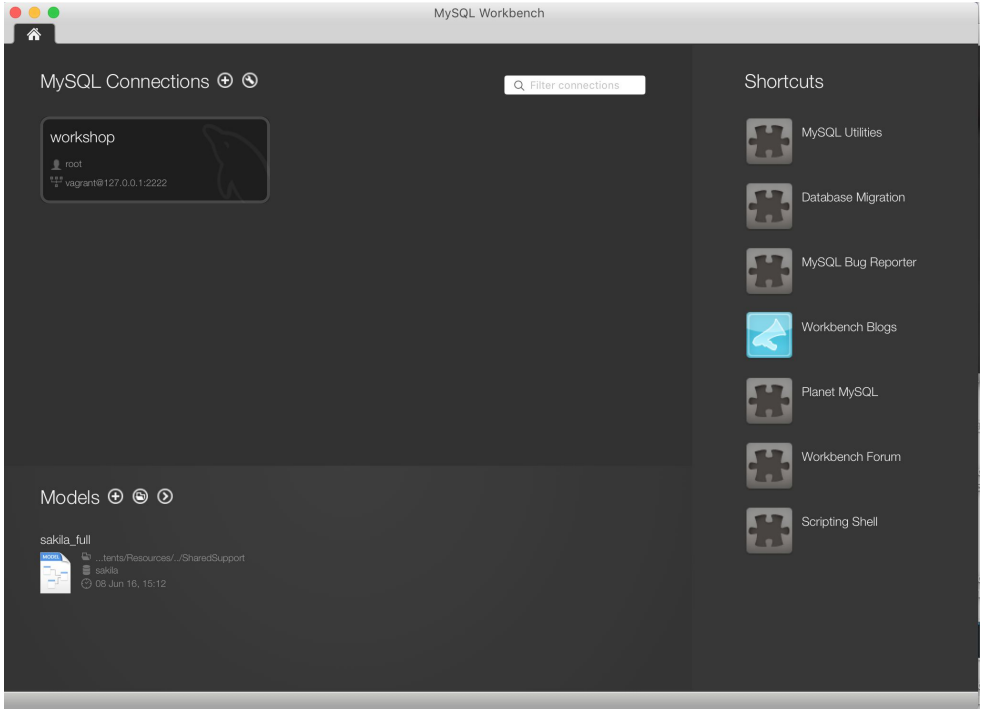
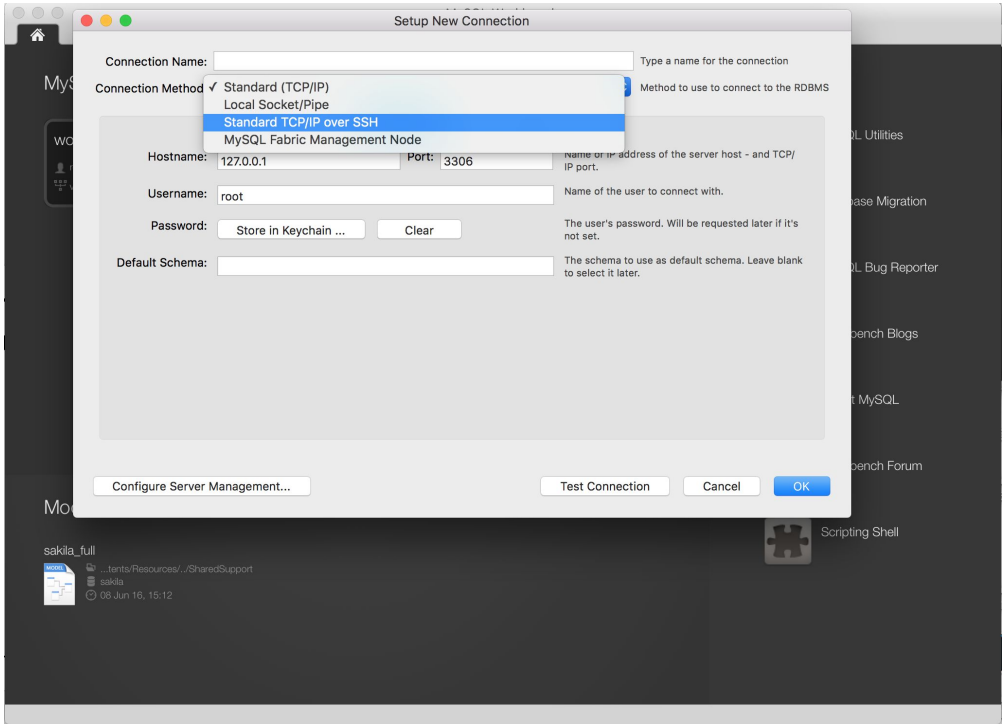
Relationele databanken met MySQL

deel 1

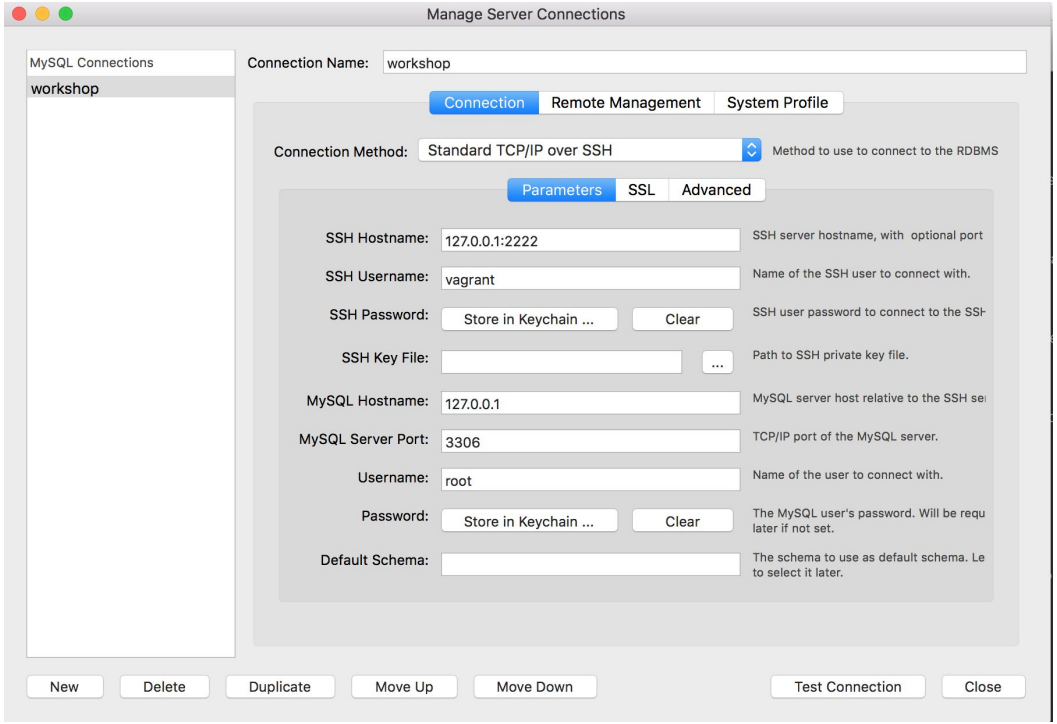
Opzet MySQLWorkbench



Opzet MySQLWorkbench



Opzet MySQLWorkbench



Passwords:
ssh: vagrant
mysql: root

Wat is een relationele database?

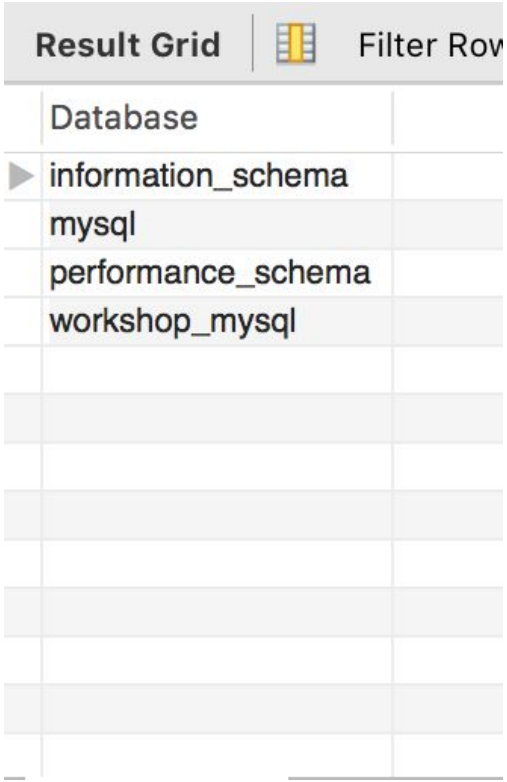
- Digitaal Archief van opgeslagen data
- Collectie van schema’s/tables/views/...
- Data is gestructureerd
- Data is adhv relaties niet redundant
- Data is consistent

Eerste database

```
CREATE DATABASE workshop_mysql;

SHOW DATABASES;

USE workshop_mysql;
```



Eerste Table

```
create table person(
    id int auto_increment unique primary key,
    first_name varchar(256),
    last_name varchar(256),
    email varchar(256) unique
);
```

CRUD Operaties

- Create
- Read
- Update
- Delete

Create

```
insert into person (first_name, last_name, email)
values ('Jerre', 'Nouws', 'jerre@pau.be');
```

Read

```
select * from person
where first_name = 'Jerre';
```

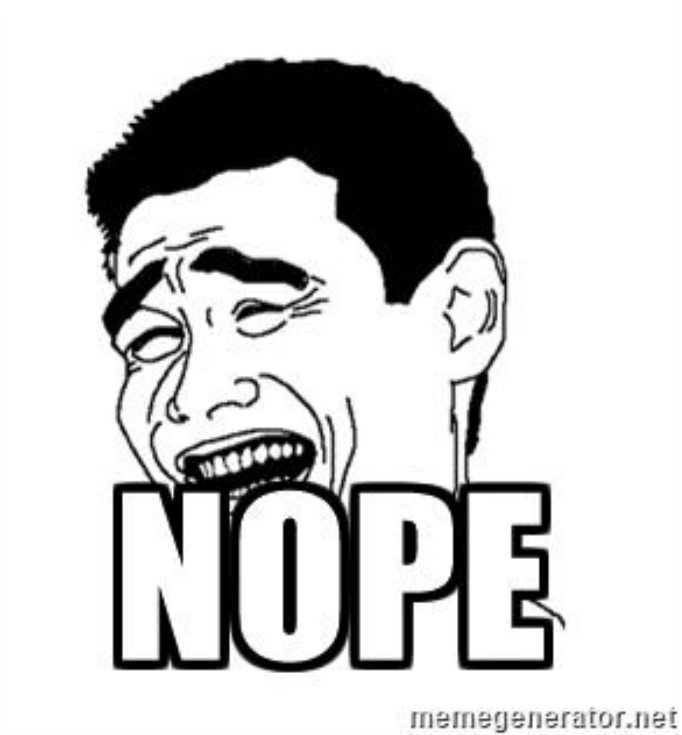
id	first_name	last_name	email
1	Jerre	Nouws	jerre@pau.be
NULL	NULL	NULL	NULL

Update

```
update person
set email='guy@pau.be'
where id = 1;
```

id	first_name	last_name	email
1	Jerre	Nouws	guy@pau.be
NULL	NULL	NULL	NULL

Sorry...



Delete

```
delete from person
where id = 1;
```

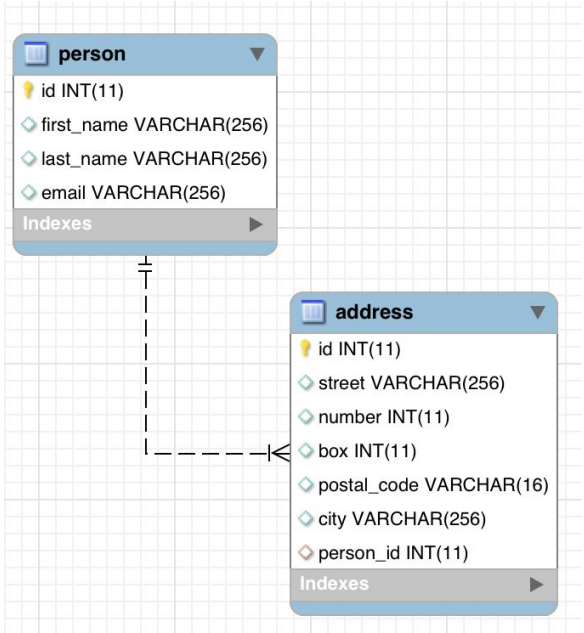
	id	first_name	last_name	email
▶	NULL	NULL	NULL	NULL

Relaties

```
create table address(
    id int auto_increment primary key,
    street varchar(256),
    number int,
    box int,
    postal_code varchar(16),
    city varchar(256),
    person_id int,

    constraint foreign key(person_id) references person(id)
);
```

Relaties


person
id INT(11)
first_name VARCHAR(256)
last_name VARCHAR(256)
email VARCHAR(256)
Indexes
address
id INT(11)
street VARCHAR(256)
number INT(11)
box INT(11)
postal_code VARCHAR(16)
city VARCHAR(256)
person_id INT(11)
Indexes

Constraint

- Zorgt ervoor dat de link tussen de tables niet zomaar verwijderd kan worden.
  - Parent table kan niet verwijderd worden als de child table nog bestaat

Relaties, relaties, relaties,...

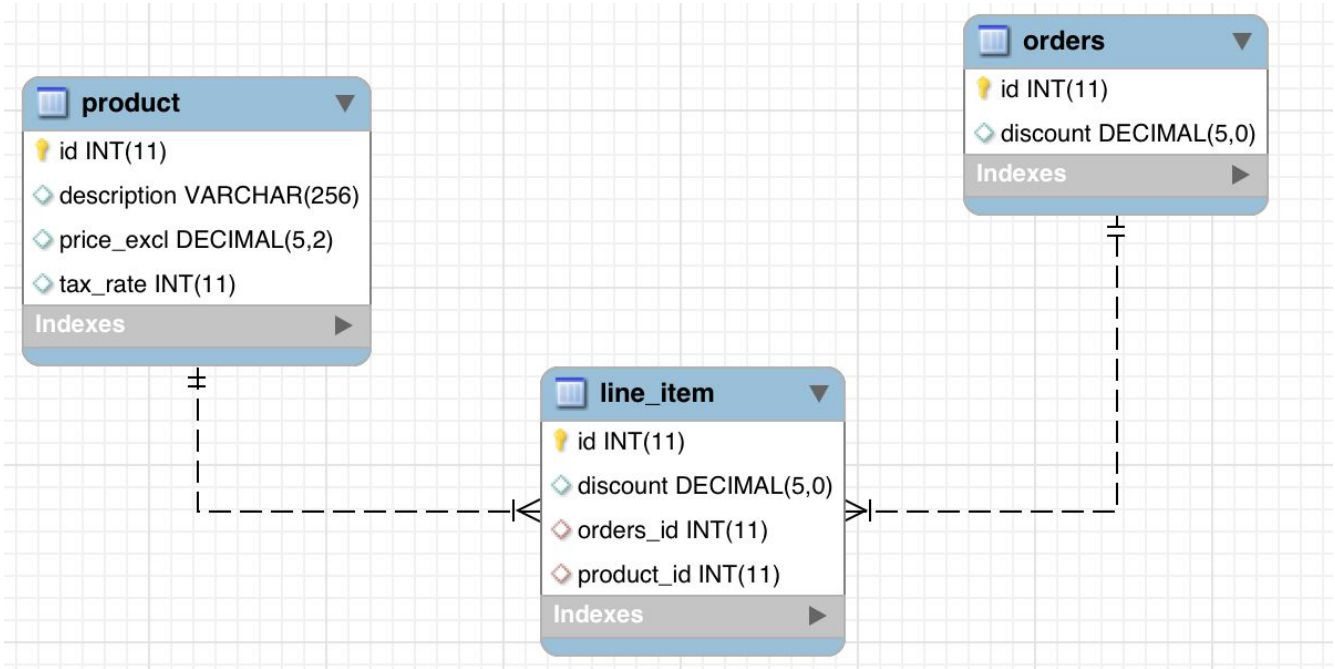
```
create table product(
    id int auto_increment primary key,
    description varchar(256),
    price_excl decimal(5,2),
    tax_rate int
);

create table orders(
    id int auto_increment primary key,
    discount decimal(5.2)
);

create table line_item(
    id int auto_increment primary key,
    discount decimal(5.2),
    orders_id int,
    product_id int,

    constraint foreign key(orders_id) references orders(id),
    constraint foreign key(product_id) references product(id)
);
```

Relaties, relaties, relaties,...



Normalisatie

- Redudantie tegengaan
- Tot 5 levels
- meest gebruikte vorm => 3de normaalvorm

Table wijzigen

```
alter table address
add isShippingAddress tinyint;


alter table orders
add person_id int;

alter table orders
add constraint fk_persons foreign key(person_id) references person(id);
```

Data consistentie afdwingen

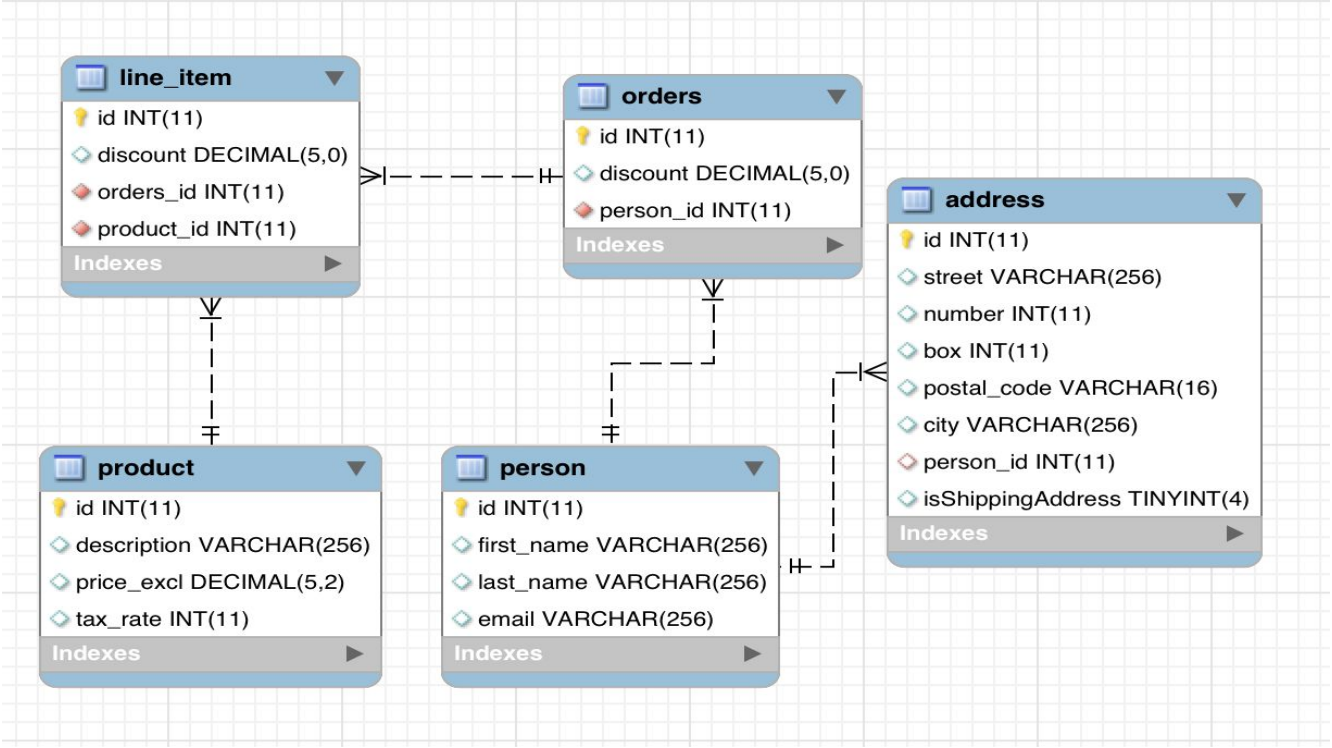
- Orders moeten altijd aan een persoon hangen
- Line-items moeten altijd aan een product en order hangen

```
alter table orders
modify person_id int not null default 0;

alter table line_item
modify orders_id int not null;

alter table line_item
modify product_id int not null;
```

Wat hebben we bereikt?



QA time