

# **Базы данных лабораторные**

## **Лабораторная работа №1 QBE**

### **Цель работы**

Создание базы данных (таблицы и связи между ними) средствами СУБД Microsoft Access или LibreOffice Base.

### **Постановка задачи**

Предметная область: Деканат (успеваемость студентов).

Основные предметно-значимые сущности: Студенты, Группы студентов, Дисциплины, Успеваемость.

Основные предметно-значимые атрибуты сущностей:

- студенты – фамилия, имя, отчество, пол, дата рождения, место рождения, группа студентов;
- группы студентов – название, курс, семестр;
- дисциплины – название, количество часов;
- успеваемость – оценка, вид контроля.

Основные требования к функциям системы:

- выбрать успеваемость студентов по группам и дисциплинам.

Характеристики таблицы-объекта: Группы студентов. Таблица 1

Описание	Имя поля	Тип данных	Свойства поля:	
			Размер поля	Индексированное
Код группы (ключевое поле)	КодГруппы	счетчик	длинное целое	Установится автоматически
Название группы	Название	текстовый	6	Нет
Курс	Курс	числовой	длинное целое	Нет
Семестр	Семестр	числовой	длинное целое	Нет

Характеристики таблицы-объекта Студенты: Таблица 2

Описание	Имя поля	Тип данных	Свойства поля:	
			Размер поля	Индексированное
Код студента (ключевое поле)	КодСтудента	счетчик	длинное целое	Установится автоматически
Код группы (внешний ключ)	КодГруппы	числовой	длинное целое	Да. Совпадения допускаются
Фамилия	Фамилия	текстовый	20	Нет
Имя	Имя	текстовый	15	Нет
Отчество	Отчество	текстовый	15	Нет
Пол	Пол	текстовый	1	Нет
Дата рождения	Дата рожд	дата/	-	Нет

		время		
Место рождения	Место рожд	MEMO	-	Нет

Характеристики таблицы-объекта: Дисциплины. Таблица 3

Описание	Имя поля	Тип поля	Свойства поля:	
			Размер поля	Индексирован
Код дисциплины	КодДисциплины	счетчик.	длинное целое	Установится автоматически
Название дисциплины	Название	текстовый	15	Нет
Общий объем часов	КолЧасов	числовой	длинное целое	Нет

Характеристики таблицы-объекта: Успеваемость. Таблица 4

Описание	Имя поля	Тип поля	Свойства поля:	
			Размер поля	Индексирован
Код оценки (ключевое поле)	КодОценки	счетчик.	длинное целое	Установится автоматически
Код дисциплины (внешний ключ)	КодДисциплины	числовой	длинное целое	Да (Допускается)
Код студента (внешний ключ)	КодСтудента	числовой	длинное целое	Да (Допускается)
Оценка	Оценка	текстовый	8	Нет
Вид контроля (экзамен, зачет, рейтинг)	Вид контроля	текстовый	7	Нет

1. Создайте новую базу данных
  2. Создайте необходимые таблицы, согласно предметной области.
  3. Установите типы данных (счетчик, текстовый, числовой и т.д.), описание и другие необходимые свойства полей (размер поля, маску ввода, подпись, значение по умолчанию и т.д.) созданных таблиц.
  4. Определите первичные ключи в созданных таблицах.
  5. Определите необходимые связи между таблицами, задайте необходимые параметры обеспечения целостности данных и вид объединения.
  6. В схеме данных проверьте правильность созданных таблиц и связей между ними.
  7. Заполните созданные таблицы данными (минимум 10 записей на таблицу).
  8. Создайте необходимые запросы на выборку, выполняющие основные требования к функциям системы:
- 1) Создать запрос на выборку. Отображать все фамилии студентов,

которые получили оценки отл/А по всем дисциплинам (Успеваемость студентов).

2) Создать параметрический запрос. Создать запрос, в результате которого будет выводиться Фамилия студента, Название группы, Дисциплина и Оценка (отл/А), полученная студентом по дисциплине.  
9. Оформите отчет о выполнении лабораторной работы.

## **Пошаговое выполнение работы**

**1. Загрузить Microsoft Access**, выполнив действия: Пуск—>Программы — Microsoft Access (или выполнив щелчок на соответствующей пиктограмме на панели Microsoft Office).

### **2. Создать новую базу данных**

После загрузки MS Access на экране появится главное окно. При первом запуске Access в главном окне выводится область задач в режиме «Приступая к работе», с помощью которой можно открыть существующие БД и «Создать файл».

При выборе команды «Создать файл» в области задач изменится режим на «Создание файла».

При выборе команды «Новая база данных» откроется окно диалога «Файл новой базы данных», в котором необходимо выбрать имя диска и директории для хранения БД, а также имя БД (тип файла устанавливается по умолчанию «Базы данных Microsoft Office Access») и щелкнуть на кнопке «Создать», будет сохранен файл с расширением .mdb В главном окне приложения появится окно БД с назначенным именем, например «Деканат: база данных (формат Access 2000)».

### **3. Создать структуры таблиц**

В появившемся на экране окне БД Деканат: база данных выбрать вкладку Таблицы и щелкнуть на пиктограмме Конструктор на панели инструментов. Появится окно Конструктора таблиц.

#### **3.1. Создать структуру таблицы Группы студентов.**

В окне Конструктора таблиц заполнить разделы Имя поля, Тип данных и Свойства поля в соответствии с таблицей 1.

Для перехода от раздела к разделу использовать клавишу <Tab> или щелчок мышью на нужной ячейке.

В разделе Тип данных для изменения типа раскрыть окно выбора типа, выполнив щелчок по кнопке раскрытия списка, затем выполнить щелчок в строке, содержащей соответствующий тип.

При этом в нижней части экрана в разделе Свойства поля появляется информация о свойствах данного поля. При необходимости туда можно вносить изменения, выполнив щелчок в соответствующей строке, удалив предыдущее значение и введя новое. Дополнительно можно задать формат поля, условие на значение и т.д.

После заполнения разделов необходимо задать ключевое поле. В качестве ключевого поля выбрать поле КодГруппы.

Для создания ключевого поля выделить поле КодГруппы, выполнив щелчок слева от имени поля на полосе выделения. Выбрать в меню Правка — Ключевое поле или выполнить щелчок по пиктограмме Ключевое поле на панели инструментов приложения. Слева от имени

поля появится изображение ключа.

После создания структуры таблицы необходимо сохранить ее. Файл. — Сохранить, или. Сохранить, как... В окне диалога Сохранение ввести имя для сохранения созданной таблицы: **Группы студентов**, затем ОК.

### **3.2. Создать структуру таблицы Студенты:**

В окне Конструктора таблиц заполнить разделы Имя поля, Тип данных и Свойства поля в соответствии с таблицей 2.

Для поля КодГруппы выбрать тип **Мастер подстановок**. Это позволит облегчить заполнение данными этого поля, так как в таблице Студенты будут отображаться не коды группы, а их названия. После выбора типа Мастер подстановок откроется первое диалоговое окно Создание подстановки

В этом окне выбирается способ, которым столбец подстановки получит свои значения: из таблицы или запроса, затем щелчок по кнопке Далее. В следующем диалоговом окне выбирается таблица, содержащая столбец подстановки – Группа студентов, щелчок по кнопке Далее.

В следующем окне выбирается поле (Название), используемое в качестве столбца подстановки, и щелчком по кнопке переносится в окно Выбранные поля. Щелчок по кнопке Далее .

Следующее окно содержит сообщения о том, какие действия выполнить со столбцом в случае необходимости. Щелчок по кнопке Далее .

В следующем окне выполнить щелчок по кнопке Готово. Появится сообщение о том, что перед созданием связи необходимо сохранить таблицу. Для этого выполнить щелчок по кнопке Да.

В разделе Тип данных будет указан тип числовой, т.е. тип, соответствующий типу поля подстановки из таблицы Группа студентов. Для поля КодГруппы установить свойство - Индексированное поле. Для этого в разделе Свойство поля выбрать строку Индексированное поле. Выполнить щелчок по кнопке раскрытия списка и выбрать строку - Да (Допускаются совпадения).

После создания структуры таблицы необходимо задать ключевое поле - КодСтудента.

Сохранить структуры таблицы, введя имя таблицы, **Студенты**.

### **3.3. Создать структуру таблицы Дисциплины.**

В окне Конструктора таблиц заполнить разделы Имя поля, Тип данных и Свойства поля в соответствии с таблицей 3.

После создания структуры таблицы необходимо задать ключевое поле - КодДисциплины.

Сохранить структуры таблицы, введя имя таблицы, **Дисциплины**.

### **3.4. Создать структуру таблицы Успеваемость.**

В окне Конструктора таблиц заполнить разделы Имя поля, Тип данных и Свойства поля в соответствии с таблицей 4.

Для полей КодДисциплины и КодСтудента выбрать тип Мастер подстановок. Это позволит облегчить заполнение данными этих полей, так как в таблице Успеваемость будут отображаться не коды дисциплины и студента, а их названия

В таблице необходимо выбрать поля КодДисциплины и КодСтудента в качестве индексированных полей. Для свойства Индексированное поле установить значение Да (Допускаются совпадения).

После создания структуры таблицы необходимо задать ключевое поле – КодОценки (можно отказаться от создания ключевого поля)

**Сохранить структуру таблицы с именем Успеваемость.**

**4. Установить связи между таблицами:**

Выбрать команду Сервис — Схема данных или выбрать пиктограмму Схема данных на панели инструментов. Появится окно Схема данных, содержащее диалоговое окно Добавление таблицы, в котором отображается список таблиц. Выделить все таблицы: Группы студентов, Студенты, Дисциплины и Успеваемость и выполнить щелчок на кнопке Добавить. В окне Схема данных появятся таблицы, после этого необходимо закрыть окно Добавление таблицы.

Далее необходимо связать таблицы Группы студентов и Студенты, отбуксировав поле КодГруппы таблицы Группы студентов, на соответствующее поле таблицы Студенты. В появившемся диалогом окне Изменение связей необходимо активизировать флагки: Обеспечить целостность данных, каскадное обновление связанных полей и каскадное удаление связанных записей, убедится в том, что тип отношений один-ко-многим и нажать кнопку Создать. В окне Схема документа появится связь один-ко-многим между таблицами Группы студентов и Студенты.

Аналогично связать таблицы Студенты и Успеваемость, отбуксировав поле КодСтудентов таблицы Студенты, на соответствующее поле таблицы Успеваемость и установив те же флагки. Кроме того, необходимо связать таблицы Дисциплины и Успеваемость, используя КодДисциплины.

Закрыть окно Связи. При запросе на сохранение выполнить щелчок на кнопке Да.

**5. Заполнение таблиц**

Для заполнения таблиц Группы студентов и Студенты данные выбираются самостоятельно, а для таблиц Дисциплины и Успеваемость данные представлены в таблицах 5 и 6.

Заполнение таблиц целесообразно начинать с таблицы Группы студентов, так как поле Код группы таблицы Студенты используется в качестве столбца подстановки для заполнения соответствующего поля таблицы Студенты.

В окне Базы данных выбрать нужную таблицу, затем выполнить щелчок по кнопке Открыть.

На экране появится структура таблицы БД в режиме таблицы.

Заполнение производится по записям, т.е. вводится информация для всей строки целиком. При заполнении текущей строки появится новая пустая строка.

Переход к следующему полю осуществляется нажатием клавиши <Tab>. Для заполнения поля МЕМО в таблице Студенты нажать комбинацию клавиш <Shift+F2>, предварительно установив курсор в поле МЕМО.

Откроется текстовое окно «Область ввода». После ввода или редактирования данных в этом окне щелкнуть по кнопке ОК.

Для заполнения данными поля Код группы в таблице Студенты использовать список поля подстановки, раскрывая его щелчком мыши по кнопке раскрытия списка. Выбор названия группы производится щелчком мыши в соответствующей строке списка.

Таблица-объект Дисциплины Таблица 5

№ п/п	Название дисциплины	Количество часов
1	Иностранный язык	108

2	История Украины	108
3	Культурология	72
4	Математика	180
5	Политэкономия	108
6	Микроэкономика	180
7	Физкультура	36
8	Эконом. информатика	180

Данные для заполнения таблицы Успеваемость Таблица 6

№ п/п	Оценка Национальная /ECTS	Описание
1	5/A	Отлично - без ошибок
2	5/B	Отлично - с незначительными ошибками
3	4/B	Очень хорошо - с несколькими ошибками
4	4/C	Хорошо - с несколькими значительными ошибками
5	3/D	Удовлетворительно - со значительными недостатками
6	3/E	Удовлетворительно - удовлетворяет минимальному критерию оценки
7	2/FX	неудовлетворительно с возможностью повторной сдачи экзамена
8	2/F	неудовлетворительно с обязательным повторным курсом

## 6. Проверка обеспечения целостности данных (каскадное удаление).

В таблице Студенты удалить код 1 и убедиться в том, что из таблиц Группы студентов и Дисциплины удалены соответствующие записи. Восстановить удаленные данные во всех таблицах.

## 7. Формирование запросов

1) Создание запроса на выборку.

Задание: создать запрос «Успеваемость студентов» (отображать все фамилии студентов, которые получили оценки отл/А по дисциплинам), содержащий поля: Группы студентов, Фамилия, Имя, Отчество, Дисциплины, Оценка. Список должен быть отсортирован по фамилии по возрастанию. Указать условия отбора в поле оценка: отл/А Для этого необходимо выполнить следующую последовательность действий:

В окне базы данных необходимо выбрать вкладку Запросы и дважды щелкнуть на пиктограмме Создание запроса в режиме конструктора. Появится активное окно Добавление таблицы на фоне неактивного окна «Запрос 1: запрос на выборку». В окне Добавление таблицы следует выбрать таблицы (Группы студентов; Студенты; Успеваемость;

Дисциплины), на основе которых будет проводиться выбор данных, и щелкнуть на кнопке Добавить. После этого закрыть окно Добавление таблицы, окно «Запрос 1: запрос на выборку» станет активным.

Переместить имена полей с источника в Бланк запроса. Из таблицы Группы студентов отбуксировать поле Название в первое поле Бланка запросов, из таблицы Студенты отбуксировать поле Фамилии во второе поле Бланка запросов, а из таблицы Успеваемость отбуксировать поле Оценка в третье поле и из таблицы Дисциплины отбуксировать поле Название в четвертое поле Бланка запросов.

Задать принцип сортировки. Курсор мыши переместить в строку Сортировка для поля Фамилии, появится кнопка открытия списка режимов сортировки: по возрастанию и по убыванию. Установить в поле Фамилия режим сортировки – по возрастанию.

В строке Условия отбора необходимо ввести критерии поиска. В поле Оценка ввести - "отл/А", т.е. отображать все фамилии студентов, которые получили оценки отл/А.

После завершения формирования запроса закрыть окно Запрос на выборку. Откроется окно диалога Сохранить – ответить Да (ввести имя созданного запроса: Успеваемость студентов), и щелкнуть OK. Вернуться в окно базы данных. В окне базы данных при выбранной вкладке Запросы появится созданный запрос.

Для выполнения запроса: Выполнить щелчок мышью по запросу Успеваемость студентов, а затем, по кнопке Открыть. На экране появится, таблица, в которой, должны отображаться записи с фамилиями студентов, названиями групп, названиями дисциплин и полученных оценок, записи отсортированы по фамилии студентов по возрастанию.

Примечание: чтобы внести изменения в запрос необходимо: выбрать его щелчком мыши, выполнить щелчок по кнопке Конструктор, внести изменения. Сохранить запрос, повторить его выполнение.

## 2) Создать параметрический запрос

Задание: создать запрос, в результате которого будет выводиться Фамилия студента, Название группы, Дисциплина и Оценка (отл/А), полученная студентом по дисциплине.

Для этого необходимо выполнить следующую последовательность действий:

задать запрос в режиме конструктора или открыть существующий запрос: «Успеваемость студентов» в режиме конструктора;

в Бланк запроса в строке Условия отбора ввести условие отбора в виде приглашения в квадратных скобках, например [Введите фамилию];

закрыть окно Запрос на выборку, на вопрос о сохранении изменения ответить – Да. Вернуться в окно базы данных, где будет отображен созданный запрос;

выполнить запрос, щелкнув по кнопке: Открыть. В появившемся на экране окне диалога «Введите значение параметра» надо ввести фамилию студента, информацию об успеваемости которого необходимо получить, выполнить щелчок по кнопке OK;

на экране появится таблица с данными о выбранном студенте.

## Лабораторная работа №2 подключение к СУБД

Цель: составление запросов с использованием различных интерфейсов доступа к СУБД

Задачи:

1. создать БД аналогичную первой лабораторной;
2. составить запросы БД;
3. выполнить запросы через различные интерфейсы:
  - командная строка CLI mysql;
  - веб-интерфейс phpmyadmin;
  - десктопное приложение mysqlworkbench илиdbeaver;
4. отчет оформить, используя шаблон отчета.

Доступы к интерфейсам:

Командная строка

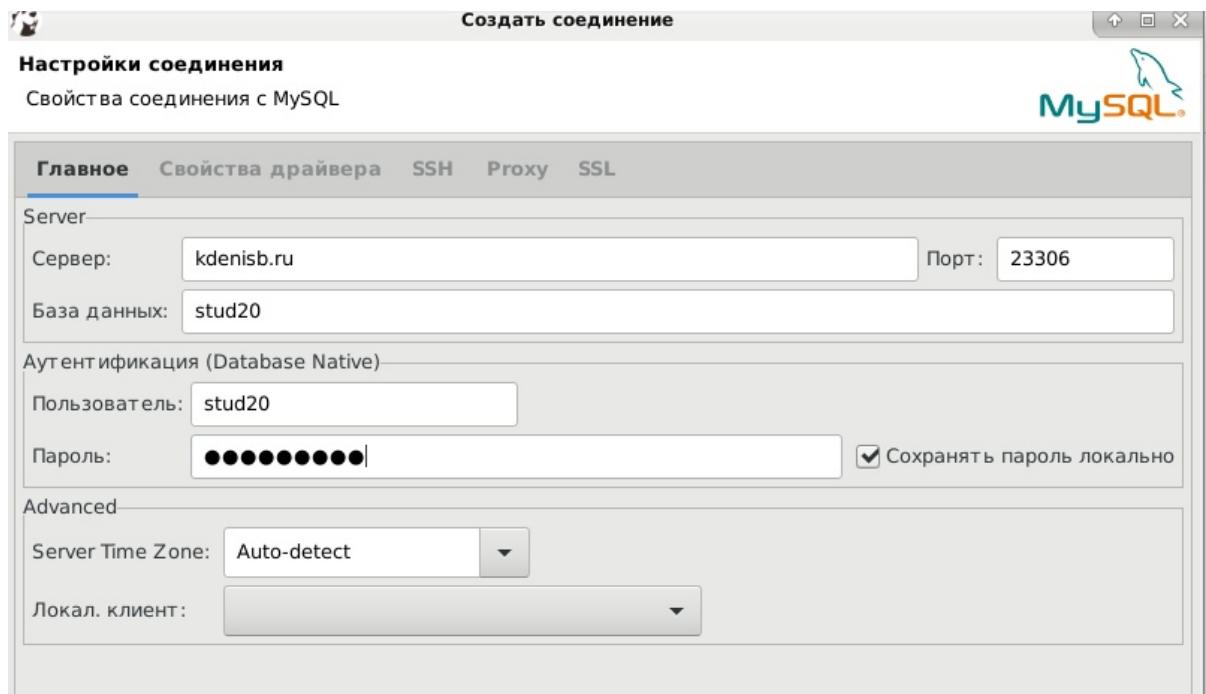
```
ssh -p 20022 stud20@kdenisb.ru
```

```
mysql --password=Zz8Gws4Vv -h 172.17.0.5 -u stud20 stud20
```

веб-интерфейс

<http://kdenisb.ru:20080/phpmyadmin/>

подключение черезdbeaver



# Лабораторная работа № 3 PHP - MySQL

Цель: формирование веб-страницы на основании результатов выполнения запросов

Задачи:

1. разработать программу на PHP в составе:
  - подключение к СУБД;
  - выполнение запроса (3 штуки);
  - получение и вывод результатов запроса;
2. разработать html-страницу для задания параметров PHP-программы;
3. разместить полученные файлы на хостинге;
4. отчет оформить, используя шаблон отчета.

Пример выполнения

Видеоурок

<https://youtu.be/XehyjjkAc6Q>

Исходные тексты

:::::::::::

t.html

:::::::::::

<body>

<form action="http://kdenisb.ru:20080/web/kdb/t.php" method="get">

Mark:

```
<select name="mark">
<option></option>
<option>5</option>
<option>4</option>
<option>3</option>
<option>2</option>
```

</select>

<button>send</button>

</form>

```
mysql_t.php

<body>
privet
<?php
$mysqli = new mysqli("kdenisb.ru:23306", "stud20", "Zz8Gws4Vv", "stud20");

if ($mysqli->connect_errno) {
    printf("err con: %s\n", $mysqli->connect_error);
    exit();
}

if ($result = $mysqli->query("SELECT f, m from fm where m=".$_GET[mark].";")) {
    printf("SelectED strings %d .\n", $result->num_rows);
}
?>
<table border=1>
<?php
while( $row = $result->fetch_array(MYSQLI_ASSOC) )
{
    printf ("<tr><td>%s</td><td>%s</td></tr>", $row["f"], $row["m"]);
}
?>
</table>
<?php

$result->close();

$mysqli->close();
```

```
?>
::::::::::
t.php
::::::::::
<body>
privet
<?php

$bdd=new PDO('mysql:host=kdenisb.ru;port=23306;dbname=stud20',
              "stud20", "Zz8Gws4Vv");

?>
<table border=1>
<?php
foreach($bdd->query("SELECT f, m from fm where m=".$_GET[mark].";") as $row) {
printf ("<tr><td>%s</td><td>%s</td></tr>", $row["f"], $row["m"]);
}

?>
</table>
<?php

?>
```

# **Лабораторная работа № 4**

## **Разработка CRUD-приложения**

Необходимо разработать программу на php для просмотра, создания, удаления и редактирования (CRUD) данных из БД.

Варианты заданий выбрать самостоятельно: либо продолжать выполнять начальное задание (по согласованию с преподавателем), либо из документа «Курсовая работа по БД» (уровень сложности «1»).

### **Порядок работы\***

1. Разработать экранные формы для приложения,
2. Реализовать функцию просмотра данных из массива,
3. Разработать или доработать таблицы в БД,
4. Реализовать функцию просмотра данных из БД,
5. Реализовать функции создания, удаления, редактирования.

\* «Порядок работы» носит рекомендательный характер.

### **Рекомендации**

Разработку начать с веб-страницы, а не с БД, потом постепенно добавить формирование страницы из массива, и только после этого подключать БД. Запросы к БД сначала отрабатывать в средствах работы с БД, например, «MySQL Workbench».

ВидеоДокумент: <https://youtu.be/r2fW3WooUJg>

### **Адреса ресурсов**

- Площадка хостинга - протокол:ssh хост:kdenisb.ru порт:20022  
имя:stud20 пароль:Zz8Gws4Vv
- Адрес программы на хостинге -  
<http://kdenisb.ru:20080/web/<СвояДиректория>/>

### **Пример выполнения**

```
CREATE TABLE `client` (
  `idclient` INT NOT NULL,
  `ima` VARCHAR(45) NOT NULL,
  `plat` INT NULL,
  `pol` CHAR(2) NULL,
  PRIMARY KEY (`idclient`));
```

```
CREATE TABLE `poss` (
  `idposs` int(11) NOT NULL AUTO_INCREMENT,
  `day` int(11) DEFAULT NULL,
```

```

`idclient` int(11) DEFAULT NULL,
`idvremya` int(11) DEFAULT NULL,
PRIMARY KEY (`idposs`)
)

CREATE TABLE `vremya` (
`idvremya` int(11) NOT NULL,
`dat` char(8) NOT NULL,
PRIMARY KEY (`idvremya`)
)

:::::::::::
index.php
:::::::::::
<!DOCTYPE html>
<html>
<head>
    <title>p read</title>
    <style type="text/css">
        .pos {border: solid black 1px;}
        .del, .edit {float: right; margin: 10px;}
    </style>
</head>
<body>
<?php
$mysqli = new mysqli("kdenisb.ru:23306", "stud20", "Zz8Gws4Vv", "stud20");

$result = $mysqli->query("SELECT idposs, ima, dat FROM poss, client, vremya
where poss.idclient=client.idclient and poss.idvremya=vremya.idvremya and day=".
$_GET[day]);

while( $row = $result->fetch_array(MYSQLI_ASSOC) )
{
?>
    <div class="pos">
        <div class="del">
            <a href="del.php?idposs=<?=$row["idposs"]?>&day=<?=$_GET
[day]?>">
                удалить</a>
        </div>
        <div class="edit">
            <a href="edit.php?idposs=<?=$row["idposs"]?>&day=<?=$_GE
T[day]?>">
                редакт</a>
        </div>
        <div class="ima"><?=$row["ima"]?></div>
        <div class="dat"><?=$row["dat"]?></div>

```

```
</div>
<?php
}
?>
<div class="pos">
    <form action="create.php" method="get">
        <input type="hidden" name="day" value="<?=$_GET[day]?>">
        <select name="client">
            <option></option>
            <?php
$result = $mysqli->query("SELECT idclient, ima FROM client;");

while( $row = $result->fetch_array(MYSQLI_ASSOC) )
{
?>
            <option value="<?=$row["idclient"]?>"><?=$row["ima"]?></option>
<?php
}
?>
        </select>
        <select name="vremya">
            <option></option>
            <?php
$result = $mysqli->query("SELECT idvremya, dat FROM vremya;");

while( $row = $result->fetch_array(MYSQLI_ASSOC) )
{
?>
            <option value="<?=$row["idvremya"]?>"><?=$row["dat"]?></option>
<?php
}
?>
        </select>
        <button>Добавить посещение</button>
    </form>
</div>

</body>
</html>
:::::::::::
create.php
:::::::::::
<!DOCTYPE html>
<html>
<head>
    <title>p create</title>
</head>
<body>
<?php
$mysqli = new mysqli("kdenisb.ru:23306", "stud20", "Zz8Gws4Vv", "stud20");
$result = $mysqli->query("insert into poss (day, idclient, idvremya) values ("
```

```

".$_GET[day].".$_GET[client].".$_GET[vremya]."");
?>
<a href=".?day=<?=$_GET[day]?>">Вернуться</a>
</body>
</html>
::::::::::
del.php
::::::::::
<!DOCTYPE html>
<html>
<head>
    <title>p delete</title>
</head>
<body>
<?php
$mysqli = new mysqli("kdenisb.ru:23306", "stud20", "Zz8Gws4Vv", "stud20");
$result = $mysqli->query("delete from poss where idposs=".$_GET[idposs]);
?>
<a href=".?day=<?=$_GET[day]?>">Вернуться</a>
</body>
</html>
::::::::::
edit.php
::::::::::
<!DOCTYPE html>
<html>
<head>
    <title>p edit</title>
</head>
<body>
<?php
$mysqli = new mysqli("kdenisb.ru:23306", "stud20", "Zz8Gws4Vv", "stud20");
$result = $mysqli->query("SELECT idclient, idvremya FROM poss where idposs=".$_GET[idposs]);
$row = $result->fetch_array(MYSQLI_ASSOC);
$idclient=$row[idclient];
$idvremya=$row[idvremya];

?>
<form action="update.php" method="get">
    <input type="hidden" name="day" value="<?=$_GET[day]?>">
    <input type="hidden" name="pos" value="<?=$_GET[idposs]?>">
    <select name="client">
        <option></option>
        <?php
$result = $mysqli->query("SELECT idclient, ima FROM client;");

while( $row = $result->fetch_array(MYSQLI_ASSOC) )
{
?>
    <option value="<?=$row["idclient"]?>">

```

```

        <?=( $row[ "idclient" ] == $idclient ? "selected": "" )?>>
        <?= $row[ "ima" ]?>
    </option>
<?php
}
?
</select>
<select name="vremya">
    <option></option>
    <?php
$result = $mysqli->query("SELECT idvremya, dat FROM vremya;");

while( $row = $result->fetch_array(MYSQLI_ASSOC) )
{
?
<option value=<?= $row[ "idvremya" ]?>">
    <?=( $row[ "idvremya" ] == $idvremya ? "selected": "" )?>>
    <?= $row[ "dat" ]?>
</option>
<?php
}
?
</select>
<button>Сохранить</button>
</form>
</body>
</html>
::::::::::
update.php
::::::::::
<!DOCTYPE html>
<html>
<head>
    <title>p update</title>
</head>
<body>
<?php
$mysqli = new mysqli("kdenisb.ru:23306", "stud20", "Zz8Gws4Vv", "stud20");

$s="update poss set day=".$_GET[day].", idclient=".$_GET[client].", idvremya=".$_GET[vremya]." where idposs=".$_GET[pos];

$result = $mysqli->query("update poss set day=".$_GET[day].", idclient=".$_GET[client].", idvremya=".$_GET[vremya]." where idposs=".$_GET[pos]);
?>
<a href=". /?day=<?= $_GET[day]?>">Вернуться</a>
</body>
</html>

```

# **Лабораторная работа № 5**

## **Хранение больших двоичных объектов (blob)**

Необходимо разработать программу на php для сохранения и просмотра файлов (blob-объектов) в БД.

### **Порядок работы\***

1. Разработать форму загрузки файла,
2. Реализовать программу сохранения файла в БД,
3. Разработать программу отображения файла.

### **Варианты**

Вариант	Тип файла
1	jpg
2	png
3	gif
4	mp3
5	mp4
6	ogg
7	ogv
8	jpg
9	png
10	gif
11	mp3
12	mp4
13	ogg
14	ogv
15	jpg
16	png
17	gif
18	mp3
19	mp4
20	ogg
21	ogv

## Рекомендации

Видеоурок: <https://youtu.be/AS7rJcJyNf8>

Адреса ресурсов

- Площадка хостинга - протокол:ssh хост:kdenisb.ru порт:20022  
имя:stud20 пароль:Zz8Gws4Vv
- Адрес программы нахостинге -  
<http://kdenisb.ru:20080/web/<СвояДиректория>/>

## Пример выполнения

```
:::::::::::  
fbot.html  
:::::::::::  
<!DOCTYPE html>  
<html>  
<head>  
    <title></title>  
</head>  
<body>  
    <form action="fbot.php" method="post" enctype="multipart/form-data">  
        <input type="file" name="imgf">  
        <button>send</button>  
    </form>  
  
</body>  
</html>  
:::::::::::  
fbot.php  
:::::::::::  
<?php  
  
$mysqli = new mysqli("kdenisb.ru:23306", "stud20", "Zz8Gws4Vv", "stud20");  
var_dump($mysqli);  
  
$stmt = $mysqli->prepare("INSERT INTO fbob (fname,fdata) VALUES(?,?)");  
$null = NULL;  
$stmt->bind_param("sb", $_FILES['imgf']['name'], $null);  
$stmt->send_long_data(1, file_get_contents($_FILES['imgf']['tmp_name']));  
$stmt->execute();  
$stmt->close();  
?>  
:::::::::::  
fbotr.php  
:::::::::::  
<?php  
  
$mysqli = new mysqli("kdenisb.ru:23306", "stud20", "Zz8Gws4Vv", "stud20");
```

```
//var_dump($mysqli);  
  
$stmt = $mysqli->prepare("SELECT fdata FROM fbob WHERE id=?");  
//var_dump($stmt);  
$id=2;  
  
$stmt->bind_param("i", $id);  
$stmt->execute();  
$stmt->store_result();  
$stmt->bind_result($image);  
$stmt->fetch();  
  
header("Content-Type: image/png");  
  
echo $image;  
  
?>
```

# **Лабораторная работа № 6**

## **NoSQL СУБД: MongoDB**

Необходимо научиться использовать документо-ориентированную СУБД MongoDB.

### **Порядок работы\***

1. Создать БД,
2. Создать коллекцию аналогичную таблице из первых лабораторных,
3. Сделать запросы аналогичные первым лабораторным

### **Рекомендации**

Видеоурок: [https://youtu.be/R\\_r9mC-\\_nKQ](https://youtu.be/R_r9mC-_nKQ)

Адреса ресурсов

- Площадка хостинга - протокол:ssh хост:kdenisb.ru порт:20022  
имя:stud20 пароль:Zz8Gws4Vv
- Учебник: <https://metanit.com/nosql/mongodb/>

### **Пример выполнения**

Подключение к СУБД

```
$ mongo
```

Создание и подключение к БД

```
> use test-kdb
switched to db test-kdb
```

Создание записей

```
> db.t1.save({name:"aaa", price:25})
WriteResult({ "nInserted" : 1 })
> db.t1.save({name:"bbbw", price:14})
WriteResult({ "nInserted" : 1 })
```

Запросы

```
> db.t1.find()
{ "_id" : ObjectId("5fd349f852cf4c9f13cb7e3"), "name" : "aaa", "price" : 25 }
{ "_id" : ObjectId("5fd34a0952cf4c9f13cb7e4"), "name" : "bbbw", "price" : 14 }
> db.t1.find({name:"aaa"})
{ "_id" : ObjectId("5fd349f852cf4c9f13cb7e3"), "name" : "aaa", "price" : 25 }
> db.t1.find("this.price>15")
{ "_id" : ObjectId("5fd349f852cf4c9f13cb7e3"), "name" : "aaa", "price" : 25 }
Отображать только поле «name»
> db.t1.find("this.price>15", {name:1})
{ "_id" : ObjectId("5fd349f852cf4c9f13cb7e3"), "name" : "aaa" }
```

# **Лабораторная работа № 7**

## **NoSQL СУБД: xml и json в MySQL**

Необходимо научиться использовать базы данных со сложными (nosql) полями.

### **Порядок работы\***

1. Создать БД,
2. Продумать структуру «сложного» поля аналогично первым лабораторным,
3. Создать таблицы для хранения и xml, и json содержимого,
4. Сделать запросы аналогичные первым лабораторным

### **Рекомендации**

Видеоурок: <https://youtu.be/OULPpDAwcNY>

#### **Адреса ресурсов**

- Площадка хостинга - протокол:ssh хост:kdenisb.ru порт:20022  
имя:stud20 пароль:Zz8Gws4Vv
- Доступ к СУБД:
  - password Zz8Gws4Vv
  - user stud20
  - host kdenisb.ru
  - port 23306
  - db stud20
- Учебные материалы:
  - <https://support.smartbear.com/alertsite/docs/monitors/api/endpoint/jsonpath.html>
  - <https://dev.mysql.com/doc/refman/8.0/en/xml-functions.html>
  - <https://adw0rd.com/2009/4/13/xpath-mysql/>

### **Пример выполнения**

	XML	JSON
Структура данных nosql-поля	<pre>&lt;diploms&gt; &lt;item&gt; &lt;org&gt;school 99&lt;/org&gt; &lt;midmark&gt;3&lt;/midmark&gt; &lt;/item&gt; &lt;item&gt; &lt;org&gt;first technar&lt;/org&gt; &lt;spec&gt;koder&lt;/spec&gt; &lt;midmark&gt;4&lt;/midmark&gt; &lt;/item&gt;</pre>	<pre>[{"org":"school 99", "midmark":3}, {"org":"first technar", "spec":"koder", "midmark":4}]</pre>

	</diploms>	
Создание таблицы	<pre>CREATE TABLE `stud20`.`stud_edux` (   `id` INT NOT NULL,   `type` INT NULL COMMENT '1 - среднее\n2 - среднее специальное\n3 - высшее',   `diplom` TEXT NULL,   PRIMARY KEY (`id`));</pre>	<pre>CREATE TABLE `stud20`.`stud_eduj` (   `id` INT NOT NULL,   `type` INT NULL COMMENT '1 - среднее\n2 - среднее специальное\n3 - высшее',   `diplom` JSON NULL,   PRIMARY KEY (`id`));</pre>
Наполнение таблицы	<pre>INSERT INTO stud_edux VALUES ('1', '2', '&lt;diploms&gt;&lt;item&gt;&lt;org&gt;school 99&lt;/org&gt;&lt;midmark&gt;3&lt;/midmark&gt;&lt;/item&gt;&lt;item&gt;&lt;org&gt;first technar&lt;/org&gt;&lt;spec&gt;koder&lt;/spec&gt;&lt;midmark&gt;4&lt;/midmark&gt;&lt;/item&gt;&lt;/diploms&gt;'); INSERT INTO stud_edux VALUES ('2', '1', '&lt;diploms&gt;&lt;item&gt;&lt;org&gt;school 42&lt;/org&gt;&lt;midmark&gt;4&lt;/midmark&gt;&lt;/item&gt;&lt;/diploms&gt;');</pre>	<pre>INSERT INTO stud_eduj VALUES ('1', '2', '[{"org":"school 99","midmark":3}, {"org":"first technar", "spec":"koder", "midmark":4}]'); INSERT INTO stud_eduj VALUES ('2', '1', '[{"org":"school 42","midmark":5}]');</pre>
Запросы	select * from stud_edux;	select * from stud_eduj;
	select id, ExtractValue(diplom, '/diploms/item[1]/midmark') from stud_edux;	select id, JSON_EXTRACT(diplom, '\$[0].midmark') from stud_eduj;
	select id, ExtractValue(diplom, '//midmark') from stud_edux;	select id, JSON_EXTRACT(diplom, '\$[*].midmark') from stud_eduj;
	select id, ExtractValue(diplom, '//spec') from stud_edux;	select id, JSON_EXTRACT(diplom, '\$[*].spec') from stud_eduj;
	select id from stud_edux where ExtractValue(diplom, '//spec') = 'koder';	select id from stud_eduj where JSON_EXTRACT(diplom, '\$[*].spec') like '%koder%';
		select id from stud_eduj where JSON_EXTRACT(diplom, '\$[*].spec') is not null;