## ГОУ ВПО «Донецкий национальный технический университет»

Факультет компьютерных наук и технологий

Кафедра программной инженерии

Утверждаю

Зори С.А.

\_\_\_\_\_\_\_\_\_\_\_

\_\_. \_\_.20\_\_ г.

## **ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

**на курсовую работу по дисциплине**

**«Программирование систем с серверами баз данных»**

выдано студенту группы ПИ-18 «Б» Моргунову А. Г.

**Тема:** **«Создание клиент-серверной информационной системы средствами СУБД»**

**Описание предметной области:**

Система учета работы комиссионного магазина должна содержать данные о комитентах (ФИО, район проживания, социальное положение (рабочий, служащий, пенсионер, домохозяйка, предприниматель,…), место работы, дата рождения, телефон), принимаемых товарах (номер квитанции, группа товара (посуда, одежда, обувь, игрушки,…), наименование товара, дата приема, количество единиц, цена за единицу) и о реализации товаров (товар, дата, количество проданных единиц).

Донецк – 2021

**Задание на курсовую работу**

1. Спроектировать концептуальную модель базы данных (БД) для заданной предметной области и представить ее в виде взаимосвязанных таблиц, находящихся в третьей нормальной форме (в случае денормализации БД – обосновать необходимость). Выделить базовые таблицы и таблицы-справочники, указать для них первичные и внешние ключи.
2. Создать базу данных в среде СУБД средствами языка SQL. Добавить таблицы, домены, индексы.
3. Разработать не менее шести триггеров (по одному для каждого типа события), как минимум для двух различных таблиц БД. Триггеры типа BEFORE INSERT должны быть созданы для всех таблиц и с использованием генераторов задавать значение первичного ключа для вновь добавляемой записи.
4. Заполнить таблицы БД с использованием соответствующих запросов на языке SQL (не менее десяти записей в каждом справочнике, не менее 10 000 - 50 000 псевдослучайных записей в таблицах).
5. Сформулировать следующие виды запросов:

* симметричное внутреннее соединение с условием (два запроса с условием отбора по внешнему ключу, два – по датам);
* симметричное внутреннее соединение без условия (три запроса);
* левое внешнее соединение;
* правое внешнее соединение;
* запрос на запросе по принципу левого соединения;
* итоговый запрос без условия;
* итоговый запрос без условия c итоговыми данными вида: «всего», «в том числе»;
* итоговые запросы с условием на данные (по значению, по маске, с использованием индекса, без использования индекса);
* итоговый запрос с условием на группы;
* итоговый запрос с условием на данные и на группы;
* запрос на запросе по принципу итогового запроса;
* запрос с использованием объединения
* запросы с подзапросами (с использованием in, not in, case, операциями над итоговыми данными).

1. Запросы без параметров реализовать в виде представлений, остальные запросы – в виде хранимых процедур и/или функций. Создать, по меньшей мере, одно модифицируемое представление, используя механизм триггеров. ВСЯ логика проектируемого ПО – на сервере.
2. Разработать клиентское приложение, которое предоставляет следующие возможности для работы с созданной базой данных:

* многопользовательский режим работы (одна программа для всех ролей – ситуативный доступ к интерфейсу)
* наличие нескольких ролей пользователя (администратор – добавление/удаление/редактирование пользователей, их прав/ролей; пользователи\_1 – …, пользователи \_2 – ...)
* просмотр содержимого таблиц и представлений (здесь и далее – с учетом прав пользователей);
* добавление, редактирование и удаление записей таблиц и модифицируемых представлений;
* работа с наборами данных, находящимися в отношении «один-ко-многим» (создать составную форму для просмотра и редактирования данных родительской и дочерней таблиц);
* поиск и фильтрация данных отображаемых таблиц;
* просмотр результатов выполнения запросов;
* визуализация результатов одного из итоговых запросов (диаграммы, экспорт в Excell).

1. Обеспечить защиту данных, информации от несанкционированного доступа

**Рекомендуемое содержание пояснительной записки**

Введение

1. Описание предметной области, постановка задачи
2. Обоснование выбора СУБД, описание возможностей СУБД
3. Обоснование выбора инструментальные средств для написания клиентской части, проектирование структуры ПО
   1. Невизуальные компоненты для работы с данными
   2. Визуальные компоненты отображения данных
   3. Разработка шаблонов приложений для работы c таблицами базы данных
4. Проектирование базы данных в выбранной СУБД
   1. Проектирование концептуальной модели БД
   2. Создание таблиц, доменов, индексов
   3. Разработка триггеров
   4. Проектирование запросов к базе данных
   5. Создание представлений и хранимых процедур, функций
5. Разработка клиентского приложения
   1. Формы и компоненты для работы с основными таблицами
   2. Формы и компоненты для работы со справочниками
   3. Формы и компоненты для отображения результатов запросов
6. Тестирование разработанной информационной системы (в т.ч. включая защиту от несанкционированного доступа, одновременную работы с данными, каскадное удаление)

Заключение

Приложение А. Техническое задание

Приложение Б. Листинг шаблонов

Приложение В. Листинг серверного приложения

Приложение Д. Листинг клиентского приложения

Приложение Е. Руководство пользователя

Приложение Ж. Руководство администратора

**График выполнения курсовой работы**

|  |  |
| --- | --- |
| Неделя | Работа |
| 1-2 | Выдача и изучение задания |
| 3 | Анализ требований к системе и способов их реализации |
| 4-5 | Проектирование и реализация БД (таблицы, домены, индексы) |
| 6-7 | Создание триггеров и заполнение таблиц БД |
| 8-9 | Создание представлений и хранимых процедур |
| 10-13 | Разработка клиентского приложения |
| 14 | Тестирование и отладка системы |
| 15 | Оформление пояснительной записки |
| 16-17 | Защита курсовой работы |

Дата выдачи задания \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Задание принял \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководители проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**