

# DBIx::Class

databaserijen als objecten



# Stukje SQL

```
begin;  
  
create table album_artist  
    ( id          serial not null primary key  
      , name      varchar not null  
      , description varchar  
    ) ;  
  
create table album  
    ( id          serial not null primary key  
      , name      varchar not null  
      , album_artist integer not null references album_artist(id)  
      , year      integer  
    ) ;  
  
create table song  
    ( id serial not null primary key  
      , name varchar not null  
      , album integer references album(id)  
      , track integer  
    ) ;  
  
commit;
```



# Van SQL naar ORM

```
dbicdump -o dump_directory=lib/ \  
         -o debug=1              \  
         Ztreet::Schema          \  
         dbi:Pg:dbname=nlpw2013 \  
         abeltje
```



# Van SQL naar ORM

```
dbicdump -o dump_directory=lib/ \  
         -o debug=1              \  
         Ztreet::Schema          \  
         dbi:Pg:dbname=nlpw2013 \  
         abeltje
```

live-demo...



# begrippen

- schema
  - het startpunt van je orm
    - databaseconnectie
    - toegang tot de data (resultset)
    - transacties
- resultset
  - de centrale entiteit
  - een verzameling van databaserijobjecten



# basis CRUD

- Create:
  - `$schema->resultset('Tabel')->create()`
- Read:
  - `$schema->resultset('Tabel')->find()`
  - `$schema->resultset('Tabel')->search()`
- Update:
  - `$object->Update()`
- Delete:
  - `$object->Delete()`



# Verbinden met de database

```
#!/usr/bin/perl
use warnings;
use strict;

use Ztreet::Schema;

# it's just like DBI...
my $db = Ztreet::Schema->connect(
    'dbi:Pg:dbname=nlpw2013',
    'abeltje',
);
```



# Create code

```
my $artist = $db->resultset('AlbumArtist')->find_or_create(  
    {  
        name => 'doe maar',  
        description => 'doe maar',  
    }  
);  
  
my $album = $artist->create_related(  
    'albums',  
    {  
        name => 'Skunk',  
        year => 1980,  
    }  
);
```





# Create code 2

```
my @songs = (  
    'Sinds Een Dag Of Twee',  
    'Te Laat',  
    'Bella Donna',  
    'Smoorverliefd',  
    'Nederwiet',  
    'Niks Voor Jou',  
    'Mis',  
    'Ruma Saja',  
    'De Laatste Keer',  
    'Dansen Met Alice',  
);  
  
for my $cnt (1 .. @songs) {  
    $album->create_related(  
        'songs',  
        {  
            track => $cnt,  
            name  => $songs[$cnt - 1],  
        }  
    );  
}
```



# Read code

```
#!/usr/bin/perl
use warnings;
use strict;

use Data::Dumper; $Data::Dumper::Indent = 1; $Data::Dumper::Sortkeys = 1;
use Ztreet::Schema;

my $schema = Ztreet::Schema->connect(
    'dbi:Pg:dbname=nlpw2013',
    'abeltje'
);

my ($album) = $schema->resultset('Album')->search(
    { name => 'Skunk' }
);

printf "%s - %s\n", $album->name, $album->album_artist->name;
```



# objecten om uit te breiden

```
# Created by DBIx::Class::Schema::Loader v0.07020 @ 2013-04-01 23:26:28
# DO NOT MODIFY THIS OR ANYTHING ABOVE! md5sum:RjMKHgbIzoWDG2aV9JUE0A

sub tekst_dump {
    my $self = shift;

    return sprintf "%s - %s", $self->album_artist->name, $self->name;
}
```

