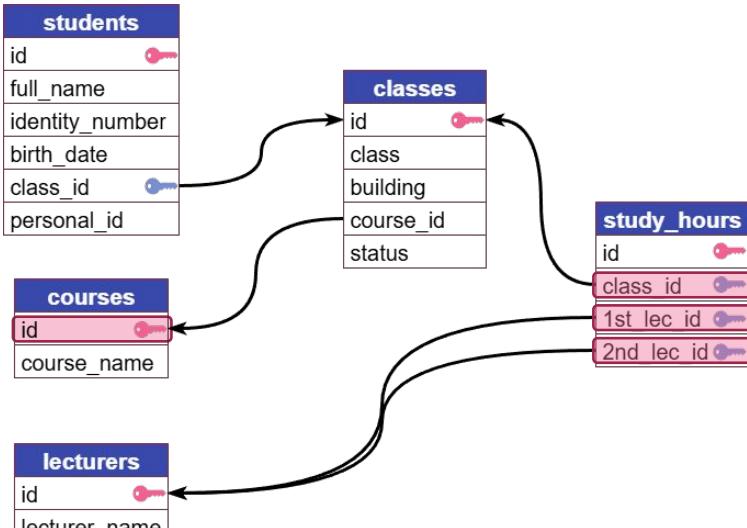
id	class	building	course_id
1	A	North	4
2	B	South	3
3	C	South	2
4	D	North	1

id	full_name	identity_number	birth_date	class_id
1	Jennifer Thomas	999412000000	04/07/2000	2
2	Linda Martinez	999289000000	30/12/1990	2
3	James Thomas	999449000000	28/07/1991	2
4	Barbara Rodriguez	999646000000	21/08/1992	2

Relationships Types
Many-to-Many (M:N)

many classes - many lectures
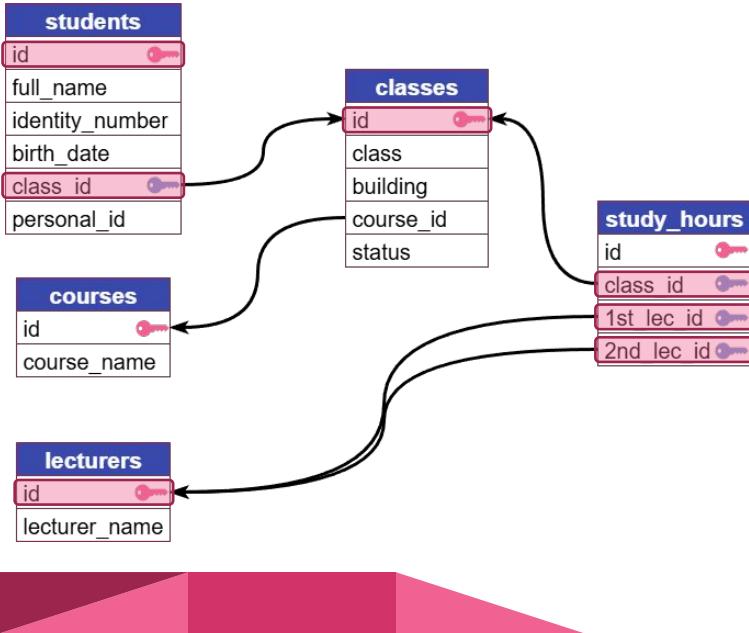
id	lecturer_name
1	Michael Jackson
2	Linda Robinson
3	John Smith
4	Michael Clark
5	James Anderson
6	David Martin
7	Michael Thompson
8	Robert Robinson

students
- id
- full_name
- identity_number
- birth_date
- class_id
- personal_id

classes
- id
- class
- building
- course_id
- status

study_hours
- id
- class_id
- 1st_lec_id
- 2nd_lec_id

courses
- id
- course_name

lecturers
- id
- lecturer_name

id	class_id	1st_lec_id	2nd_lec_id
1	1	1	1
2	2	8	2
3	3	7	4
4	4	2	7

Relationships Types
Bridge \ Junction Table

Many lectures are linked to many students

through the **classes** and **study_hours** tables

id	class_id	1st_lec_id	2nd_lec_id
1	1	1	1
2	2	8	2
3	3	7	4
4	4	2	7

id	class	building	course_id
1	A	North	4
2	B	South	3
3	C	South	2
4	D	North	1

students
- id
- full_name
- identity_number
- birth_date
- class_id
- personal_id

classes
- id
- class
- building
- course_id
- status

study_hours
- id
- class_id
- 1st_lec_id
- 2nd_lec_id

courses
- id
- course_name

lecturers
- id
- lecturer_name

id	lecturer_name
1	Michael Jackson
2	Linda Robinson ●
3	John Smith
4	Michael Clark ◆
5	James Anderson
6	David Martin
7	Michael Thompson ▲
8	Robert Robinson ■

id	full_name	class_id
1	Jennifer Thomas ●■	2
2	Linda Martinez ●■	2
6	Laura Martinez ●▲	4
20	Robert Martinez ▲◆	3

Class Exercises

1. Look at the following pairs of tables (from our school database) and identify the type of relationship: (One-to-One, One-to-Many, or Many-to-Many)
   a. students & classes
   b. classes & courses
   c. lecturers & study_hours
   d. students & lecturers
2. Which table (or tables) act as bridge tables that connect two other entities and which columns ar used as a primary key and and as a foreign key?
   a. classes & lecturers
   b. students & lecturers

Summary

- Understand that each table represents an entity.

- Identify why data should be split into multiple tables to avoid redundancy and anomalies.

- Define and recognize relationships between tables.

- Understand the three relationship types - (1:1), (N:1), (M:N)

- Use a Bridge (Junction) Table to connect (M:N) relationships.

- See how all entities connect in the School Database diagram.