$user = new User;

$list = [

['id'=>1, 'name'=>'thinkphp', 'email'=>'thinkphp@qq.com'],

['id'=>2, 'name'=>'onethink', 'email'=>'onethink@qq.com'],

];

$user->saveAll($list, false);

静态方法

还可以直接静态调用create方法创建并写入：

$user = User::create([

'name' => 'thinkphp',

'email' => 'thinkphp@qq.com'

]);

echo $user->name;

echo $user->email;

echo $user->id; // 获取自增ID

和save方法不同的是，create方法返回的是当前模型的对象实例。

助手函数

系统提供了model助手函数用于快速实例化模型，并且使用单例实现，例如：

// 使用model助手函数实例化User模型

$user = model('User');

// 模型对象赋值

$user->data([

'name' => 'thinkphp',

'email' => 'thinkphp@qq.com'

]);

$user->save();

或者进行批量新增：

$user = model('User');

// 批量新增

$list = [

['name'=>'thinkphp','email'=>'thinkphp@qq.com'],

['name'=>'onethink','email'=>'onethink@qq.com']

];

$user->saveAll($list);