Redis database driver

Это расширение позволяет использовать хранилище ключей и значений Redis в любом проекте на платформе Yii2. Он содержит обработчики кэша и хранилища сеансов, а также расширение, реализующее шаблон ActiveRecord для доступа к записям базы данных Redis.

Подготовка

1. Создайте новое приложение с помощью диспетчера пакетов Composer, как описано в официальном руководстве по адресу  
   <http://www.yiiframework.com/doc-2.0/guide-start-installation.html>.   
   По русски <http://yiiframework.domain-na.me/doc/guide/2.0/ru/start-installation>.
2. Установите хранилище: <http://redis.io>.
3. Установите все миграции с помощью следующей команды:

composer require yiisoft/yii2-redis

Как это сделать...

Прежде всего, настройте класс соединения в файле конфигурации:

return [

//....

'components' => [

'redis' => [

'class' => 'yii\redis\Connection',

'hostname' => 'localhost',

'port' => 6379,

'database' => 0,

],

]

];

Прямое использование

Для низкоуровневой работы с командами Redis можно использовать метод executeCommand компонента connection:

Yii::$app->redis->executeCommand('hmset', ['test\_collection', 'keyl', 'vall', 'key2', 'val2']);

Можно также использовать упрощенные сочетания вместо вызовов executeCommand:

Yii::$app->redis->hmset('test\_collection', 'key1','val1','key2','val2')

Использование ActiveRecord

Для доступа к записям Redis через шаблон ActiveRecord класс record должен быть расширен из базового класса yii\redis\ActiveRecord и реализовать метод attributes:

class Customer extends \yii\redis\ActiveRecord  
{

public function attributes()

{

return ['id', 'name', 'address', 'registration\_date'];

}

public function getOrders()

{

return $this->hasMany(Order::className(), ['customer\_id' => 'id']);

}

}

Первичный ключ любой модели может быть определен с помощью метода primaryKey (), который по умолчанию имеет значение id, если не указан. Первичный ключ необходимо поместить в список атрибутов, если он не указан вручную в методе primaryKey.

Ниже приведен пример использования:

$customer = new Customer();

$customer->name = 'test';

$customer->save();

echo $customer->id;

// id will automatically be incremented if not set explicitly  
// find by query

$customer = Customer::find()->where(['name' => 'test'])->one();

Как это работает...

Расширение предоставляет компонент подключения для низкоуровневого доступа к записям хранения Redis.

Вы также можете использовать и ActiveRecord-похожие модели с ограниченным набором методов (where(), limit(), offset(), и indexBy()). Другие методы не существуют, поскольку Redis не поддерживает запросы SQL.

В Redis нет таблиц, поэтому Вы не можете определить отношения через имя таблицы соединений. Вы можете определить отношения "многие ко многим" только через другие отношения hasMany.

Для общей информации о том, как использовать Yii’s ActiveRecord, пожалуйста см. Глава 3, ActiveRecord, Model, and Database.

Смотрите так же

* Дополнительные сведения об использовании расширения см. в разделе:

° [https ://github.com/yiisoft/yii2-redis/blob/ma stpr/docs/guide/RF ADMF.md](https://github.com/yiisoft/yii2-redis/blob/master/docs/guide/README.md)

По русски https://github.com/yiisoft/yii2-redis/tree/master/docs/guide-ru  
o [http://www.viiframework.com/doc-2.0/ext-redis-index.html](http://www.yiiframework.com/doc-2.0/ext-redis-index.html)

* Сведения о "ключ-значение" redis для хранения, см.: <http://redis.io/documentation>
* Глава 3, ActiveRecord, Model, and Database для использования ActiveRecord