МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГТУ», ВГТУ)

Факультет информационных технологий и компьютерной безопасности

Кафедра компьютерных интеллектуальных технологий проектирования

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7

По дисциплине: «Системы хранения и обработки данных»

Тема: «Разработка логической структуры базы данных»

Выполнил работу студент группы мИИВТ-241: Боченков А.С.

подпись, дата

Принял: Короленко В.В.

подпись, дата

Воронеж 2024

Учебная задача

3. Определить основные сущности для разрабатываемой базы данных (не менее 4). Обосновать выбор.

В качестве примера была выбрана задача построить базу данных «Учет нарушений правил дорожного движения». Для ее создания были введены следующие сущности:

* Cars (Автомобили)
* Violations (Нарушения ПДД)
* Humans (Люди)
* Insurance (Страхование)
* Facts (Факты нарушения ПДД)

Этот набор сущностей был выбран и потому, что максимально полно описывает ситуацию транспортного происшествия, а именно в ней участвуют водители, их автомобили. У каждого автомобиля должна быть страховка. При ДДП фиксируется и устанавливается факт и вид нарушения.

4. Определить атрибуты сущностей.

Cars (Автомобили):

* id (первичный ключ)
* brand (бренд)
* model (модель)
* year\_create (год выпуска)
* number\_gov (государственный номер автомобиля)

Violations (Нарушения ПДД):

* id (первичный ключ)
* name (название нарушения)
* payment (сумма штрафа)

Humans (Люди):

* id (первичный ключ)
* fio (Ф.И.О.)
* sex (пол)
* birthday (дата рождения)

Insurance (Страхование):

* insurance\_num (номер страховки, первичный ключ)
* body\_payment (стоимость страхового объекта)
* percent\_insurance (процент страховки)
* payment (сумма страховых выплат)

Facts (Факты нарушения ПДД):

* id (первичный ключ)
* car\_id (внешний ключ таблицы cars)
* driver (внешний ключ таблицы humans)
* owner (внешний ключ таблицы humans)
* date (дата ДДТ)
* violation\_id (внешний ключ таблицы violations)
* drive\_right (право на вождение, кто находился за рулем автомобиля)
* insurance\_num (внешний ключ таблицы insurance)

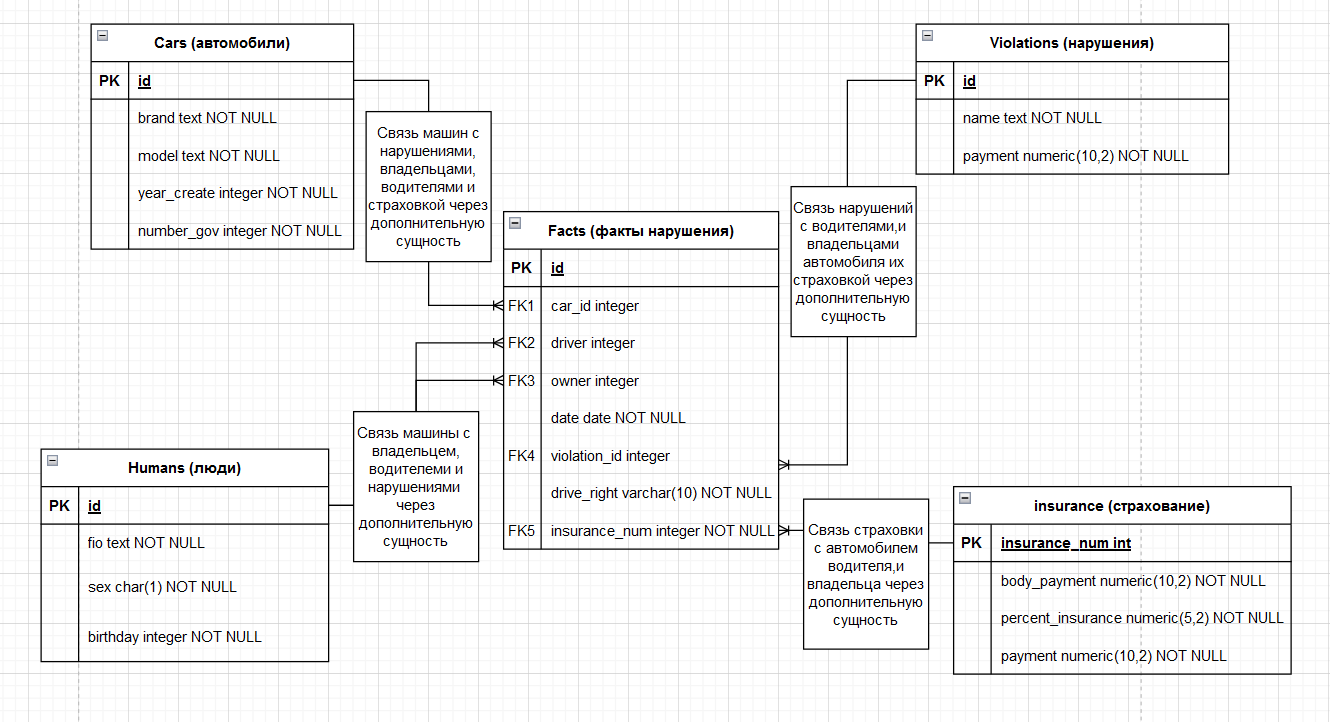
5. Выделить ключевые атрибуты сущностей (первичные и внешние ключи). Пояснить свой выбор.

У каждой таблицы введен атрибут id, который служит в роли первичного ключа. У таблицы Insurance (Страхование) роль первичного ключа выполняет атрибут insurance\_num (номер страховки), который называется естественным первичным ключом. Реляционная теория требует, чтобы в каждом домене каждый кортеж был уникальной единицей, для этого вводится дополнительный атрибут, либо один из естественных атрибутов выполняет эту роль, если это возможно.

6. Определить связи между сущностями. Как минимум одна связь должна быть «многие ко многим».

С помощью комбинации сущностей: автомобили, люди, страхование, можно построить связь «многие ко многим». Например, связь автомобили-люди связаны отношением владения. Один человек имеет право управлять несколькими машинами, а у одной машины могут быть несколько людей, которые имеют право ей управлять. Эта связь в реляционной теории не допустима, поэтому мы найдем другое решение, а именно введем новую сущность факты нарушения, которая будет связывать все остальные сущности и это не вызовет коллизии, потому что каждой записи можно будет установить единственное, точное соответствие.

7 – 11.



На данном рисунке изображена итоговая логическая схема базы данных «Учет нарушений правил дорожного движения». На ней видно, что путем добавления сущности факты нарушения мы избавились от связей «многие ко многим». Так же на ней представлены и подписаны виды отношений к другим сущностям через нее, указаны и подписаны ключи первичные (PK) и внешние (FK).

Контрольные вопросы

1. Дайте определения следующим понятиям: данные, база данных, СУБД, ведение базы данных.

*Данные* – представление объектов реального мира и их свойств в формализованном виде, пригодном для хранения, передачи, интерпретации или обработки. В случае использования данных для уменьшения неопределенности знаний о каком-либо объекте данные превращаются в информацию.

*База данных (БД)* – совокупность данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и манипулирования данными, независимая от прикладных программ.

*Система управления базами данных (СУБД)* – совокупность программ и языковых средств, предназначенных для управления данными в базе данных, ведения базы данных и обеспечения взаимодействия ее с прикладными программами.

*Ведение базы данных* – деятельность по обновлению, восстановлению и перестройке структуры базы данных с целью обеспечения ее целостности, сохранности и эффективности использования.

2. В чем отличие данных от информации?

Информация - факты, события, вещи, процессы, идеи, понятия или иные касающиеся объектов знания, которые имеет особое значение в определенном контексте. Задача информации - влиять на суждение и поведение получателя. В отличие от данных у информации имеются смысл, значимость и назначение. Данные становятся информацией, если их создатель добавляет к ним смысл.

3. В чем отличие базы данных от банка данных и СУБД?

База данных представляет собой совокупность, организованных по определенным правилам, системы управления базами данных – комплекс программ по манипулированию данными в базах, а банк данных - это совокупность одной или нескольких баз данных и средств управления (манипулирования) данными, является важнейшей составной частью автоматизированных информационных и других систем, функционирование которых связано с обработкой больших объемов данных.

4. Назовите основные компоненты банка данных и их назначение.

Компоненты банка данных:

* БД
* СУБД
* Приложения (комплекс языков, программ для получения, преобразования данными)
* Технические средства (средства ЭВМ)
* Администрирование

5. Классифицируйте АИС по типу хранимых данных.

Классификация по типу хранимых данных:

1. *Документальные информационно-поисковые системы (ДИПС)* предназначены для хранения и обработки документальных данных – адресов хранения документов, наименований, описаний и рефератов, а также текстов документов. Такие данные представляются в неструктурированном виде. Примером ДИПС являются библиотечные, библиографические АИС.

2. *Фактографические информационно-поисковые системы (ФИПС)* хранят и обрабатывают фактографическую информацию – структурированные данные в виде чисел и текстов. Над такими данными можно выполнять различные операции. Большинство разрабатываемых АИС представляют собой системы класса ФИПС.

6. Что понимается под трехуровневой архитектурой ANSI/SPARC?

Архитектура ANSI-SPARC (также трёхуровневая архитектура) — принципы, согласно которому рекомендуется строить системы управления базами данных (СУБД). Проект архитектуры был выдвинут в 1975 году подкомитетом SPARC ANSI.

Выделяют три уровня системы:

1. Внешний (пользовательский) уровень - представление базы данных с точки зрения пользователей. Каждый пользователь имеет определенный доступ к базам данным и может получать данные в удобной для него форме.

2. Промежуточный (концептуальный) уровень – обобщающее представление базы данных. Этот уровень описывает логическую структуру, какие данные хранятся в базе данных, связи между ними. Полное представление требований к данным со стороны организации.

3. Внутренний (физический) описывает физическую реализацию базы данных для достижения максимальной надежности и производительности, экономию места в памяти.

7. Дайте определения внешней схеме БД, концептуальной схеме БД, внутренней схеме БД.

Внешняя схема описывает ту часть базы данных, в которой заинтересован конкретный пользователь. Она скрывает от пользователя несвязанные детали базы данных. Может быть множество внешних представлений для каждой базы данных.

Концептуальная схема описывает структуру базы данных всей базы данных для сообщества пользователей. Эта схема скрывает информацию о физических структурах хранения и фокусируется на описании типов данных, объектов их отношений.

Внутренняя схема определяет физическую структуру хранения базы данных. Внутренняя схема представляет собой очень низкоуровневое представление всей базы данных.

8. Каковы особенности иерархической модели организации данных?

Особенности иерархической модели:

* достаточно эффективное использование памяти
* средние временные показатели выполнения операций над данными
* предназначена для работы с иерархически организованными данными
* сложные логические связи
* громоздкость в обработке данных

9. Каковы особенности сетевой модели организации данных?

Особенности сетевой модели:

* базу данных сложно реорганизовать
* сложность механизма доступа к данным
* сложность реализации СУБД

10. Каковы особенности многомерной модели организации данных?

Особенности многомерной модели:

* удобство и эффективность анализа больших данных, имеющих временную связь
* быстрота реализации сложных нерегламентированных запросов
* громоздкость для решения стандартных задач оперативной обработки

11. Каковы особенности постреляционной модели организации данных?

Особенности постреляционной модели:

* допускает многозначные поля, значение которых состоят из подзначений и набор значений воспринимается как самостоятельная таблица
* поддерживает множественные группы, называемые ассоциированными множественными полями, а совокупность множественных полей называется ассоциацией

12. Что относится к неструктурированным данным?

Неструктурированные данные определяются как данные, представленные в абсолютно необработанной форме. Эти данные сложно обрабатывать из-за их сложной организации и форматирования. Неструктурированные данные включают публикации в социальных сетях, чаты, электронные письма и презентации.

13. В чем преимущество использования колоночной СУБД по сравнению с реляционной?

Данные об одном объекте хранятся в одном столбце, что позволяет хранить различные значения физически рядом и это позволяет быстрее манипулировать данными объекта.

Колоночная СУБД лучше традиционной реляционной, если:

* подавляющее большинство запросов — на чтение
* данные обновляются достаточно большими пачками (> 1000 строк), а не по одной строке, или не обновляются вообще
* данные добавляются в БД, но не изменяются
* запросы идут сравнительно редко (обычно не более сотни в секунду на сервер)
* требуется высокая пропускная способность при обработке одного запроса (до миллиардов строк в секунду на один сервер)

14. Каким образом осуществляется связь между таблицами в реляционной СУБД?

Связь в реляционной СУБД осуществляется с помощью ключей (первичных, внешних и составных). Для того, чтобы связать две таблицы в одной должен быть уникальный идентификатор (первичный ключ), а во второй должен быть атрибут, который хранит значения первичного ключа из первой таблицы, такой ключ называется внешним. Составной ключ – это уникальный идентификатор кортежа, в котором значение уникальности гарантируется не значением одного атрибута, а нескольких. Например, номер и серия паспорта является составным ключом и к тому же естественным.

15. Каким образом на этапе проектирования решается проблема дублирующих записей в таблице?

Анализом постоянно повторяющихся данных и выделением их в отдельную сущность, а в основной таблице иметь внешний ключ ссылающийся на новую сущность.