



# SQL Joins

A modern conference room with large windows and a long table. The room is empty, with several chairs arranged around the table. The view outside the windows shows a cityscape. The text "Introduction to Joins" is overlaid in the center.

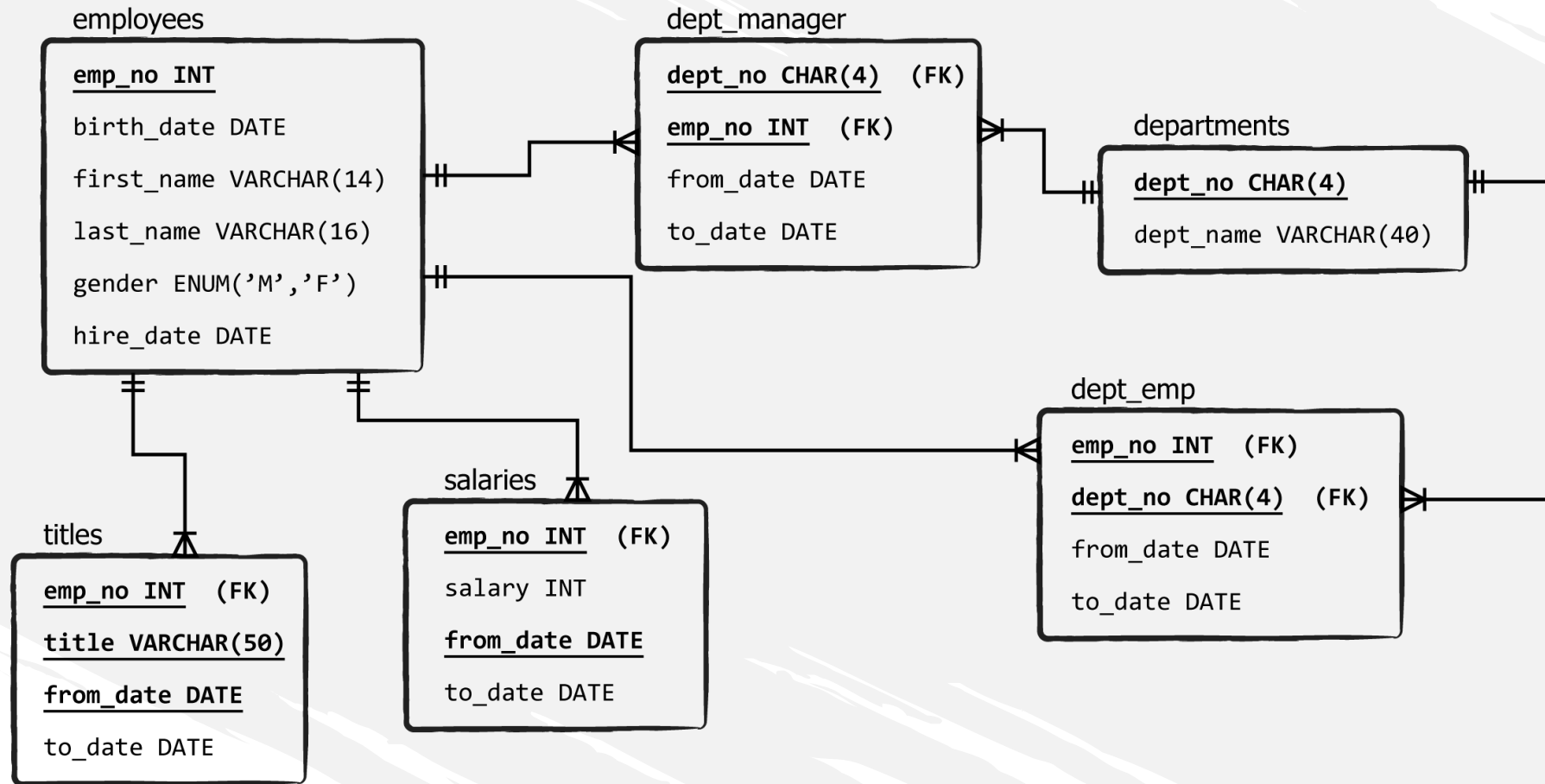
# Introduction to Joins

# Introduction to Joins

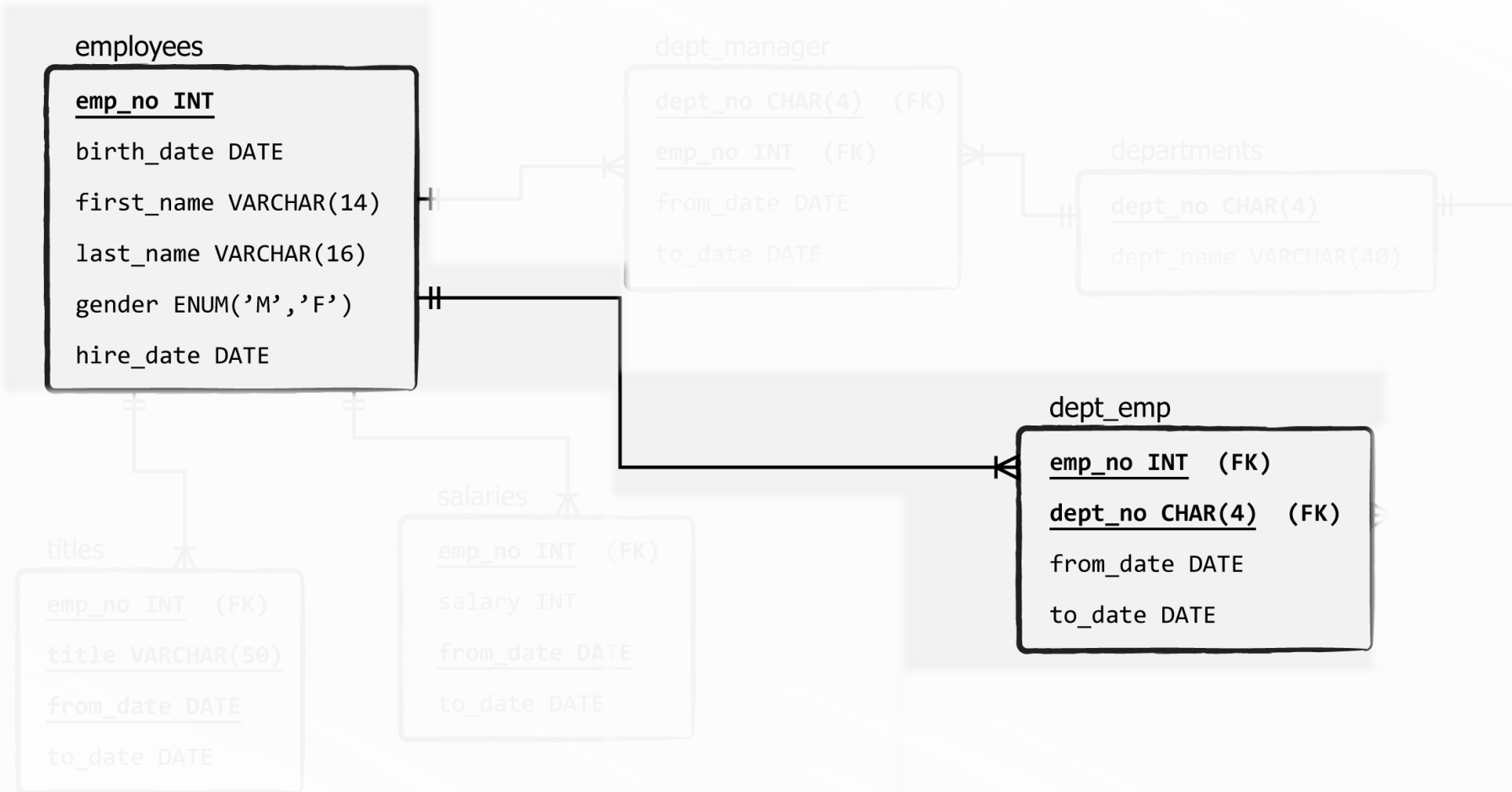
- joins

the SQL tool that allow us to construct a relationship between objects

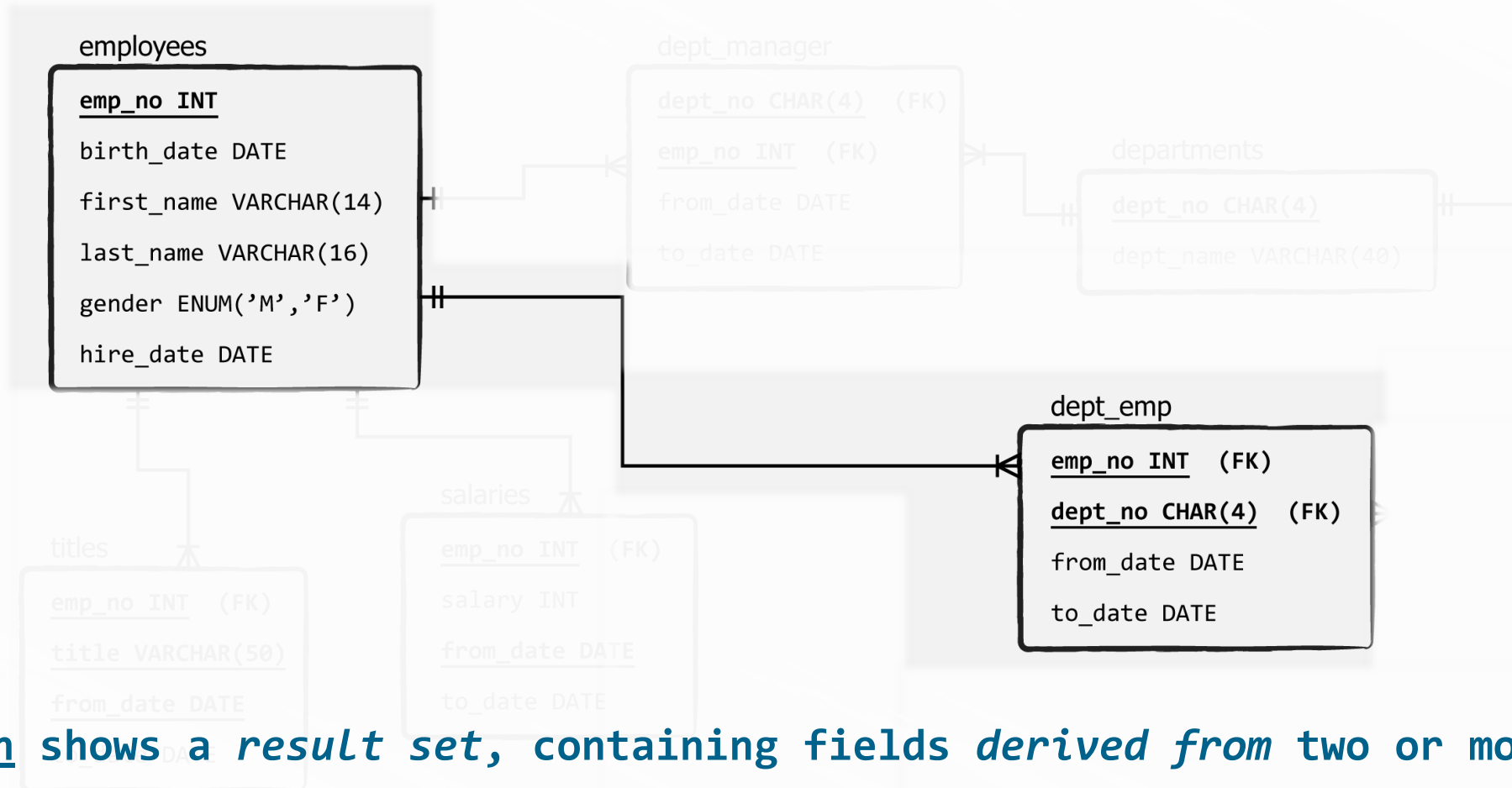
# Introduction to Joins



# Introduction to Joins



# Introduction to Joins



a join shows a *result set*, containing fields *derived from* two or more tables

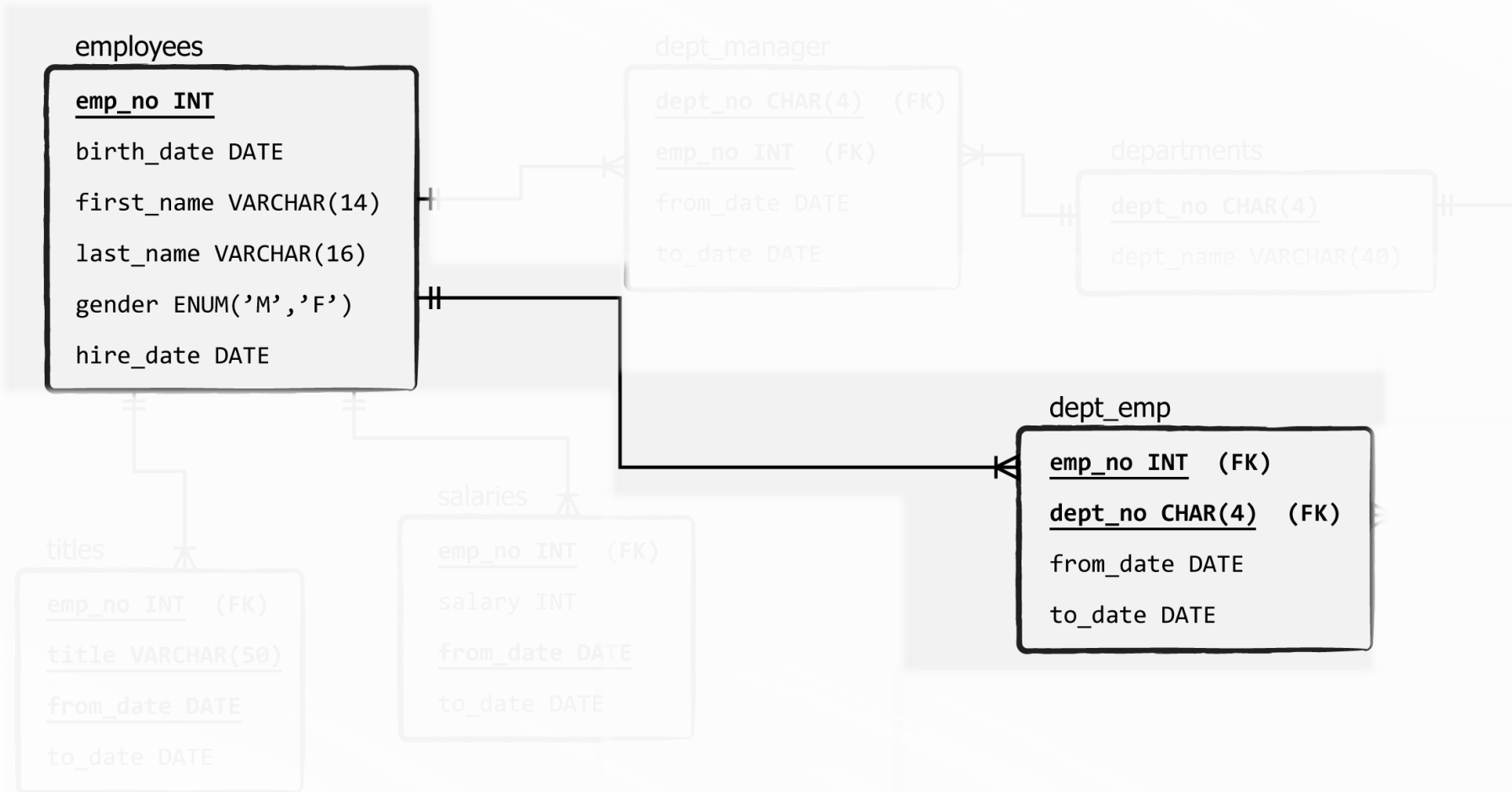
# Introduction to Joins

## ● joins

- we must find a *related column* from the two tables that contains the same *type* of data



# Introduction to Joins



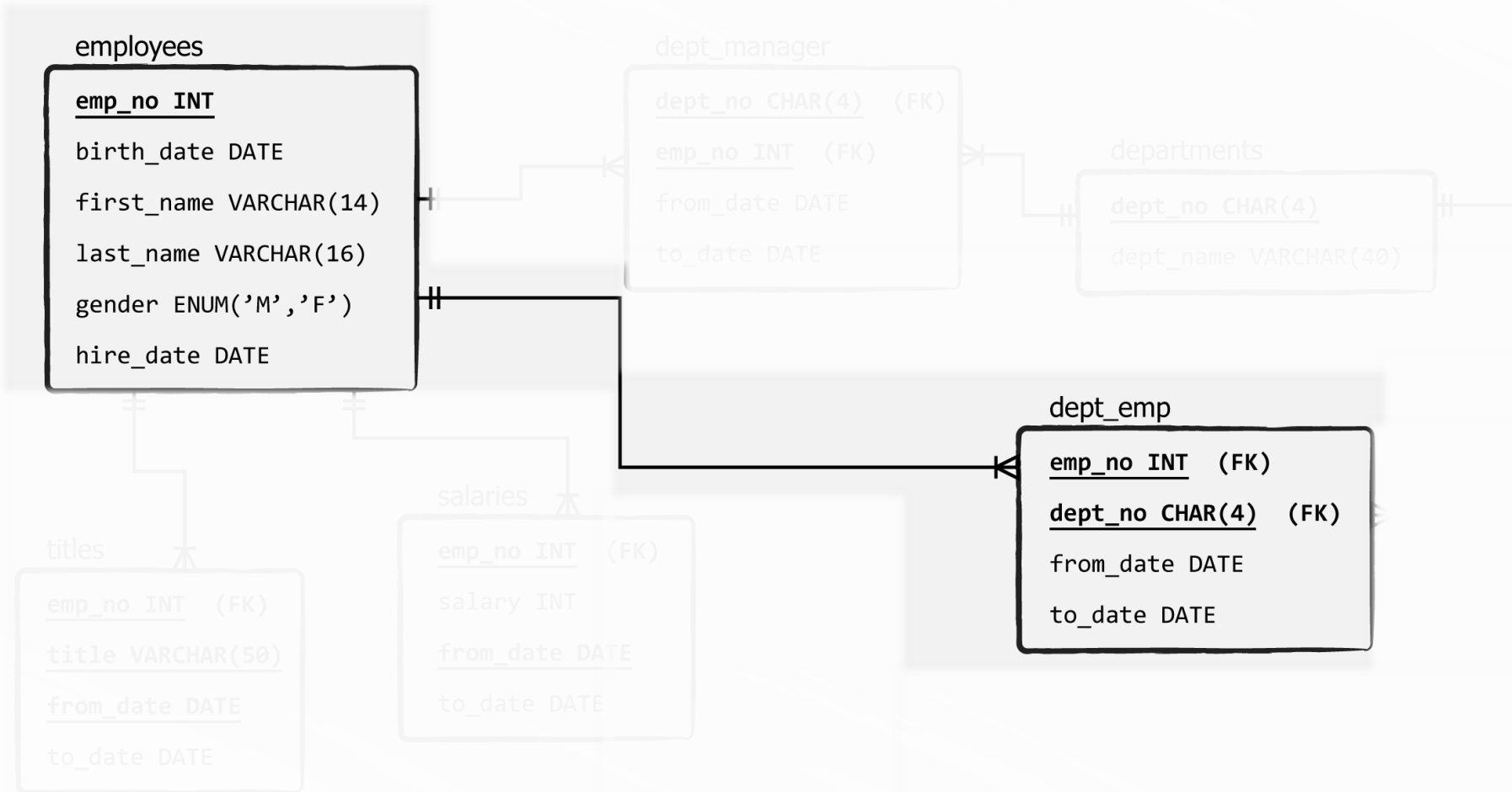


# Introduction to Joins

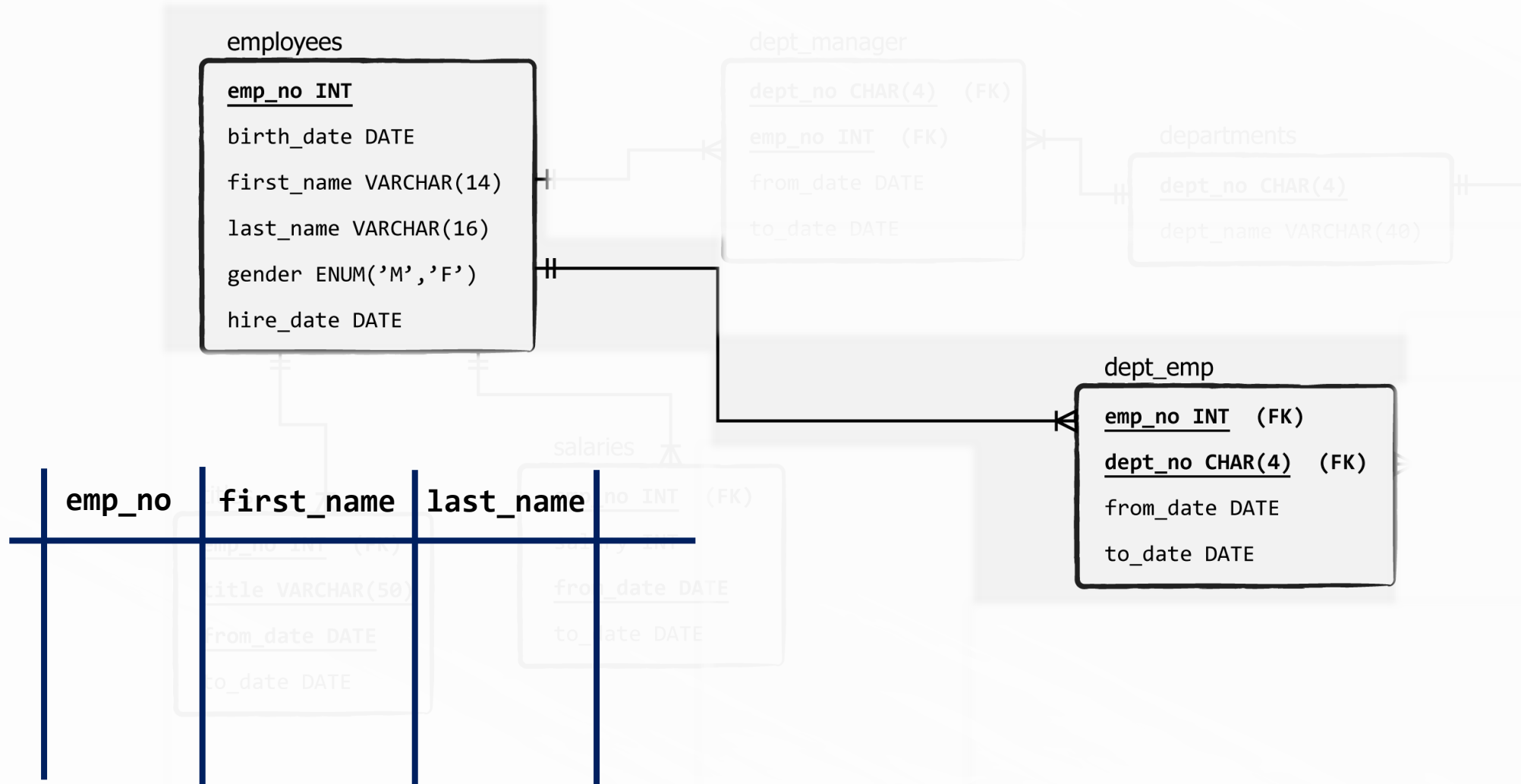
## ● joins

- we must find a *related column* from the two tables that contains the same *type* of data
- we will be free to add columns from these two tables to our output

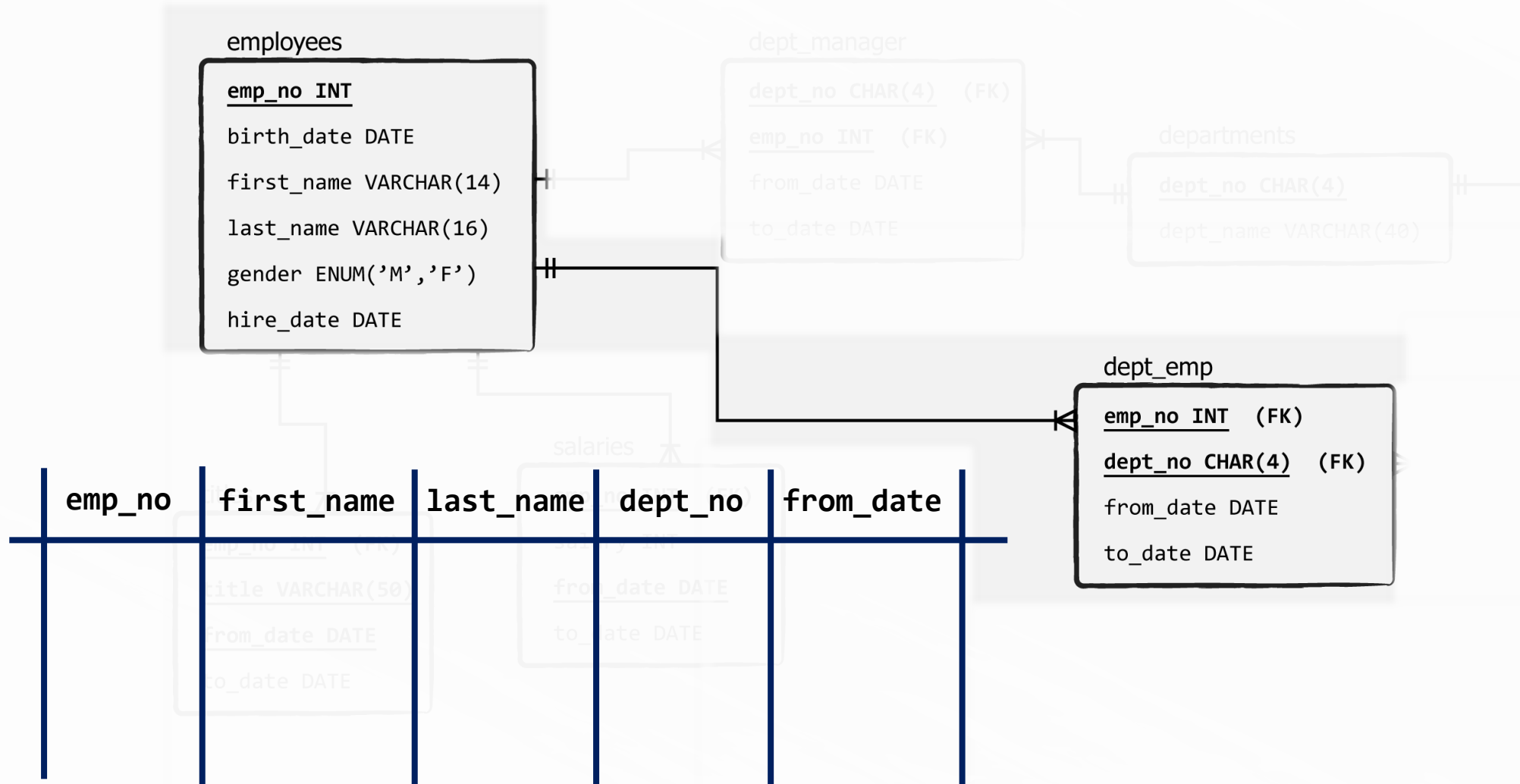
# Introduction to Joins



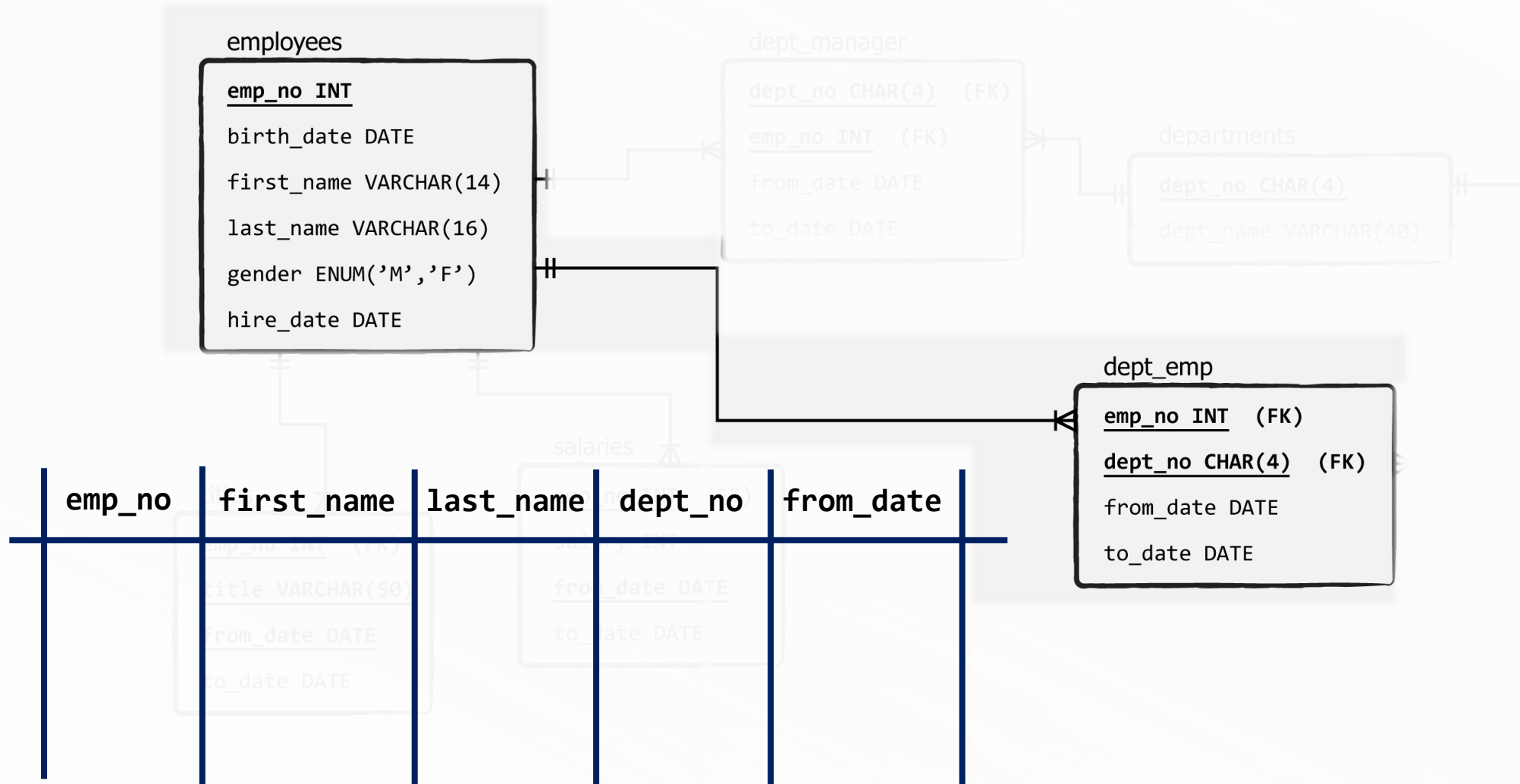
# Introduction to Joins



# Introduction to Joins



# Introduction to Joins

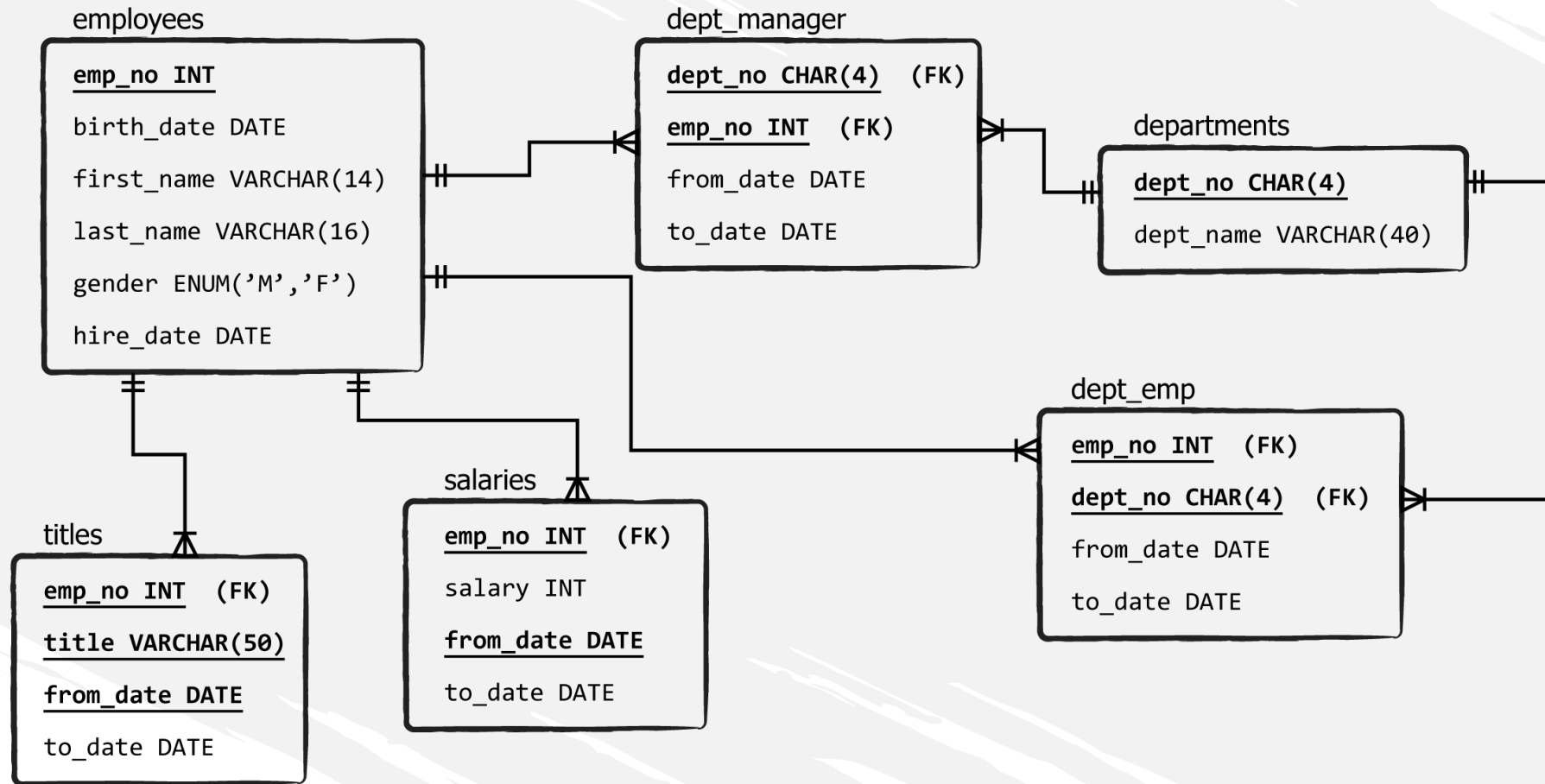


# Introduction to Joins

## ● joins

- the columns you use to relate tables must represent the same object, *such as id*
- the tables you are considering need not be logically adjacent

# Introduction to Joins





# Introduction to Joins

- We will use two duplicate tables:

- 'departments\_dup'
- 'dept\_manager\_dup'

Next:

INNER JOIN