

# Документация к сервису «База данных»

Версия 3

## Авторизация в приложении

Чтобы попасть в приложение, нужно зарегистрироваться. Сохраните аутентификационные данные — они пригодятся при работе с API.

## Рабочие вкладки

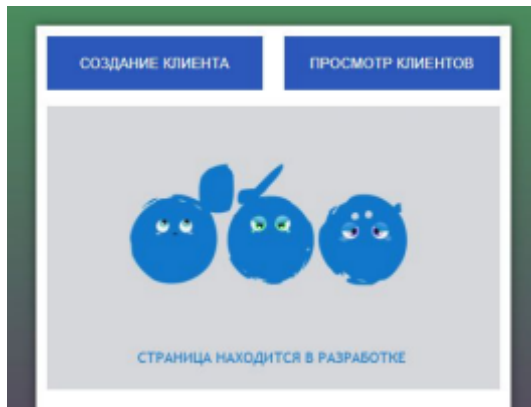
Приложение «База данных» состоит из двух вкладок: «Создание клиента» и «Просмотр клиентов».

- Первая вкладка «Создание клиента» позволяет сохранить на сервер карточку клиента.



The screenshot shows a web form titled «Создание клиента» (Create Client). At the top, there are two buttons: «СОЗДАНИЕ КЛИЕНТА» (Create Client) in green and «ПРОСМОТР КЛИЕНТОВ» (View Clients) in blue. Below these are five input fields: «Фамилия» (Surname), «Имя» (Name), «Отчество» (Patronymic), «ДД.ММ.ГГГГ» (Date of birth), and «Место проживания» (Place of residence). At the bottom of the form is a green button labeled «Создать» (Create).

- Вторая вкладка «Просмотр клиентов» — это список клиентов, в котором можно открыть карточку каждого из них. Список можно редактировать и фильтровать по параметрам: ФИО, дата рождения и место проживания. Графический интерфейс страницы находится в разработке, а серверная логика уже реализована. Подробнее об организации серверной логики можно прочитать в разделе «Методы API» ниже.



На вкладке «Просмотр клиентов» должны быть возможными удаление конкретного клиента и полная очистка базы.

## Карточка клиента

Состоит из шести полей:

- Id — уникальное поле, заполняется автоматически на backEnd;
- «Имя» — обязательное поле;
- «Фамилия» — обязательное поле;
- «Отчество» — необязательное поле;
- «Дата рождения» — необязательное поле;
- «Место проживания» — необязательное поле.

База клиентов чувствительна к регистру: значения «Дмитрий» и «дмитрий» — разные.

## Валидации

Валидации графического интерфейса.

- Незаполненные поля «Имя» и «Фамилия» выделяются красной рамкой.

Валидации сервера.

- Ответ сервера «Name and Surname must be filled» появляется при попытке создать пользователя без имени и фамилии.
- Ответ сервера «User already exists. UserId: int» появляется при попытке создать или обновить пользователя с существующим Id.

На второй вкладке предусмотрен просмотр пользователей. Однако функциональность ещё не реализована, поэтому стоит заглушка.

## Методы API

Карточка клиента сохраняется в базу данных.

Поле в графическом интерфейсе	Поле в в таблице на сервере
-------------------------------	-----------------------------

Имя	name
Фамилия	surname
Отчество	patronymic
Дата рождения	dateOfBirth
Место проживания	location

Для работы с сервером реализованы методы API.

Приложение «База данных» развернуто на трёх площадках:

1. <http://api-qa.skillbox.ru/practice1> — для практических заданий и видео.
2. <http://api-qa.skillbox.ru/practice2> — для домашней работы по модулю «Модуль 21. Тестирование API. Чтение данных с сервера».
3. <http://api-qa.skillbox.ru/practice3> — для домашней работы по модулю «Модуль 22. Тестирование API. Манипуляция данными на сервере».

У каждой площадки есть API-методы. Чтобы обратиться к ним, в запросе нужно вместо **N** указать номер площадки: 1, 2 или 3 (например practice1).

Метод	Название	Запрос
GET	Получение списка клиентов	<a href="http://api-qa.skillbox.ru/practiceN/api/users/">http://api-qa.skillbox.ru/practiceN/api/users/</a>
GET	Получение конкретного клиента	<a href="http://api-qa.skillbox.ru/practiceN/api/users/{userId}">http://api-qa.skillbox.ru/practiceN/api/users/{userId}</a>
POST	Создание клиента	<a href="http://api-qa.skillbox.ru/practiceN/api/users/create">http://api-qa.skillbox.ru/practiceN/api/users/create</a>  Пример body в формате json: <pre>{   "itemsOfClients":   [     {       "id": "int",       "person": {         "name": "string",         "surname": "string",         "patronymic": "string",         "dateOfBirth": "datetime",         "location": "string"       }     }   ] }</pre>

		<pre>     ]   } </pre> <p>При создании клиента через графический интерфейс Id задаёт сервер.</p> <p>При создании каждого нового пользователя сервер определяет максимальный Id в списке и прибавляет к нему 1.</p>
PUT	Обновление клиента	<a href="http://api-qa.skillbox.ru/practiceN/api/users/{userId}/update">http://api-qa.skillbox.ru/practiceN/api/users/{userId}/update</a> <pre> {   "update": {     "name": "string",     "surname": "string",     "dateOfBirth": "datetime",     "location": "string"   } } </pre>
DELETE	Удаление конкретного клиента	<a href="http://api-qa.skillbox.ru/practiceN/api/users/{userId}/delete">http://api-qa.skillbox.ru/practiceN/api/users/{userId}/delete</a>
DELETE	Удаление всех клиентов	<a href="http://api-qa.skillbox.ru/practiceN/api/users/allUsers/delete">http://api-qa.skillbox.ru/practiceN/api/users/allUsers/delete</a>