

МОДЕЛИ ДАННЫХ и СИСТЕМЫ УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ

Общие требования:

1. Для выполнения лабораторных работ вы можете использовать любую РЕЛЯЦИОННУЮ СУБД (ex. PostgreSQL, MySQL)
2. Для сдачи лабораторной работы Вам нужно будет создать git репозиторий, где README.md файл содержит ваши данные (ИМЯ, ФАМИЛИЯ, НОМЕР ГРУППЫ)

В данном курсе вы будете проектировать базу данных в контексте приложения, предметную область которого вы определите в первой лабораторной. Все дальнейшие лабораторные работы будут направлены на проектирование и разработку базы данных, каждая последующая лабораторная будет дополнять предыдущую и расширять функционал вашей БД.

В конце курса вы получите реализованную базу данных, набор SQL-запросов и инструкций, готовых для применения в коде проекта. Взаимодействие с базой данных в вашем проекте будет производиться **исключительно** посредством данного набора. Проект должен также иметь интерфейс взаимодействия с пользователем.

Последняя лабораторная работа будет представлять собой небольшой готовый проект, в котором в качестве базы данных будет выступать реализованная вами в ходе семестра БД.

Лабораторная работа №1

В данной лабораторной работе необходимо:

1. Определить тему разрабатываемого в семестре проекта.
2. Определить функциональные требования к проекту.
3. Определить и обосновать перечень сущностей БД, отвечающих функциональным требованиям проекта.
 - a. Количество связанных сущностей должно быть не меньше 10 (промежуточная таблица для МТМ не является сущностью, если не несет в себе дополнительной информации кроме непосредственной связи двух таблиц)
 - b. Должен присутствовать каждый из видов связей.
4. Схематично изобразить не нормализованную схему БД.
5. Описать каждую сущность (имя поля, тип, ограничения, связь с другими сущностями)

Acceptance Criteria:

1. README.md файл содержит выбранную вами тему, ФИО, номер группы.
2. Определен, описан и закоммичен файл с функциональными требованиями к проекту. Минимальные **ОБЯЗАТЕЛЬНЫЕ** функциональные требования:
 - a. Авторизация/аутентификация пользователя.
 - b. Управление пользователями (CRUD).
 - c. Система ролей.
 - d. Журналирование действий пользователя.
3. Определен список таблиц для БД (см. п. 3).
4. Каждая таблица описана в соответствии с условием лабораторной (см. п. 5).
5. Схематично изображена не нормализованная предлагаемая инфологическая модель БД (может быть использован любой графический редактор, e.g. draw.io, Miro.com, Microsoft Word, Power Point, Enterprise arch., etc.)
6. Теоретические знания по теме лабораторной.

Лабораторная работа №2

В данной лабораторной работе необходимо:

1. Разработать даталогическую модель утвержденной в первой лабораторной БД.
2. Привести БД к третьей нормальной форме.
 - а. В случае необходимости денормализации БД – быть готовым доказать свою точку зрения.

Acceptance Criteria:

1. Разработана даталогическая модель БД.
2. Выполнен пункт 2 из условия лабораторной.
3. Готовность привести БД к последующим нормальным формам (Бойса-Кодда, четвертой, пятой, шестой).
4. Теоретические знания по теме лабораторной.

Лабораторная работа №3

В данной лабораторной работе необходимо:

1. Разработать физическую модель базы данных (создать БД на вашем устройстве).
2. Наложить на базу данных ограничения.
3. Заполнить базу данных тестовыми значениями.
4. Создать пул запросов, необходимых для простых операций над данными в БД.

Acceptance Criteria:

1. Создана физическая модель базы данных с наложенными ограничениями в соответствии со схемой, утвержденной во второй лабораторной работы.
 - a. Все таблицы, связи, сущности должны быть написаны при помощи SQL скриптов самостоятельно, без автоматической генерации кода средой.
 - b. **ВСЕ ЗАПРОСЫ** для создания и заполнения БД необходимо **СОХРАНИТЬ** для демонстрации преподавателю.
 - c. Проставлены индексы, констрейнты, прочие ограничения.
2. База данных заполнена тестовыми данными.
3. Составлен пул SQL запросов для простых операций с БД.
4. Готовность продемонстрировать работу с простейшими (и немного усложненными) SQL запросами на существующей базе данных.
5. Готовность пояснить смысл всех операторов SQL, использованных в коде + операторов для простейшей выборки и фильтрации.

Лабораторная работа №4

В данной лабораторной работе необходимо:

1. Создать пул запросов для сложной выборки из базы данных.
2. Создать пул запросов для получения представлений в базе данных.
3. Создать пул запросов для получения сгруппированных данных.
4. Создать пул запросов, необходимых для сложных операций над данными в БД.
5. Проверить написанные запросы на вашей БД.

Acceptance Criteria:

1. Составлен пул SQL запросов для сложной выборки из БД:
 - a. Запросы с несколькими условиями.
 - b. Запросы с вложенными конструкциями
 - c. Прочие сложные выборки, необходимые в вашем проекте.
2. Составлен пул SQL запросов для получения представлений в БД:
 - a. JOIN-запросы различных видов (INNER, OUTER, FULL, CROSS, SELF)
3. Составлен пул SQL запросов для получения сгруппированных данных:
 - a. GROUP BY + агрегирующие функции
 - b. PARTITION, PARTITION OVER + оконные функции
 - c. HAVING
 - d. UNION
4. Составлен пул SQL запросов для сложных операций с данными:
 - a. EXISTS
 - b. INSERT INTO SELECT
 - c. CASE
 - d. EXPLAIN
5. Вышеперечисленные запросы являются только примерными, не обязательно все использовать в коде для вашей БД.
6. Готовность пояснить смысл всех операторов SQL, использованных в коде + операторов, перечисленных выше + готовность написать код любого оператора из лабораторной для ваших данных.
7. Теоретические знания по запросам и операторам.

Лабораторная работа №5

В данной лабораторной работе необходимо:

1. Создать пул триггеров, необходимых для корректной работы логики вашей базы данных.
2. Создать пул хранимых процедур, необходимых для корректной работы вашей базы данных.
3. Проверить написанные триггеры и процедуры на вашей БД.

Acceptance Criteria:

1. Составлен пул триггеров, необходимых для корректной работы БД в контексте вашего существующего ТЗ и утвержденной схемы:
 - a. Пример: пересчет суммы заказа при добавлении нового товара в заказ / логирование
 - b. Триггеры написаны на SQL и подключены к БД
 - c. Триггеры должны реализовывать необходимую логику будущего приложения
 - d. Один триггер на логирование действий не засчитывается как сданная лабораторная работа: пул триггеров должен реализовывать автоматизированную логику вашего приложения
2. Составлен пул хранимых SQL процедур для выполнения необходимых операций над БД:
 - a. Процедуры должны быть созданы логически для запросов, часто используемых в процессе работы приложения
 - b. Процедуры должны храниться в базе и вызываться из других запросов или консоли
3. Способность продемонстрировать работу триггеров и процедур
4. Способность реализовать новые триггеры и процедуры
5. Теоретические знания в контексте лабораторной

Лабораторная работа №6

В данной лабораторной работе необходимо:

1. Создать приложение с пользовательским интерфейсом, в котором разработанная в процессе семестра БД выступает в качестве основной базы данных проекта.

Acceptance Criteria:

1. Приложение должно иметь интерфейс взаимодействия с пользователем:
 - a. Если приложение консольное: реализован список команд, через которые пользователь взаимодействует с системой
 - b. Если приложение веб/десктоп: реализован условный frontend, через который пользователь взаимодействует с системой
2. Приложение должно соответствовать ТЗ, утвержденному во 2-й лабораторной
3. В качестве БД для проекта выступает база данных, спроектированная в ЛР 1-5 и подключенная к проекту напрямую
4. Работа и взаимодействие с БД осуществляется одним из трех способов:
 - a. Прямые запросы в БД из приложения
 - b. Самописная ORM (демонстрация ВАШЕГО исходного кода ORM обязательна)
 - c. ORM.raw_sql() или аналоги в ваших технологиях

Таким образом, взаимодействие с БД должно осуществляться **ИСКЛЮЧИТЕЛЬНО** при помощи SQL так или иначе. В противном случае лабораторная засчитана **НЕ БУДЕТ**

5. Для взаимодействия с БД необходимо использовать запросы, разработанные в ЛР 3-5
6. Способность добавить некоторые детали в код проекта непосредственно при сдаче
7. Теоретические знания в контексте лабораторной