ERHVERVSAKADEMI KØBENHAVN

Introduktion til databaser og SQL



Hvorfor Docker?

Lokal installation:

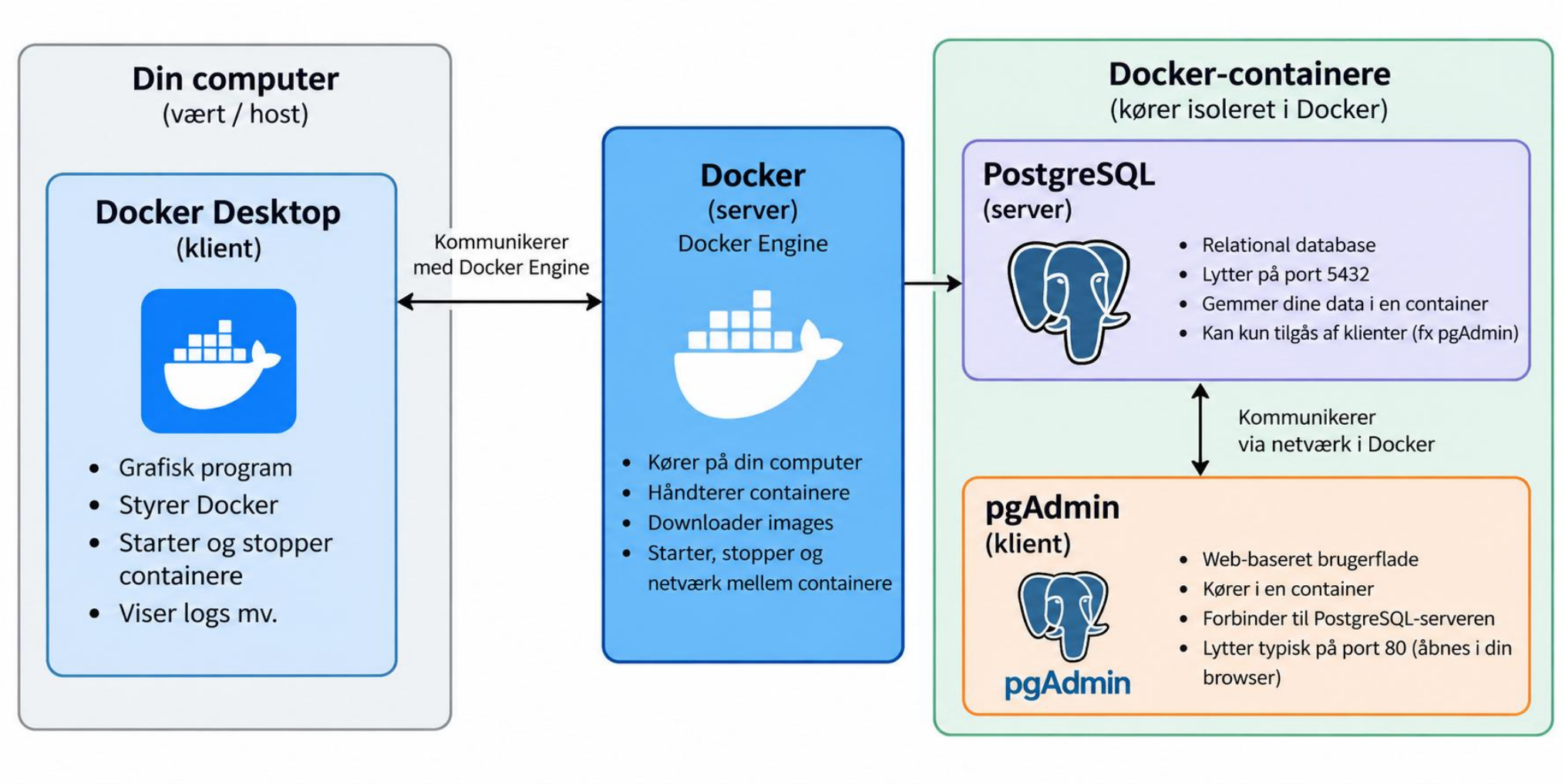
- Download PostgreSQL installer
- Klik gennem setup wizard
- Konfigurer porte og brugere
- Installer pgAdmin separat
- Håb at alt virker sammen
- Problemer med andre versioner
- Bonus problem: ENORMT besværligt at afinstallere!

Installeret på Docker:

- Samme setup hos alle
- Docker giver os et isoleret miljø og sørger for forbindelse mellem programmer
- Vi kan deploye det samlet til fx Digital Ocean
- Det er let (… hvis man ikke har problemer med virtualisering…)

```
docker compose up -d
```

Docker, Docker Desktop, PostgreSQL og pgAdmin



Hvad betyder

Query

CRUD

Primærnøgle

Tabel

Database

DDL

SQL

Fremmednøgle

DML

Kolonne

Række

DDL, DML og CRUD

DDL
- CREATE
- ALTER
- DROP

DML
- SELECT
- INSERT
- UPDATE
- DELETE

CRUD
- Create
- Read
- Update
- Delete

Guidet tur i pgAdmin

- Servers
- Databases
- Schemas
- Tables
- Keys
- Columns
- Constraints
- Query Tool

Naming conventions

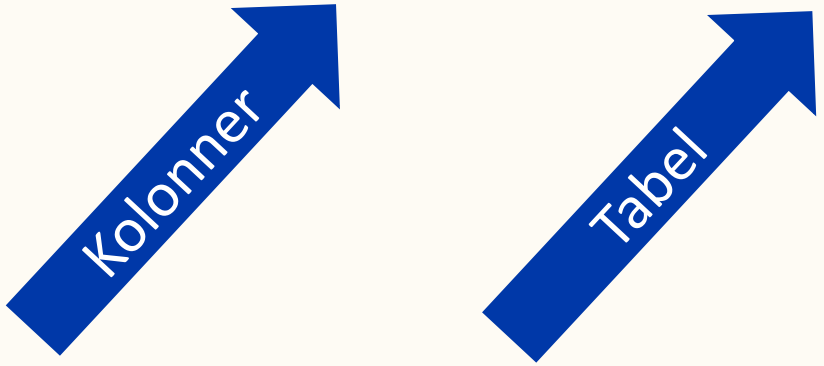
Convention	Eksempel	Typisk brug
camelCase	bookTitle	Java, JavaScript, variabler og metoder
PascalCase	BookTitle	Java-klasser, interfaces, typer
snake_case	book_title	Databaser, SQL, Python m.fl.
kebab-case	book-title	URL'er, filnavne, CSS

Nøgler

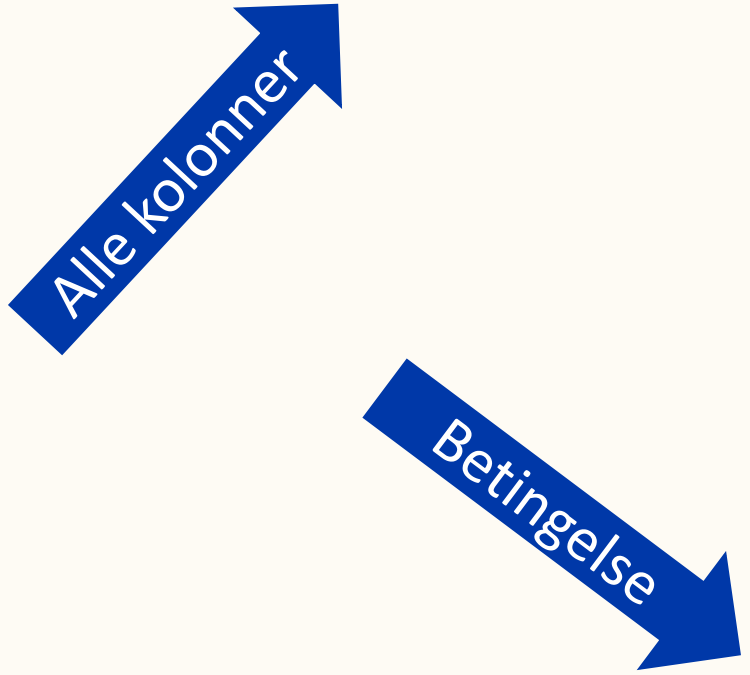
Type	Hvad er det?	Eksempel
Primærnøgle (PK)	Identificerer entydigt en række i en tabel	book_id
Fremmednøgle (FK)	Refererer til en primærnøgle i en anden tabel	author_id
Naturlig nøgle	En nøgle, der allerede findes i domænet	ISBN
Kunstig nøgle	En nøgle, vi selv opretter, fordi vi har brug for en identifikator	book_id = 42
Sammensat nøgle	En nøgle bestående af flere kolonner	book_id + author_id

SQL SELECT

```sql
SELECT user_name, user_id FROM users
```



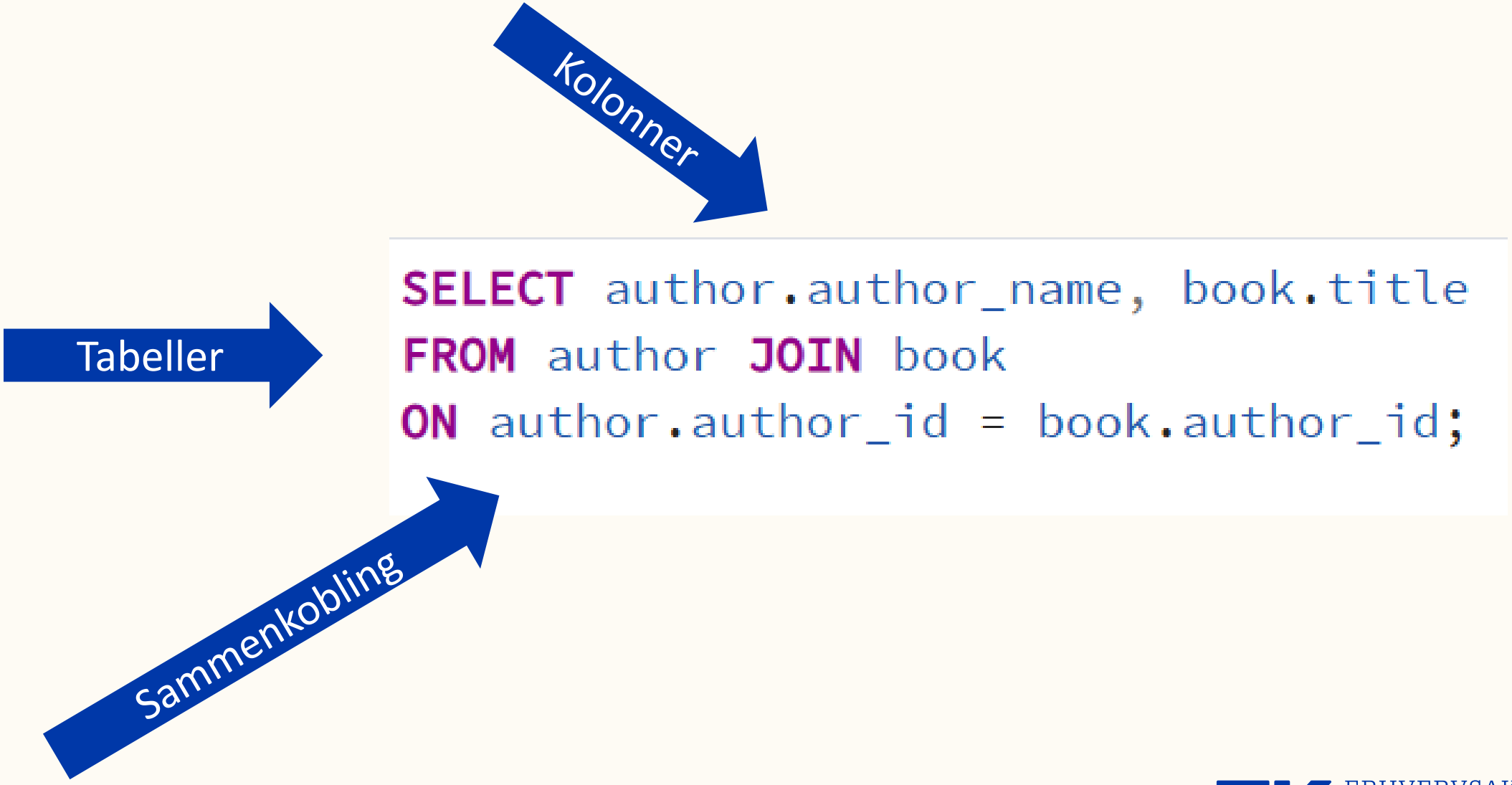

```sql
SELECT * FROM users
```




```sql
SELECT title FROM book WHERE author=11;
```


ERHVERVSAKADEMI KØBENHAVN

Join


Kolonner
Tabeller
SELECT author.author_name, book.title
FROM author JOIN book
ON author.author_id = book.author_id;
Sammenkobling

Join med USING



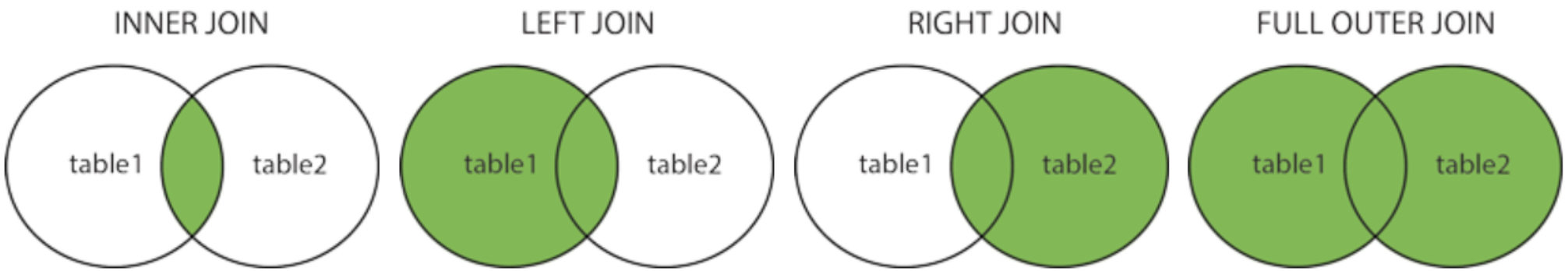

```sql
SELECT author.author_name, book.title
FROM author JOIN book
USING (author_id);
```

Typer af joins

Different Types of SQL JOINs

Here are the different types of the JOINs in SQL:

- `(INNER) JOIN`: Returns records that have matching values in both tables
- `LEFT (OUTER) JOIN`: Returns all records from the left table, and the matched records from the right table
- `RIGHT (OUTER) JOIN`: Returns all records from the right table, and the matched records from the left table
- `FULL (OUTER) JOIN`: Returns all records when there is a match in either left or right table



Views – et udsnit af en database

```sql
CREATE VIEW italian_customers AS SELECT customernumber, customername, phone FROM customers
```



```sql
SELECT customername FROM italian_customers;
```