ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«ЛЕНИНГРАДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ИМЕНИ А.С. ПУШКИНА»**

УТВЕРЖДАЮ

Проректор по учебно-методической

работе

\_\_\_\_\_\_\_\_\_\_\_\_ С.Н.Большаков

РАБОЧАЯ ПРОГРАММА

дисциплины

**Б1.В.03.ДВ.01.02 ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИЙ WWW ДЛЯ ДОСТУПА К БАЗАМ ДАННЫХ**

Направление подготовки **44.03.05 Педагогическое образование
(с двумя профилями подготовки)**

Направленность (профиль) **Информатика и математика**

(год начала подготовки - 2022)

Санкт-Петербург

2022

**1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ:**

Процесс изучения дисциплины направлен на формирование следующих компетенций:

|  |  |  |
| --- | --- | --- |
| Индекс компетенции | Содержание компетенции (или ее части) | Индикаторы компетенций (код и содержание) |
| ПК-3 | Способен применять предметные знания при реализации образовательного процесса | ИПК-3.1. Знает основные классы баз данных (БД) и области их применения, принципы функционирования БД; основы технологии «клиент-сервер»; инструменты разработки удаленных БД; модели организации доступа к удаленным БД; терминологию и базовые понятия теории БД; основные классы задач, решаемые с использованием БД; особенности организации удаленных БД.ИПК-3.2. Умеет грамотно и эффективно использовать готовые удаленные БД; систематизировать информацию и представлять её в виде реляционной БД; реализовывать удаленные БД средствами сетевых реляционных СУБД; разрабатывать web-приложения БД.ИПК-3.3. Владеет навыками использования удаленных БД для решения прикладных задач; навыками грамотного и эффективного использования удаленных БД; навыками проектирования и разработки удаленных БД и web-приложений БД. |

**2. Место дисциплины в структуре ОП:**

Цель дисциплины: знакомство обучающихся с архитектурой и принципами работы удаленных баз данных, основами организации доступа к базам данных через web-интерфейс, формирование представления о методах и средствах разработки удаленных баз данных и web-приложений баз данных, получение навыков самостоятельной разработки web-приложений баз данных.

Задачи дисциплины:

* знакомство обучающихся с историей развития баз данных;
* знакомство обучающихся с основными понятиями теории баз данных (база данных, банк данных, система управления базой данных, приложение базы данных и т.д.);
* формирование представления об основных классах задач, решаемых с использованием баз данных;
* знакомство обучающихся с основными типами баз данных и моделями данных;
* формирование представления о моделях организации доступа к базам данных;
* знакомство обучающихся с архитектурой и принципами работы удаленных баз данных;
* знакомство обучающихся с клиент-серверными технологиями Web, языками разработки клиентских и серверных web-приложений;
* формирование представления о методах и средствах разработки удаленных баз данных и web-приложений баз данных;
* получение навыков самостоятельной разработки серверных web-приложений баз данных.

Дисциплина относится к дисциплинам по выбору части, формируемой участниками образовательных отношений, модуль Средства обработки информационных структур. Полученные в результате изучения дисциплины навыки позволят бакалавру образования создавать эффективные удаленные базы данных, организовывать к ним доступ через web-интерфейс и использовать их при создании программ учебного назначения, образовательных ресурсов и информационных образовательных сред.

Освоение дисциплины и сформированные при этом компетенции необходимы в последующей деятельности.

**3. Объем дисциплины и виды учебной работы:**

Общая трудоемкость освоения дисциплины составляет 6 зачетных единиц, 216 академических часов *(1 зачетная единица соответствует 36 академическим часам).*

Очная форма обучения

|  |  |
| --- | --- |
| Вид учебной работы | Трудоемкость в акад.час |
|  |  | Практическая подготовка |
| **Контактная работа (аудиторные занятия) (всего):** | 108 |
| в том числе: |  |
| Лекции | 30 | - |
| Лабораторные работы / Практические занятия (в т.ч. зачет) | 78/- | -/- |
| **Самостоятельная работа (всего)** | 81 |
| **Вид промежуточной аттестации (экзамен):** | 27 |
| контактная работа | 2,35 |
| самостоятельная работа по подготовке к экзамену | 6,65 |
| **Общая трудоемкость дисциплины (в час. /з.е.)** | 216/6 |

Заочная форма обучения

|  |  |
| --- | --- |
| Вид учебной работы | Трудоемкость в акад.час |
|  |  | Практическая подготовка |
| **Контактная работа (аудиторные занятия) (всего):** | 24 |
| в том числе: |  |
| Лекции | 8 | - |
| Лабораторные работы/ Практические занятия | 16/- | -/- |
| **Самостоятельная работа (всего)** | 179 |
| **Вид промежуточной аттестации (зачет):** | 4 |
| контактная работа | 0,25 |
| самостоятельная работа по подготовке к зачету | 3,75 |
| **Вид промежуточной аттестации (экзамен):** | 9 |
| контактная работа | 2,35 |
| самостоятельная работа по подготовке к экзамену | 6,65 |
| **Общая трудоемкость дисциплины (в час. /з.е.)** | 216/6 |

**4.СОДЕРЖАНИЕ ДИСЦИПЛИНЫ:**

При проведении учебных занятий обеспечивается развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств (включая при необходимости проведение интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализ ситуаций и имитационных моделей, составленных на основе результатов научных исследований, проводимых организацией, в том числе с учетом региональных особенностей профессиональной деятельности выпускников и потребностей работодателей**).**

**4.1 Блоки (разделы) дисциплины.**

|  |  |
| --- | --- |
| № | Наименование блока (раздела) дисциплины |
| 1 | Основные понятия теории БД |
| 2 | Основы работы с СУБД MySQL |
| 3 | Разработка клиентских web-сценариев  |
| 4 | Создание серверных приложений |
| 5 | Основы создания серверных web-приложений БД |
| 6 | Проектирование БД  |
| 7 | Разработка серверных сценариев для обработки строк и массивов |
| 8 | Разработка клиент-серверного приложения для просмотра и редактирования БД |
| 9 | Проектирование БД |

**4.2. Примерная тематика курсовых работ (проектов):**

Курсовая работа по дисциплине не предусмотрена учебным планом.

**4.3. Перечень занятий, проводимых в активной и интерактивной формах, обеспечивающих развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств. Практическая подготовка\*.**

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Наименование блока (раздела) дисциплины** | **Занятия, проводимые в активной и интерактивной формах** | **Практическая подготовка\*** |
| **Форма проведения занятия** | **Наименование видов занятий** |
| 1. | Основные понятия теории БД | лекционное занятие | компьютерные презентации |  |
| 2. | Основы работы с СУБД MySQL | лекционное занятие | компьютерные презентации |  |
| 3. | Разработка клиентских web-сценариев  | лекционное занятие | компьютерные презентации |  |
| 4. | Создание серверных приложений | лекционное занятие | компьютерные презентации |  |
| 5. | Основы создания серверных web-приложений БД | лекционное занятие | компьютерные презентации |  |
| 6. | Проектирование БД  | лекционное занятие | компьютерные презентации |  |
| 7. | Разработка серверных сценариев для обработки строк и массивов | лабораторное занятие | работа в команде |  |
| 8. | Разработка клиент-серверного приложения для просмотра и редактирования БД | лабораторное занятие | работа в команде |  |
| 9. | Проектирование БД | лабораторное занятие | работа в команде  |  |

**\***Практическая подготовка при реализации учебных предметов, курсов, дисциплин (модулей) организуется путем проведения практических занятий, практикумов, лабораторных работ и иных аналогичных видов учебной деятельности, **предусматривающих участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.**

**5. Учебно-методическое обеспечение для самостоятельной работы обучающихся по дисциплине:**

**5.1. Вопросы для подготовки к лабораторным занятиям:**

**Создание и модификация однотабличных БД средствами MySQL**

1. Какие сведения хранятся в служебной базе данных MySQL?
2. Для чего нужна служебная БД?
3. Каким образом можно просмотреть список существующих БД?
4. Что такое текущая БД? Каким образом можно сменить текущую БД?
5. Что необходимо учитывать при выборе типа поля?
6. Что такое первичный ключ? Для каких целей он используется?
7. Приведите общий вид оператора, позволяющего создать новую таблицу.
8. Приведите общий вид оператора, позволяющего внести в таблицу данные.
9. Каким образом можно просмотреть структуру таблицы?
10. Какие команды модификации структуры таблицы гарантированно могут быть выполнены без ошибок, даже если в таблицу уже занесены данные?

**Учетные записи пользователей**

1. Каким образом в СУБД MySQL хранятся учетные записи пользователей?
2. Каким образом можно просмотреть существующие учетные записи пользователей?
3. Какая команда позволяет создать новую учетную запись?
4. Какие привилегии могут быть назначены пользователю?
5. Каков минимальный объект, к которому пользователю могут быть назначены права доступа?
6. Можно ли изменить назначенные ранее привилегии? Если да, то каким образом?
7. Каким образом СУБД MySQL будет реагировать на попытки выполнения действий, не предусмотренных текущими привилегиями?

**Извлечение информации из БД**

1. Какие объекты могут быть указаны в списке полей оператора SELECT?
2. Что такое псевдоним? В каких случаях его целесообразно использовать?
3. Каким образом производится вложенная сортировка. Приведите примеры.
4. Каким образом из результатов запроса можно исключить повторения?
5. Перечислите все известные вам способы задания следующего условия отбора: значение целочисленного поля должно принадлежать определенному диапазону.
6. Каким образом для текстовых полей можно задать условие отбора с помощью шаблона?
7. Перечислите правила использования в запросах агрегирующих функций. Приведите примеры.

**Манипулирование данными**

1. Какова структура запроса на удаление на языке SQL?
2. Почему в запросе на удаление не указываются названия полей?
3. Какова структура запроса на модификацию данных на языке SQL?
4. Можно ли с помощью одного запроса на модификацию данных изменить значения сразу нескольких полей указанных записей?
5. Какова структура запроса на добавление данных на языке SQL?
6. Каким образом можно заполнить необязательные поля, которые не получили значения при создании записи?
7. Что произойдет, если в запросе на добавление записи некоторые из обязательных полей не будут определены?

**Создание многотабличных БД средствами MySQL**

1. Что такое внешний ключ? Для каких целей он используется?
2. Перечислите условия, накладываемые на поля для связи таблиц.
3. Какие виды связей вы знаете? Приведите примеры, иллюстрирующие каждый из видов связей.
4. Какие параметры связей вы знаете? Приведите примеры случаев, в которых целесообразно использовать каждый из параметров.
5. Опишите порядок заполнения таблиц многотабличной БД.
6. Какие варианты источника могут быть указаны в разделе FROM оператора SELECT при работе с многотабличной БД?
7. Какие варианты источника могут быть указаны в разделе FROM оператора DELETE при работе с многотабличной БД?

**Создание запросов на выборку к многотабличным базам данных**

1. Что такое объединение? В какой момент оно создается?
2. Какие виды объединений вы знаете? Приведите примеры.
3. Влияют ли заданные пользователем в запросе параметры объединения на результат запроса? Ответ обоснуйте.
4. Каким образом следует записывать имена полей при формировании запроса к многотабличной БД на языке SQL?
5. Каковы правила использования в запросах агрегирующих функций?
6. Что такое группировка? В каких случаях её целесообразно использовать?
7. В чем отличие разделов WHERE и HAVING?

**Служебные утилиты для работы в Интернет. Изучение протокола HTTP**

1. Что такое IP-адрес? Какие форматы IP-адреса вы знаете?
2. Какие части можно выделить в IP-адресе? Каким образом это можно сделать?
3. Каким образом можно определить IP-адреса, назначенные сетевым интерфейсам компьютера?
4. Каким образом можно проверить достижимость некоторого узла сети?
5. Каким образом можно проследить маршрут прохождения пакетов к определенному узлу сети?
6. Опишите структуру HTTP-запроса клиента.
7. Опишите структуру HTTP-ответа сервера.

**Создание простейших сценариев средствами JavaScript**

1. Какова структура сценария JavaScript?
2. Перечислите способы подключения сценария.
3. Каким образом в сценарии можно создать и использовать переменные?
4. Перечислите основные типы данных JavaScript
5. Какие управляющие операторы JavaScript вы знаете? Приведите примеры, демонстрирующие целесообразность использования оператора switch.
6. Чем отличается цикл while от цикла do… while? В каких случаях целесообразно использовать каждый из них? Приведите примеры.
7. В каких случаях в сценариях целесообразно использовать массивы? Приведите конкретные примеры.

**Обработка событий в JavaScript**

1. Что такое событие? Приведите примеры событий.
2. Что такое активирующее действие? Приведите примеры событий, имеющих разные активирующие действия.
3. Что такое обработчик события?
4. Перечислите известные вам способы назначения обработчиков.
5. Что такое объект «событие»? Какую информацию из него мы можем получить?
6. Что такое всплывающие события? В каком случае возникает всплытие? Как его можно прервать?
7. В каких случаях можно изменить действие браузера по умолчанию?

**Использование регулярных выражений для электронной обработки текстов**

1. Что такое регулярное выражение? Приведите примеры регулярных выражений.
2. Какими способами можно определить регулярное выражение?
3. Что такое управляющая последовательность? Приведите примеры управляющих последовательностей.
4. Каким образом в регулярных выражениях задается количество повторений того или иного элемента?
5. Что такое альтернатива в регулярном выражении? Каким образом можно задать альтернативу? Приведите пример задачи, для решения которой необходимо использовать регулярное выражение с альтернативами.
6. Что такое группировка в регулярном выражении? Каким образом задается группировка? В каких случаях целесообразно использовать группировку? Приведите примеры.
7. Какие методы класса String используются для поиска по шаблону? Приведите примеры задач, для решения которых целесообразно использовать каждый из методов.

**Основы разработки сценариев на языке PHP**

1. Какое программное обеспечение необходимо иметь для того чтобы проверить php-сценарий?
2. Что генерируется при выполнении php-сценария?
3. Можно ли в php-сценарии реализовать проверку отдельного поля заполняемой пользователем формы на соответствие определенным критериям? Ответ обоснуйте.
4. Перечислите основные типы данных языка php.
5. Какие управляющие операторы php вы знаете?
6. Каким образом оборвать выполнение цикла? Приведите пример задачи, для решения которой целесообразно использовать цикл, обрывающийся при выполнении некоторого условия.
7. Опишите технологию создания собственной функции.

**Разработка серверных сценариев для обработки строк и массивов**

1. Каким образом производится сравнение строк?
2. Перечислите основные возможности форматирования строк.
3. Приведите классификацию задач по обработке строк. Приведите примеры задач каждого класса.
4. В каких случаях в сценарии целесообразно использовать массивы?
5. Каким образом можно описать массив и как его можно инициализировать?
6. Какие возможности сортировки массивов средствами стандартных функций вы знаете?
7. Каким образом содержимое массива можно легко сохранить в файл?

**Разработка серверных сценариев для работы с файлами и каталогами**

1. Какие режимы работы с файлами допускаются в php? Каким образом необходимо определить режим работы с открываемым файлом?
2. Какие стандартные функции позволяют определять такие свойства файла как размер, дата создания, дата последнего изменения и дата обращения к файлу?
3. Каким образом производится чтение информации из файла?
4. Каким образом можно записать информацию в файл?
5. Каким образом можно получить содержимое файла в виде массива? В каких случаях это целесообразно?
6. Какие операции с каталогами реализованы средствами стандартных функций php?
7. Каким образом можно избежать ошибки при попытке открыть несуществующий файл?

**Написание гостевой книги, счетчика посещений, голосований**

1. Какие параметры обычно задают перед созданием гостевой книги?
2. Почему для страницы, на которой отображаются записи гостевой книги, устанавливают запрет на кеширование? Каким образом это можно сделать?
3. Каким образом введенные пользователем специальные символы можно конвертировать в мнемоники HTML?
4. Каким образом обеспечивается вывод определенного количества записей на одной странице?
5. Опишите алгоритм работы счетчика посещений, если информация о количестве посещений хранится в текстовом файле.
6. Какие действия необходимо реализовать в сценарии для учета голоса?
7. Каким образом можно исключить повторное голосование?

**Разработка клиент-серверного приложения для просмотра БД**

1. Какие параметры необходимо задать для подключения к БД?
2. Чем обычное соединение с сервером отличается от постоянного? В каких случаях целесообразно использовать каждый из этих видов соединений?
3. От чего зависит выбор функции для подключения к БД?
4. Каким образом можно передать БД запрос на выборку?
5. Что происходит в результате выполнения запроса на выборку, переданного БД из php-сценария?
6. Каким образом можно сохранить в файле результаты выполнения запроса на выборку?
7. Каким образом можно вывести на страницу результаты выполнения запроса на выборку?

**Разработка клиент-серверного приложения для просмотра и редактирования БД**

1. Каким образом можно передать БД запрос на удаление данных?
2. Каким образом можно обновить значение некоторого поля для всех записей БД?
3. Каким образом можно отредактировать значение поля определенной записи?
4. Каким образом можно создать форму, позволяющую пользователю в удобном виде ввести данные о новом объекте?
5. Каким образом можно организовать добавление введенных пользователем данных в БД?
6. Как организовать запрет на добавление введенных пользователем данных в случае, когда пользователь не задал значение обязательного поля?
7. Каким образом можно получить расширенную информацию об ошибках, возникших при выполнении запросов?

**Система разделения прав доступа в веб-приложении**

1. Что собой представляет дискреционная модель доступа? Какие её основные достоинства и недостатки?
2. Что такое мандатная модель управления доступом? Какие её основные достоинства и недостатки?
3. Что такое ролевая модель управления доступом? Какие её основные достоинства и недостатки?
4. Что такое аутентификация? Перечислите основные компоненты системы аутентификации.
5. Назовите классы аутентифицирующей информации. Каковы достоинства и недостатки каждого класса?
6. Каковы пути повышения надежности парольной защиты?
7. Опишите алгоритм проверки возможности выполнения некоторым пользователем некоторого действия над определенным объектом.

**Проектирование БД методом нормальных форм**

1. Что такое аномалия? Приведите примеры аномалий обновления.
2. Перечислите ограничения первой нормальной формы.
3. Перечислите ограничения второй нормальной формы.
4. Опишите алгоритм приведения отношений, находящихся в первой нормальной форме, ко второй нормальной форме.
5. Что такое транзитивная зависимость? Приведите примеры.
6. Перечислите ограничения третьей нормальной формы.
7. Опишите алгоритм приведения отношений, находящихся во второй нормальной форме, к третьей нормальной форме.

**Проектирование БД методом ER-диаграмм**

1. Раскройте понятия сущности, связи и атрибута. Приведите примеры.
2. Приведите примеры атрибутов сущности и атрибутов связи.
3. Опишите алгоритм построения концептуальной схемы БД методом ER-диаграмм.
4. Опишите известные вам графические нотации ER-диаграмм.
5. Что такое степень связи? Каким образом определяется данная характеристика? Приведите примеры.
6. Что такое класс принадлежности? Каким образом определяется данная характеристика?
7. Перечислите правила формирования набора предварительных отношений.

**Проектирование триггеров. Рефакторинг**

1. Что такое триггеры БД? Для чего они предназначены? Приведите примеры задач, которые могут быть решены с помощью триггеров.
2. Чем определяется тип триггера?
3. Каким образом триггеры позволяют обеспечить ссылочную целостность при манипулировании данными?
4. Что такое рефакторинг качества данных? Приведите примеры.
5. Что такое рефакторинг ссылочной целостности? В каких случаях целесообразно к нему прибегать? Приведите примеры.
6. Что такое рефакторинг структуры? Приведите примеры.
7. Что такое секционирование таблицы. В каких случаях целесообразно прибегать к горизонтальному и в каких случаях – к вертикальному секционированию? Приведите примеры.

**6. Оценочные средства для текущего контроля успеваемости**

**6.1. Текущий контроль**

| №пп | № и наименование блока (раздела) дисциплины | Форма текущего контроля |
| --- | --- | --- |
| 1 | 1, 2, 3, 4, 5, 6 | Устный опрос |
| 2 | 2, 3, 4, 5, 6 | Защита отчёта по результатам выполнения лабораторных занятий |

**6.2. Примеры оценочных средств для текущего контроля по дисциплине**

***Вопросы для устного опроса***

1. История возникновения БД. Этапы развития БД.
2. Основные понятия теории баз данных.
3. Жизненный цикл базы данных.
4. Основные классы задач, решаемых с использованием баз данных
5. Типология БД: документальные и фактографические.
6. Типология БД: локальные и удаленные.
7. Типология БД: централизованные и распределенные.
8. Классификация моделей данных. Иерархическая и сетевая модели.
9. Реляционная модель данных: основные понятия, свойства отношений, представление отношений при помощи таблиц.
10. Связывание отношений в реляционной модели данных.
11. Целостность данных в реляционной модели: понятие, ограничения.
12. СУБД: понятие, структура, классификация.
13. Объекты реляционных СУБД.
14. СУБД MySQL: основные характеристики, область применения.
15. Язык SQL. Общая характеристика, состав, разновидности, достоинства.
16. Язык SQL. Описание схемы БД.
17. Язык SQL. Манипулирование данными.
18. Язык SQL. Выборка из БД.
19. Язык SQL. Представления.
20. Создание запросов на выборку к многотабличным базам данных.
21. Эталонная модель взаимодействия открытых систем (ISO OSI)
22. Система адресации в Internet.
23. Основные сервисы Internet.
24. Технология клиент-сервер.
25. Реализация клиентских сценариев на JavaScript.
26. Серверные web-приложения. Стандарт CGI.
27. Основные инструменты для разработки программ на PHP.
28. Механизмы и технологии доступа к БД из web-приложений.
29. Классификация приложений, использующих базы данных.
30. Разработка клиент-серверного приложения для просмотра БД.
31. Обработка ошибок.
32. Разработка триггеров.
33. Нормализация данных: понятие, цели. Аномалии модификации.
34. Этапы проектирования базы данных.
35. Проектирование базы данных методом ER-диаграмм.
36. Проектирование базы данных методом нормальных форм.
37. Рефакторинг.

***Задания для лабораторных занятий***

**Раздел 2. СУБД MySQL**

**Лабораторное занятие № 3. Извлечение информации из однотабличной БД**

Пусть в таблицу Sinema занесены следующие данные:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **time** | **film** | **vid** | **minut** | **cena** | **kol\_mest** | **kol\_prod** |
| 10:00 | Барбоскины | м/ф | 65 | 350 р. | 50 | 20 |
| 11:00 | Скромный приём | мелодрама | 115 | 300 р. | 100 | 70 |
| 12:00 | Как приручить дракона 2 | фэнтези | 102 | 350 р. | 50 | 42 |
| 12:00 | Годзилла | фильм ужасов | 125 | 300 р. | 100 | 39 |
| 14:00 | Как приручить дракона 2 | фэнтези | 102 | 350 р. | 50 | 38 |
| 15:00 | Ной | драма | 135 | 320 р. | 100 | 72 |
| 17:00 | Приключения Гекльберри Финна | семейное кино | 124 | 350 р. | 50 | 41 |
| 17:00 | Годзилла | фильм ужасов | 125 | 350 р. | 100 | 68 |
| 19:00 | Оборотень | фильм ужасов | 89 | 450 р. | 50 | 37 |
| 21:00 | Оборотень | фильм ужасов | 89 | 500 р. | 50 | 24 |
| 21:00 | Уроки выживания | фильм ужасов | 104 | 450 р. | 100 | 62 |

1. Определите назначение приведенного ниже запроса и укажите результат его работы для приведенных выше данных:

**SELECT** film **AS** [Название], time **AS** [Начало сеанса], cena **AS** Билет

**FROM** Sinema

**WHERE** (vid Like ”\*драма\*”) **AND** (cena <= 300)

**ORDER BY** 3 DESC**;**

1. Определите назначение приведенного ниже запроса и укажите результат его работы для приведенных выше данных:

**SELECT** vid, avg(cena)

**FROM** Sinema

**WHERE** minut>90

**GROUP** **BY** vid

**HAVING** max(cena)>300

1. Приведите текст SQL-запроса, который позволял бы вывести название самого короткого фильма, демонстрируемого после 14:00 в большом зале.

**Лабораторное занятие № 6. Создание запросов к многотабличным БД**

Имеется многотабличная БД следующей структуры (поля data\_begin и data\_end предназначены для хранения *даты приёма* на работу и *даты увольнения* соответственно):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FizLiza** |  | **Sotrud** |  | **Dolgn** |
| ☑ | **IDFL** |  | ☑ | **IDFL** |  | ☑ | **IDdolj** |
|  | FIO |  | ☑ | **IDdolj** |  |  | dolj |
|  | birthday |  |  | data\_begin |  |  | oklad |
|  | address |  |  | data\_end |  |  |  |
|  | telephone |  |  |  |  |  |  |

В таблицу **Sotrud** занесены следующие данные:

|  |  |  |  |
| --- | --- | --- | --- |
| **IDFL** | **IDdolj** | **data\_begin** | **data\_end** |
| 173254 | 021 | 01.03.2013 |  |
| 178145 | 034 | 07.04.2011 |  |
| 179562 | 025 | 05.11.2012 | 01.05.2013 |
| 176321 | 016 | 28.12.2001 |  |
| 175684 | 016 | 06.08.2007 |  |

В таблицу **Dolgn** занесены следующие данные:

|  |  |  |
| --- | --- | --- |
| **IDdolj** | **dolj** | **oklad** |
| 016 | Менеджер по клинингу | 12 500 р. |
| 019 | Офис-менеджер | 15 350 р. |
| 021 | Кассир | 16 200 р. |
| 025 | Администратор | 21 500 р. |
| 031 | Старший менеджер | 42 900 р. |
| 034 | Управляющий | 85 100 р. |

1. Запишите операторы, которые позволят создать многотабличную БД описанной выше структуры.
2. Определите назначение приведенного ниже запроса, укажите результат объединения таблиц и результат работы запроса для приведенных выше данных:

**SELECT** Dolgn.dolj, count(Sotrud.IDFL) AS [Количество]

**FROM** Dolgn **LEFT JOIN** Sotrud **ON** Dolgn.IDDolg = Sotrud.IDDolg

**GROUP BY** Dolgn.dolj

1. Приведите текст SQL-запроса, который позволял бы увеличить на 5% величину оклада тех должностей, на которых работает менее 3 человек.

**Раздел 4. Разработка клиентских web-сценариев**

**Лабораторное занятие № 10. Использование регулярных выражений для электронной обработки текстов**

1. Опишите шаблоны, заданные следующим образом:

**VAR** quotedText = /'([^']\*)'/;

 expressionPattern = /(\d+)\s\*([\+\\*\-\/])\s\*(\d+)/,

1. Опишите назначение приведенного ниже фрагмента сценария и укажите, что будет выведено в результате его работы:

names = **new** String ( "Harry Trump ;Fred Barney; Helen Rigby ;\
       Bill Abel ;Chris Hand ")

document.write ("---------- Original String" + "<BR>" + "<BR>");
document.write (names + "<BR>" + "<BR>");

pattern = /\s\*;\s\*/;

nameList = names.split (pattern);

pattern = /(\w+)\s+(\w+)/;

bySurnameList = new Array;

document.write ("---------- After manipulation" + "<BR>");
**for** ( i = 0; i < nameList.length; i++) **{**
   document.write (nameList[i] + "<BR>");
   bySurnameList[i] = nameList[i].replace (pattern, "$2, $1")
**}**

document.write ("---------- Results" + "<BR>");
**for** ( i = 0; i < bySurnameList.length; i++) **{**
   document.write (bySurnameList[i] + "<BR>")
**}**

1. Напишите сценарий, позволяющий проверить введенные пользователем значения (вид формы для заполнения приведен ниже). Прежде всего, необходимо убедиться в том, что заполнены все обязательные поля, затем проверить, что в имени пользователя отсутствуют какие-либо символы, кроме букв, пробела и дефиса и, наконец, проверить, соответствует ли введенная в поле адреса электронной почты строка общепринятой структуре электронного адреса.



**Раздел 5. Создание серверных приложений**

**Лабораторное занятие № 12. Разработка серверных сценариев для обработки строк и массивов**

1. Опишите назначение приведенного ниже фрагмента сценария и приведите пример его работы для конкретных данных:

**function** split\_to\_chunks($to,$text){

 $total\_length = (140 - strlen($to));

 $text\_arr = explode(" ",$text);

 $i=0;

 $message[0]="";

 foreach ($text\_arr as $word){

 **if** (strlen($message[$i] . $word . ' ')<=$total\_length ){

 **if** ($text\_arr[count($text\_arr)-1] == $word){

 $message[$i] .= $word;

 } **else** { $message[$i] .= $word . ' ';

 } } **else** {

 $i++;

 **if** ($text\_arr[count($text\_arr)-1] == $word){

 $message[$i] = $word;

 } **else** {

 $message[$i] = $word . ' ';

 } } }

 **return** $message;}

1. Напишите функцию для подсчета количества вхождений заданной подстроки в строку.
2. Напишите функцию для сортировки массива по убыванию методом Шейкерной сортировки.

**Лабораторное занятие № 13. Разработка серверных сценариев для работы с файлами и каталогами**

1. Опишите назначение приведенного ниже фрагмента сценария:

**function** infoFile ( $f )

{

 **if** ( !file\_exists( $f ) )

 {

 echo "$f нет!";

 **return**;

 }

 echo "$f - ".( is\_file( $f ) ? "" : "не " )."файл<br>";

 echo "$f - ".( is\_dir( $f ) ? "" : "не " )."каталог<br>";

 echo "$f ".( is\_readable( $f ) ? "" : "не " )."чтение <br>";

 echo "$f ".( is\_writable( $f ) ? "" : "не " )."запись <br>";

 echo "$f - ".( filesize( $f ) )." б <br>";

 echo "редактирование $f - ".( date( "d M Y H:i", filemtime( $f )))."<br>";

 echo "чтение к $f - ".( date( "d M Y H:i", fileatime( $f ) ) )."<br>";

}

1. Напишите функцию для вывода на экран второй половины файла.
2. Напишите функцию (функции), позволяющую создать каталог, скопировать в него файл и дописать (в конец скопированного файла) количество символов, которые в него были записаны.

**Раздел 6. Создание серверных web-приложений БД**

**Лабораторное занятие № 16. Разработка клиент-серверного приложения для просмотра и редактирования БД**

1. Опишите назначение приведенного ниже фрагмента сценария:

**function** process\_data()

 {

 $sql = trim($\_REQUEST["sql"]);

 **if** (strcasecmp("select", substr($sql,0,6))!=0)

 exit("Недопустимый SQL запрос!");

 $connection = mysql\_connect("localhost", "логин", "пароль")

 **or** exit("Ошибка соединения с сервером базы данных!");

 $db = mysql\_select\_db("имя базы данных", $connection)

 **or** exit("Ошибка соединения с базой данных!");

 $result = mysql\_query($sql);

 **if** ($result)

 {

 $rows=mysql\_num\_rows($result);

 $cols=mysql\_num\_fields($result);

 echo "<b> ИТОГО:</b> ".$rows." Записей <br/>";

 echo "<table><tr>";

 **for**($i=0; $i<$cols; $i++)

 {

 $field = mysql\_field\_name($result, $i);

 echo "<th>$field</th>";

 }

 echo "</tr>";

 **for**($i=0; $i<$rows; $i++)

 {

 echo "<tr>";

 for($j=0; $j<$cols; $j++)

 {

 $field=mysql\_result($result, $i, $j);

 echo "<td>$field</td>";

 }

 echo "</tr>";

 }

 echo "</table>";

 }

 **else** {echo "<b>Ошибка:</b> ".mysql\_error()."<br/>";}

 mysql\_close($connection);

 }

1. Опишите назначение приведенного ниже фрагмента сценария:

**function** process\_data()

 {

 $sql = trim($\_REQUEST["sql"]);

 $connection = mysql\_connect("localhost", "логин", "пароль")

 **or** exit("Ошибка соединения с сервером базы данных!");

 $db = mysql\_select\_db("имя базы данных", $connection)

 **or** exit("Ошибка соединения с базой данных!");

 mysql\_query("start ") **or** exit("start transaction: ".mysql\_error());

 mysql\_query($sql);

 mysql\_query("commit");

 mysql\_close($connection);

 }

1. Напишите функцию, позволяющую проверить полноту и корректность полученных от пользователя данных и (в случае отсутствия ошибок) добавить запись в БД (на примере любой БД).

**Раздел 7. Проектирование реляционных БД**

**Лабораторное занятие № 18. Проектирование БД методом нормальных форм**

1. Определите, в какой нормальной форме находится данное отношение:

|  |  |  |
| --- | --- | --- |
| **ФИО родителя** | **Имя ребенка** | **Услуга** |
| *Смородина И.Л.* | *Елена* | *проверка зрения* |
| *Клубникин П.П.* | *Вячеслав, Татьяна* | *общий анализ крови* |
| *Барбарискина Н.М.* | *Николай* | *справка 026/у* |

1. Определите, в какой нормальной форме находятся данные отношения:

|  |  |  |
| --- | --- | --- |
| **СОТРУДНИКИ** |  | **ОТДЕЛЫ** |
| **🗝** | **код сотрудника** |  | **🗝** | **код отдела** |
| **🗝** | **код проекта** |  |  | название отдела |
|  | ФИО сотрудника |  |  | ФИО начальника |
|  | должность |  |  |  |
|  | образование |  |  |  |
|  | код отдела |  |  |  |

1. Опишите процесс приведения описанного в предыдущем задании отношений к третьей нормальной форме.

**Лабораторное занятие № 19. Проектирование БД методом ER-диаграмм**

1. Определите характеристики связи между сущностями «Преподаватели» и «Дисциплины», если известно, что каждый преподаватель должен вести только одну дисциплину, а дисциплина может читаться несколькими преподавателями.
2. На приведенном ниже рисунке представлена ER-диаграмма, построенная на основе анализа предметной области. Дайте описание предметной области, соответствующее приведенной ER-диаграмме:

|  |
| --- |
|  |

1. Для приведенной в предыдущем задании ER-диаграммы постройте предварительный набор отношений, подробно описав все выполняемые действия и использованные правила.

**7. ПЕРЕЧЕНЬ УЧЕБНОЙ ЛИТЕРАТУРЫ:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № п/п | Наименование | Авторы | Место издания | Год издания | Наличие |
| печатные издания | ЭБС (адрес в сети Интернет) |
| 1. | Базы данных: проектирование: учебник для вузов | Стружкин Н.П., Годин В.В. | М.: Издательство Юрайт | 2022 |  | <https://urait.ru/bcode/489099> |
| 2. | Базы данных: учебник и практикум для вузов | Нестеров С.А.  | М.: Издательство Юрайт | 2022 |  | <https://urait.ru/bcode/489693> |
| 3. | Базы данных: учебник для вузов | Советов Б.Я., Цехановский В.В., Чертовской В.Д. | М.: Издательство Юрайт | 2022 |  | <https://urait.ru/bcode/488866> |
| 4. | Базы данных: практикум по проектированию реляционных баз данных: учебное пособие | Сидорова Н.П.  | М.; Берлин: Директ-Медиа | 2020 |  | [https://biblioclub.ru/](https://biblioclub.ru/index.php?page=book&id=575080) |
| 5. | Базы данных: учебно-методическое пособие | Жуков Р.А.  | М.; Берлин: Директ-Медиа | 2019 |  | [https://biblioclub.ru/](https://biblioclub.ru/index.php?page=book&id=575080) |
| 6. | Базы данных: технологии доступа: учебное пособие для вузов | Стасышин В.М., Стасышина Т.Л. | М.: Издательство Юрайт | 2022 |  | <https://urait.ru/bcode/492177> |

**8. Ресурсы информационно-телекоммуникационной сети «Интернет»:**

1. «НЭБ». Национальная электронная библиотека. – Режим доступа: [http://нэб.рф/](http://www.biblioclub.ru/)

2. «eLibrary». Научная электронная библиотека. – Режим доступа: [https://elibrary.ru](https://elibrary.ru/)

3. «КиберЛенинка». Научная электронная библиотека. – Режим доступа: <https://cyberleninka.ru/>

4. ЭБС «Университетская библиотека онлайн». – Режим доступа: [http://www.biblioclub.ru/](http://www.knigafund.ru/)

5. Российская государственная библиотека. – Режим доступа: <http://www.rsl.ru/>

**9. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, ИСПОЛЬЗУЕМЫЕ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ:**

В ходе осуществления образовательного процесса используются следующие информационные технологии:

- средства визуального отображения и представления информации (LibreOffice) для создания визуальных презентаций как преподавателем (при проведении занятий) так и обучаемым при подготовке докладов для семинарского занятия.

- средства телекоммуникационного общения (электронная почта и т.п.) преподавателя и обучаемого.

- использование обучаемым возможностей информационно-телекоммуникационной сети «Интернет» при осуществлении самостоятельной работы.

**9.1. Требования к программному обеспечению учебного процесса**

Для успешного освоения дисциплины, обучающийся использует следующие программные средства:

* Windows 10 x64
* MicrosoftOffice 2016
* LibreOffice
* Firefox
* GIMP

**9.2. Информационно-справочные системы (при необходимости):**

Не используются

**10. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

Для проведения занятий лекционного типа предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий.

Для изучения дисциплины используется следующее оборудование: аудитория, укомплектованная мебелью для обучающихся и преподавателя, доской, ПК с выходом в интернет, мультимедийным проектором и экраном.

Для самостоятельной работы обучающихся используется аудитория, укомплектованная специализированной мебелью и техническими средствами (ПК с выходом в интернет и обеспечением доступа в электронно-информационно-образовательную среду организации).