

Интеграция базы данных клиентов с использованием фреймворка Yii2

Ран Максим Александрович

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье рассматривается процесс подключения и интеграции базы данных клиентов к фреймворку Yii2, установленному на локальный сервер. В качестве примера используется база данных clients, содержащая информацию о клиентах, их статусе и контактных данных. Описан процесс создания таблиц, настройки подключения и вывода данных на веб-страницу. Статья демонстрирует, как эффективно использовать Yii2 для работы с базами данных в веб-приложениях.

Ключевые слова: Yii2, база данных, MySQL, интеграция, клиенты, веб-приложение, PHP, phpMyAdmin.

Integration of the customer database using the Yii2 framework

Ran Maksim Aleksandrovich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article discusses the process of connecting and integrating a customer database to the Yii2 framework installed on a local server. As an example, the client's database is used, which contains information about clients, their status, and contact information. The process of creating tables, configuring connections, and displaying data on a web page is described. The article demonstrates how to effectively use Yii2 to work with databases in web applications.

Keywords: Yii2, database, MySQL, integration, clients, web application, PHP, phpMyAdmin.

1 Введение

1.1 Актуальность

В современном мире веб-приложения играют ключевую роль в управлении данными, особенно когда речь идет о взаимодействии с клиентами. Эффективное хранение и обработка информации о клиентах, таких как их контактные данные, статусы и предпочтения, являются важной задачей для многих компаний. Использование баз данных позволяет не только структурировать и сохранять эти данные, но и обеспечивать быстрый доступ к ним, что значительно повышает эффективность работы веб-приложений.

Фреймворк Yii2, благодаря своей гибкости и производительности, является одним из наиболее популярных инструментов для разработки веб-приложений на PHP. Его интеграция с базами данных, такими как MySQL, позволяет создавать мощные и масштабируемые решения для управления клиентской базой. Это особенно актуально для компаний, которые стремятся автоматизировать процессы взаимодействия с клиентами, улучшить качество обслуживания и повысить уровень персонализации услуг.

1.2 Обзор исследований

В последние годы фреймворк Yii2 активно используется для разработки различных веб-приложений, что подтверждается множеством научных исследований. В работах различных авторов рассматриваются подходы к созданию серверных приложений, интернет-магазинов, систем складского учета и других решений на базе Yii2.

П. Ю. Масолыкин и С. Ю. Александров рассмотрели процесс создания серверных приложений с использованием Yii2. Описано, что фреймворк предоставляет мощные инструменты для работы с базами данных, что делает его идеальным выбором для разработки сложных веб-приложений. Их исследование демонстрирует, как Yii2 может быть использован для создания масштабируемых и производительных решений [1]. И. В. Филимонов и И. Б. Разин описали подходы к созданию интернет-магазинов с использованием Yii2. Акцентировано внимание на инновационных методах интеграции модулей, что позволяет повысить гибкость и функциональность веб-приложений. Их работа демонстрирует, как Yii2 может быть использован для создания сложных e-commerce решений [2]. Д. Р. Салихов и А. Р. Халиков рассмотрели процесс создания системы складского учета с использованием Yii2. Фреймворк позволяет эффективно управлять большими объемами данных, что делает его подходящим для разработки систем учета и управления ресурсами [3]. Д. Р. Салихов подробно описал процесс создания и управления миграциями баз данных в Yii2. Миграции позволяют эффективно управлять изменениями в структуре базы данных, что особенно важно при разработке крупных проектов [4]. Д. Р. Вихляев рассмотрел использование Yii2 для работы с внешними API, в частности, для получения данных от метеостанций. Описано как Yii2 может быть использован для интеграции с внешними системами и обработки данных в реальном времени [5].

1.3 Цель исследования

Целью исследования является разработка и реализация процесса интеграции базы данных клиентов с фреймворком Yii2, установленным на локальный сервер. В рамках исследования планируется создать базу данных, содержащую информацию о клиентах (идентификатор, имя, статус и электронную почту), настроить её подключение к Yii2, а также разработать механизм вывода данных на веб-страницу. Это позволит продемонстрировать

эффективность использования Yii2 для управления клиентскими данными и создания современных веб-приложений.

2 Материалы и методы

Для выполнения работы использовались следующие инструменты и технологии:

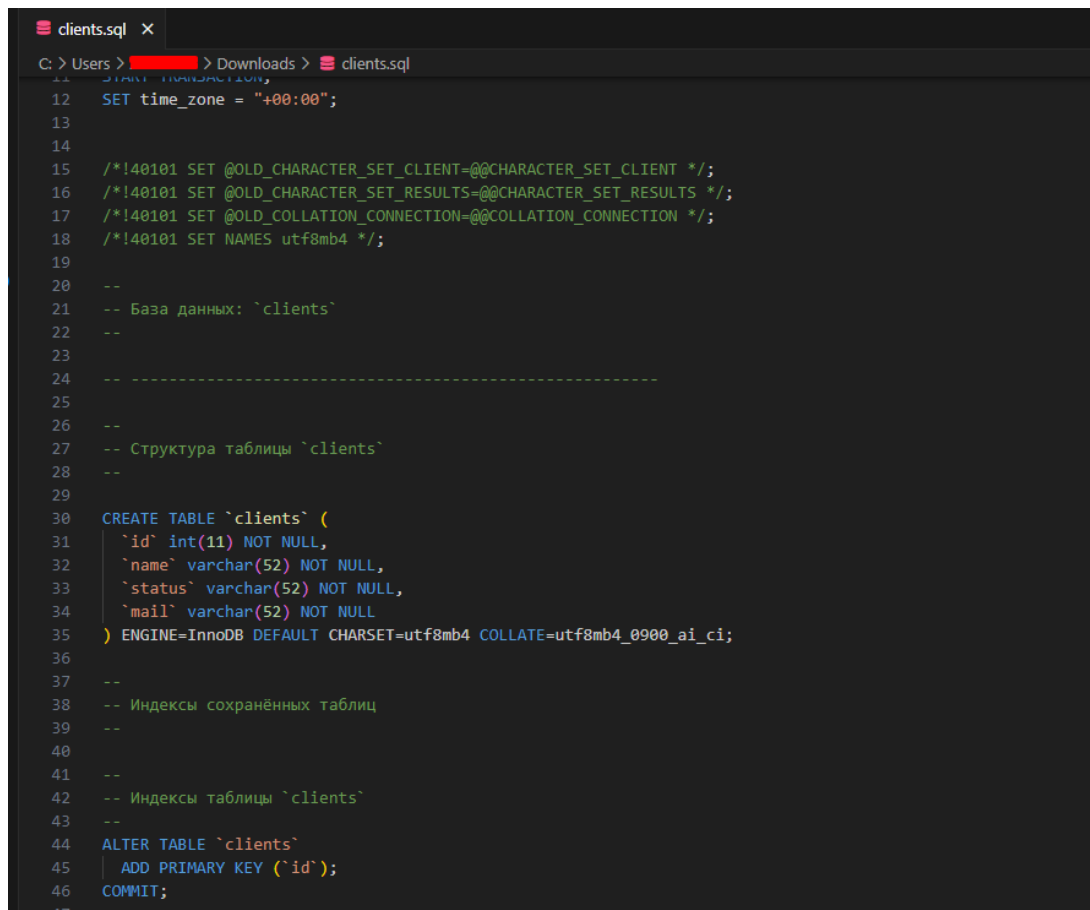
- Фреймворк Yii2: Установлен на локальный сервер для разработки веб-приложения.
- База данных MySQL: Создана база данных clients с таблицей, содержащей поля id, name, status и mail.
- phpMyAdmin: Использовался для управления базой данных и выполнения SQL-запросов.
- Локальный сервер: Настроен на базе OpenServer для работы с PHP, MySQL и Yii2.

3 Результаты и обсуждения

Для начала необходимо создать базу данных. В данной работе используется база данных clients, которая была создана с помощью phpMyAdmin. Внутри базы данных была создана таблица с именем clients, содержащая следующие поля:

1. id (уникальный идентификатор клиента, тип INT, первичный ключ),
2. name (имя клиента, тип VARCHAR),
3. status (статус клиента, тип VARCHAR),
4. mail (электронная почта клиента, тип VARCHAR).

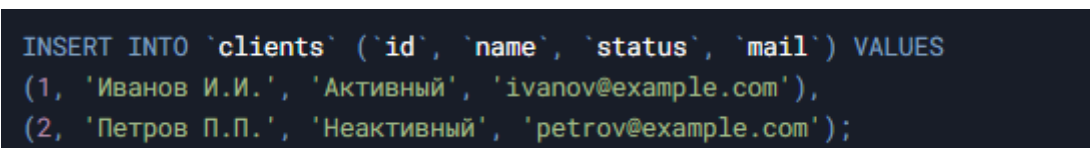
Структура таблицы представлена на рисунке 1.



```
clients.sql x
C:\Users\> Downloads > clients.sql
12 SET time_zone = "+00:00";
13
14
15 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
16 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
17 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
18 /*!40101 SET NAMES utf8mb4 */;
19
20 --
21 -- База данных: `clients`
22 --
23
24 -- -----
25
26 --
27 -- Структура таблицы `clients`
28 --
29
30 CREATE TABLE `clients` (
31   `id` int(11) NOT NULL,
32   `name` varchar(52) NOT NULL,
33   `status` varchar(52) NOT NULL,
34   `mail` varchar(52) NOT NULL
35 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
36
37 --
38 -- Индексы сохранённых таблиц
39 --
40
41 --
42 -- Индексы таблицы `clients`
43 --
44 ALTER TABLE `clients`
45   ADD PRIMARY KEY (`id`);
46 COMMIT;
47
```

Рис. 1. Структура БД

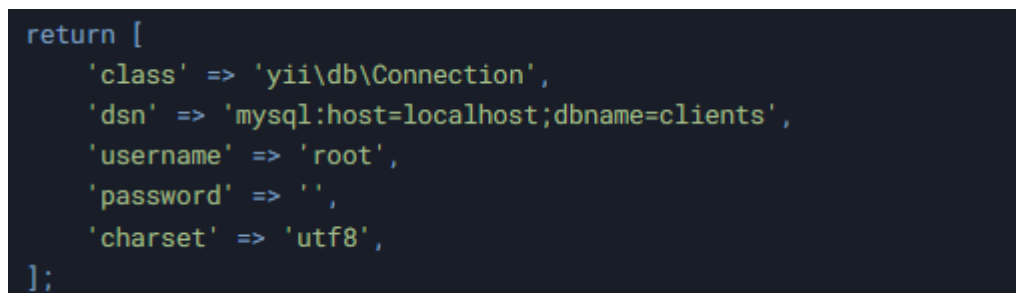
После создания таблицы она была заполнена тестовыми данными. Пример SQL-запроса для добавления данных (рис. 2).



```
INSERT INTO `clients` (`id`, `name`, `status`, `mail`) VALUES
(1, 'Иванов И.И.', 'Активный', 'ivanov@example.com'),
(2, 'Петров П.П.', 'Неактивный', 'petrov@example.com');
```

Рис. 2. SQL-запрос для заполнения данных

Для подключения базы данных к фреймворку Yii2 необходимо настроить конфигурационный файл db.php, который находится в папке config. В этом файле указываются параметры подключения к базе данных (рис. 3).



```
return [
    'class' => 'yii\db\Connection',
    'dsn' => 'mysql:host=localhost;dbname=clients',
    'username' => 'root',
    'password' => '',
    'charset' => 'utf8',
];
```

Рис. 3. Настройка конфигурационного файла

В данном файле:

1. dsn – строка подключения к базе данных (указывается хост и имя базы данных),
2. username – имя пользователя базы данных,
3. password – пароль пользователя,
4. charset – кодировка данных.

Для работы с таблицей clients в Yii2 необходимо создать модель. Модель была создана с помощью консольной команды Gii, встроенного генератора кода Yii2. Модель Clients была размещена в папке models. Пример кода модели (рис. 4).

```
namespace app\models;

use yii\db\ActiveRecord;

class Clients extends ActiveRecord
{
    public static function tableName()
    {
        return 'clients';
    }
}
```

Рис. 4. Создание модели

Модель наследует класс ActiveRecord, который предоставляет методы для работы с базой данных. Метод tableName() указывает, с какой таблицей связана модель.

Для обработки запросов и вывода данных был создан контроллер ClientsController. Контроллер был размещен в папке controllers. Пример кода контроллера (рис. 5).

```
namespace app\controllers;

use yii\web\Controller;
use app\models\Clients;

class ClientsController extends Controller
{
    public function actionIndex()
    {
        $clients = Clients::find()->all();
        return $this->render('index', ['clients' => $clients]);
    }
}
```

с. 5. Настройка контроллера

Метод `actionIndex()` выполняет запрос к базе данных для получения всех записей из таблицы `clients`. Данные передаются в представление `index` для отображения.

Для отображения данных на веб-странице было создано представление `index.php`, которое находится в папке `views/clients`. Пример кода представления (рис. 6).

```
<?php
use yii\helpers\Html;
?>
<h1>Список клиентов</h1>
<table class="table">
    <thead>
        <tr>
            <th>ID</th>
            <th>Имя</th>
            <th>Статус</th>
            <th>Email</th>
        </tr>
    </thead>
    <tbody>
        <?php foreach ($clients as $client): ?>
            <tr>
                <td><?= Html::encode($client->id) ?></td>
                <td><?= Html::encode($client->name) ?></td>
                <td><?= Html::encode($client->status) ?></td>
                <td><?= Html::encode($client->mail) ?></td>
            </tr>
        <?php endforeach; ?>
    </tbody>
</table>
```

Рис. 6. Создание представления

Данные из таблицы `clients` выводятся в виде HTML-таблицы. Используется метод `Html::encode()` для безопасного вывода данных. После настройки модели, контроллера и представления данные из базы данных были успешно выведены на веб-страницу (рис. 7).

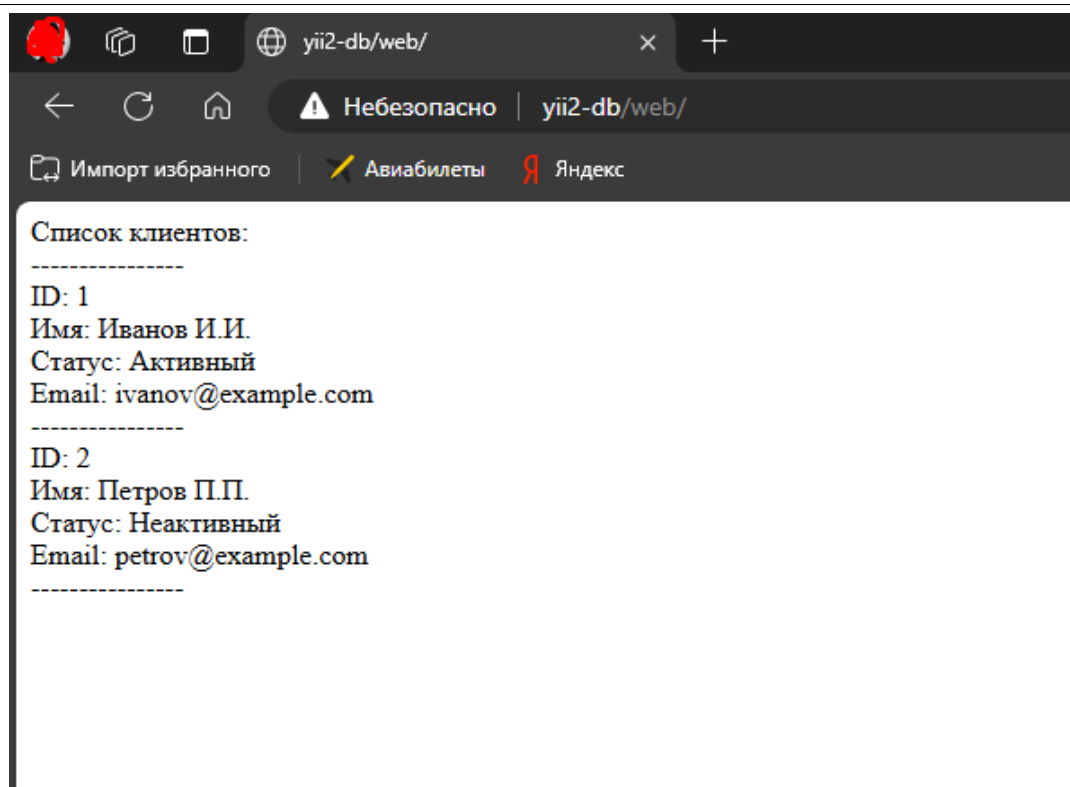


Рис. 7. Вывод клиентов

Для проверки работоспособности приложения необходимо перейти по следующей ссылке в браузере: <http://localhost/yii2/web/index.php?r=clients>.

Выводы

В данной работе был успешно реализован процесс подключения базы данных MySQL к фреймворку Yii2 и вывода данных на веб-страницу. В качестве примера использовалась база данных clients, содержащая информацию о клиентах, их статусах и контактных данных. Были выполнены следующие шаги: создание базы данных и таблицы с помощью phpMyAdmin, настройка подключения базы данных к Yii2 через конфигурационный файл db.php, создание модели Clients для работы с таблицей базы данных, разработка контроллера ClientsController для обработки запросов и передачи данных в представление, а также создание представления index.php для отображения данных в виде HTML-таблицы. После настройки всех компонентов данные из базы данных были успешно выведены на веб-страницу, что подтвердило корректность подключения и работы приложения. Результаты работы демонстрируют, что фреймворк Yii2 является мощным инструментом для разработки веб-приложений, интегрированных с базами данных. Его гибкость и простота использования позволяют быстро создавать функциональные решения для управления данными. В дальнейшем планируется расширить функциональность приложения, добавив возможности редактирования и удаления записей, а также улучшив интерфейс для более удобной работы с данными.

Библиографический список

1. Масолыкин П. Ю., Александров С. Ю. Разработка серверного приложения на базе фреймворка Yii2 // Научно-техническое творчество аспирантов и студентов. Материалы всероссийской научно-технической конференции студентов и аспирантов: в 2 частях. 2018. С. 214–216.
2. Филимонов И. В., Разин И. Б. Интернет-магазин с использованием инновационных методов соединения модулей в среде разработки Yii2 // Инновационное развитие легкой и текстильной промышленности. Сборник материалов Всероссийской научной студенческой конференции. 2017. С. 97–99.
3. Салихов Д. Р., Халиков А. Р. Разработка веб-приложения для складского учета на базе фреймворка Yii2 // Инновации в науке и практике. Сборник статей по материалам X международной научно-практической конференции. В 4 частях. Ответственный редактор Халиков А. Р., 2018. С. 192–196.
4. Салихов Д. Р. Разработка миграций баз данных на базе фреймворка Yii2 // Актуальные вопросы современной науки. Сборник статей по материалам XIX международной научно-практической конференции. 2019. С. 53–57.
5. Вихляев Д. Р. Обращение к данным метеостанций на Yii2 с помощью REST API // Постулат. 2022. № 6 (80).