



База данных

представляет собой совокупность структурированных данных, хранимых в памяти вычислительной системы и отображающих состояние объектов и их взаимосвязей в рассматриваемой предметной области



Система управления базами данных (СУБД)

это комплекс языковых и программных средств, предназначенный для создания, ведения и совместного использования БД многими пользователями.

- ✓ MySQL
- ✓ MS SQL Server
- ✓ SQLite
- ✓ Oracle Database



Технологии компонентного программирования

Требования к базе данных

- ✓ хранение данных;
- ✓ обеспечение доступа к данным;
- ✓ наличие системы пользовательских привилегий;
- ✓ возможность резервного копирования и восстановления данных;
- ✓ наличие универсального языка;
- ✓ эффективный доступ к данным.



Модели БД

- ✓ **Реляционная** - данные представляются в виде двумерных таблиц. Табличная структура отражает отношения между реальными объектами и их характеристиками.
- ✓ **Иерархическая.**
- ✓ **Сетевая.**



Технологии компонентного программирования

Сущность – любой различимый объект (объект, который мы можем отличить от другого), информацию о котором необходимо хранить в базе данных. Сущностями могут быть люди, места, самолеты, рейсы, вкус, цвет и т.д.

Атрибут — это свойство, описывающее определенный аспект объекта, значение которого следует зафиксировать в описании предметной области.

Связь – ассоциирование двух или более сущностей.



Технологии компонентного программирования

tbl_client		
	Имя поля	Тип данных
🔑	client_id	Счетчик
	client_fio	Текстовый
	client_phone	Текстовый
	client_mail	Текстовый
	client_address	Текстовый
	client_discount	Числовой
	client_summa	Числовой



Технологии компонентного программирования

tbl_client						
client_id	client_fio	client_phone	client_mail	client_address	client_discount	client_sumr
3	Кашкевич Александр Алексеевич	+375(29)7856966	kaa_1956@mail.ru	г.Гродно, ул. Пестрака 34, кв. 5	5	0



Первичный ключ

Столбец или набор столбцов для однозначного определения каждой строки таблицы



phpMyAdmin

phpMyAdmin - это программа написанная на PHP и предназначенная для управления сервером MySQL.

phpMyAdmin поддерживает широкий набор операций над MySQL. Наиболее часто используемые операции поддерживаются с помощью пользовательского интерфейса (управление базами данных, таблицами, полями, связями, индексами, пользователями, правами, и т. д.), одновременно вы можете напрямую выполнить любой SQL запрос.



Возможности phpMyAdmin

- ✓ Создание, просмотр, редактирование и удаление: баз данных, пользователей, таблиц и записей в них
- ✓ Удобный и понятный импорт и экспорт БД, таблиц и записей
- ✓ Понятный поиск
- ✓ Администрирование БД и пользователей



Вход в phpmyadmin для локального сервера:

phpMyAdmin

127.0.0.1

Базы данных SQL Состояние Пользователи Экспорт Импорт Настройки Синхронизировать Ещё

(Недавние таблицы) ...

- information_schema
- mysql
- performance_schema
- phpmyadmin

Основные настройки

Сопоставление кодировки соединения с MySQL : utf8_general_ci

Настройки внешнего вида

Язык - Language : Русский - Russian

Тема: pmahomme

Размер шрифта: 82%

Дополнительные настройки

Сервер баз данных

- Сервер: 127.0.0.1 via TCP/IP
- Программа: MySQL
- Версия программы: 5.5.25 - MySQL Community Server (GPL)
- Версия протокола: 10
- Пользователь: root@localhost
- Кодировка сервера: UTF-8 Unicode (utf8)

Веб-сервер

- Apache/2.2.22 (Win32) mod_ssl/2.2.22 OpenSSL/1.0.1c PHP/5.2.12
- Версия клиента базы данных: libmysql - 5.0.51a
- PHP расширение: mysqli



Меню `phrtyadmin`

- **Базы данных** – обзор имеющихся баз данных на сервере;
- **SQL** – позволяет создать и выполнить SQL-запрос к базе данных;
- **Экспорт и Импорт** – позволяют выполнить готовый sql-скрипт на сервере или сделать бэкап имеющейся на сервере базы данных.



Типы данных

- ✓ **INT** Диапазон от -2 147 483 648 до 2 147 483 647.
- ✓ **FLOAT** Число с плавающей точкой небольшой точности.
- ✓ **VARCHAR** Может хранить не более 255 символов.
- ✓ **DATE** Дата в формате ГГГГ-ММ-ДД
- ✓ **DATETIME** Дата и время в формате ГГГГ-ММ-ДД ЧЧ:ММ:СС



Создание базы данных

Чтобы создать новую базу данных следует перейти на вкладку «Базы данных»

The screenshot shows the phpMyAdmin interface. The top navigation bar includes the phpMyAdmin logo and several icons. The main navigation tabs are 'Базы данных', 'SQL', 'Состояние', 'Пользователи', and 'Экспорт'. The 'Базы данных' tab is active, displaying the 'Базы данных' section. Below this, there is a 'Создать базу данных' button with a question mark icon, a text input field, a 'Сравнение' dropdown menu, and a 'Создать' button. Below the input field, there is a table listing existing databases:

База данных	
<input type="checkbox"/> information_schema	Проверить привилегии
<input type="checkbox"/> mysql	Проверить привилегии
<input type="checkbox"/> performance_schema	Проверить привилегии
<input type="checkbox"/> phpmyadmin	Проверить привилегии
Всего: 4	



Указав название базы данных, следует нажать кнопку «Создать» и новая база отобразится слева в обозревателе.

phpMyAdmin



(Недавние таблицы) ...

- information_schema
- my_bd
- mysql
- performance_schema
- phpmyadmin

127.0.0.1

Базы данных SQL Состояние

Базы данных

Создать базу данных

My_BD

Сравнение

База данных

- | | |
|---|----------------------|
| <input type="checkbox"/> information_schema | Проверить привилегии |
| <input type="checkbox"/> My_BD | Проверить привилегии |
| <input type="checkbox"/> mysql | Проверить привилегии |
| <input type="checkbox"/> performance_schema | Проверить привилегии |
| <input type="checkbox"/> phpmyadmin | Проверить привилегии |

Всего: 5



Создание таблиц

Чтобы просмотреть имеющиеся в базе данных таблицы или создать новую таблицу, необходимо кликнуть в обозревателе по имени нужной базы.

В открывшейся вкладке следует указать имя таблицы и количество столбцов в ней и нажать кнопку «ОК»



phpMyAdmin



(Недавние таблицы) ... ▾

my_bd ▾

Таблиц в базе данных не обнаружено.

Создать таблицу

127.0.0.1 » my_bd

Структура

SQL

Поиск

Запрос по шаблону

Экспорт

Таблиц в базе данных не найдено

Создать таблицу

Имя:

Количество столбцов:



Технологии компонентного программирования

После того как таблица будет создана, в ней необходимо задать названия полей и их атрибуты.

После этих действий таблицу следует сохранить и она будет готова к работе с web-приложением (web-сайтом)



phpMyAdmin



(Недавние таблицы) ...

my_bd

my_table

Создать таблицу

Имя таблицы: Добавить поле(я)

Структура

Имя	Тип	Длина/значения	По умолчанию	Сравнение	Атрибуты	Null	Индекс
<input type="text"/>	INT	<input type="text"/>	Нет	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---
<input type="text"/>	INT	<input type="text"/>	Нет	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---
<input type="text"/>	INT	<input type="text"/>	Нет	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---
<input type="text"/>	INT	<input type="text"/>	Нет	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---
<input type="text"/>	INT	<input type="text"/>	Нет	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---

Комментарий к таблице:

Тип таблиц:

InnoDB

Сравнение:

Определение разделов (PARTITION):

Сохранить



Чтобы добавить запись в таблицу, следует:

1. Выбрать в обозревателе нужную базу данных;
2. Кликнуть по названию таблицы;
3. В Верхнем меню выбрать пункт «Вставить»



Технологии компонентного программирования

127.0.0.1 » my_bd » my_table

Обзор Структура SQL Поиск Вставить Экспорт Импорт Операции Слежен

Поле	Тип	Функция	Null	Значение
pole1	int(11)	<input type="text"/>		<input type="text" value="1"/>
pole2	varchar(200)	<input type="text"/>		<input type="text" value="name1"/>
pole3	varchar(200)	<input type="text"/>		<input type="text" value="name2"/>
pole4	varchar(100)	<input type="text"/>		<input type="text" value="name3"/>
pole5	varchar(100)	<input type="text"/>		<input type="text" value="name4"/>

OK



Чтобы добавить запись в таблицу, следует:

1. Выбрать в обозревателе нужную базу данных;
2. Кликнуть по названию таблицы;
3. В Верхнем меню выбрать пункт «Вставить»



Технологии компонентного программирования

127.0.0.1 » my_bd » my_table

Обзор Структура SQL Поиск Вставить Экспорт Импорт Операции Слежен

Поле	Тип	Функция	Null	Значение
pole1	int(11)	<input type="text"/>		<input type="text" value="1"/>
pole2	varchar(200)	<input type="text"/>		<input type="text" value="name1"/>
pole3	varchar(200)	<input type="text"/>		<input type="text" value="name2"/>
pole4	varchar(100)	<input type="text"/>		<input type="text" value="name3"/>
pole5	varchar(100)	<input type="text"/>		<input type="text" value="name4"/>

OK



Когда таблица будет содержать записи их можно будет просмотреть:

1. В обозревателе выбрать нужную базу данных;
2. Выбрать нужную таблицу (кликнуть по ее названию);
3. В верхнем меню выбрать кнопку «Обзор».



phpMyAdmin



(Недавние таблицы) ...

my_bd

my_table

Создать таблицу

127.0.0.1 » my_bd » my_table

Обзор Структура SQL Поиск Вставить Экспорт Импорт Операции

Отображает строки 0 - 0 (~1 всего), Запрос занял 0.0006 сек.

```
SELECT *  
FROM `my_table`  
LIMIT 0, 30
```

Профилирование [Быстрая правка] [Изменить]

Показать : Начальная строка: 0 Количество строк: 30 Заголовки каждые 100 строк

+ Параметры

		pole1	pole2	pole3	pole4	pole5		
<input type="checkbox"/>	Изменить	Копировать	Удалить	1	name1	name2	name3	name4

Отметить все / Снять выделение С отмеченными: Изменить Удалить Экспорт



SQL

Структурированный язык запросов.

Декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей СУБД.



Инструкция **INSERT INTO**

используются для вставки новых строк в таблицу

Синтаксис

INSERT INTO имя_таблицы **VALUES** (значение1, значение2, значение3,...);

Пример

INSERT INTO my_table **VALUES** ('1','OK','NO','HELLO');



Инструкция **SELECT**

позволяет производить выборки данных из таблиц и преобразовывать к нужному виду полученные результаты

Синтаксис

SELECT * **FROM** имя_таблицы;

Пример

SELECT * **FROM** my_table;



Пример

Выбрать поля с названиями **pole1** и **pole2** из
таблицы **my_table**

```
SELECT pole1, pole2 FROM my_table;
```

Выбрать все поля из таблицы **my_table** где
поле **id** равно 5

```
SELECT * FROM my_table WHERE id=5;
```



Инструкция **UPDATE**

изменяет имеющиеся данные в таблице

Синтаксис

UPDATE имя таблицы **SET** имя столбца =
новое_значение **WHERE** ...

Пример

```
UPDATE my_table SET pole1 = 'Vadim' WHERE id = 3;
```



Инструкция **DELETE** удаляет строки из таблиц

Синтаксис

DELETE FROM имя таблицы **WHERE** условие;

Пример

DELETE FROM my_table **WHERE** pole3 < 20;