Implementing single table inheritance

Relational databases do not support inheritance. If we need to store inheritance in the database, we should somehow support it through code. This code should be efficient, so that it should generate as few JOINs as possible. A common solution to this problem was described by Martin Fowler and is named single table inheritance.

When we use this pattern, we store all the class tree data in a single table and use the type field to determine a model for each row.

As an example, we will implement the single table inheritance for the following class tree:

Car |- SportCar |- FamilyCar

Getting ready

1. Create a new application using the Composer package manager, as described in the official guide at [http://www.yiiframework.com/doc-2.0/gurde-start-installation.html](http://www.yiiframework.com/doc-2.0/guide-start-installation.html).

2. Create and set up a database. Add the following table:

DROP TABLE IF EXISTS 'car';

CREATE TABLE 'car' (

'id' int(10) UNSIGNED NOT NULL AUTO\_INCREMENT,

'name' varchar(255) NOT NULL,

'type' varchar(100) NOT NULL,

PRIMARY KEY ('id')

);

INSERT INTO 'car' ('name', 'type')

VALUES ('Ford Focus', 'family'),

('Opel Astra', 'family'),

('Kia Ceed', 'family'),

('Porsche Boxster', 'sport'),

('Ferrari 550', 'sport');

3. Use Gii to create a Car model for the car table and generate ActiveQuery for the Car model.

How to do it...

1. Add the following method and property to models/CarQuery .php: /\*\*

\* @var \*/

public $type;

/\*\*

\* @param \yii\db\QueryBuilder $builder