**Государственное бюджетное профессиональное**

**образовательное учреждение Астраханской области**

**«Астраханский колледж вычислительной техники»**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **К защите допустить:** | | | | | | | |
| **Старший методист УМС** | | | | | | | |
|  |  | | | **Л.О.Земцова** | | | |
| **«** |  | **»** |  | | **20** |  | **г.** |

**курсОВАЯ РАБОТА**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| по МДК | 04.01 | Технология разработки и защиты базы данных | | | | | | |
|  | | | | | | | | |
|  | | | 6 семестр | | | | | |
| Тема курсовой работы | | | «Фирма по продаже автозапчастей» | | | | | |
|  | | |  | | | | | |
|  | | |  | | | | | |
| Аквт.09.02.07.КрТРЗБД.13ПЗ | | | | | | | | |
| Листов: | | | | 13 |  | | | |
|  | |  | | | | | | |
|  | |  | | | | | | |
| Разработчик | |  | | | | Студент гр. | | ПБ – 31 |
|  | |  | | | | Акобян Э. А | | |
|  | |  | | | |  |  | |
| Руководитель | |  | | | | Преподаватель | | |
|  | |  | | | | Храмцовский И.А. | | |
|  | |  | | | |  | | |
|  | |  | | | |  | | |

**2023**

Оглавление

[Введение 3](#_Toc137849531)

[1 Современное проблемы баз данных 4](#_Toc137849536)

[1.1 Классификация баз данных 4](#_Toc137849537)

[1.2 Анализ рынка программного обеспечения СУБД 6](#_Toc137849538)

[2.1 Описание предметной области 8](#_Toc137849544)

[2.2 Перечень сущностей 8](#_Toc137849558)

[3 Разработка базы данных для продажи запчастей авто. 9](#_Toc137849559)

[3.1Построение логической модели, схемы базы данных 9](#_Toc137849560)

## 4.1 Создание таблиц в Postgresql……………………………………………………………………………………………………………………10

[ЗАКЛЮЧЕНИЕ 12](#_Toc137849561)

# **Введение**

# Процесс проектирования БД на основе принципов нормализации представляет собой последовательность переходов от неформального словесного описания информационной структуры предметной области к формализованному описанию объектов предметной области в терминах некоторой модели.

# Инфологическая модель применяется на втором этапе проектирования БД, то есть после словесного описания предметной области. Процесс проектирования длительный и требует обсуждений с заказчиком и со специалистами в предметной области. Наконец, при разработке серьезных корпоративных информационных систем проект базы данных является тем фундаментом, на котором строится вся система в целом, и вопрос о возможном кредитовании часто решается экспертами банка на основании именно грамотно сделанного инфологического проекта БД. Следовательно, инфологическая модель должна включать такое формализованное описание предметной области, которое легко будет «читаться» не только специалистами по базам данных. И это описание должно быть настолько емким, чтобы можно было оценить глубину и корректность проработки проекта БД, и конечно, оно не должно быть привязано к конкретной СУБД. Выбор СУБД - это отдельная задача, для корректного ее решения необходимо иметь проект, который не привязан ни к какой конкретной СУБД.

# Инфологическое проектирование прежде всего связано с попыткой представления семантики предметной области в модели БД.

# Целью данной курсовой работы является систематизация, накопление и закрепление знаний о построении инфологической модели и построение инфологической модели базы данных приложения (магазина) «Автозапчасти».

**1 Современное проблемы баз данных**

* 1. **Классификация баз данных**

База данных – это информационная модель предметной области, совокупность взаимосвязанных, хранящихся вместе данных при наличии такой минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений. Данные (файлы) хранятся во внешней памяти и используются в качестве входной информации для решения задач.

СУБД – это программа, с помощью которой реализуется централизованное управление данными, хранимыми в базе, доступ к ним, поддержка их в актуальном состоянии.

Классификация по модели данных:

– иерархическая,

– объектная и объектно-ориентированная,

– объектно-реляционная СУБД (ОРСУБД),

– реляционная модель данных (РМД),

– сетевая модель данных,

Иерархическая – это модель данных, где используется представление базы данных в виде древовидной(иерархической) структуры, состоящей из объектов (данных) различных уровней. Между объектами существуют связи, каждый объект может включать в себя несколько объектов более низкого уровня. Такие объекты находятся в отношении предка (объект более близкий к корню) к потомку (объект более низкого уровня), при этом возможна ситуация, когда объект-предок не имеет потомков или имеет их несколько, тогда как у объекта-потомка обязательно только один предок. Объекты, имеющие общего предка, называются близнецами (в программировании применительно к структуре данных дерево устоялось название братья).

Объектная и объектно-ориентированная – являются системой управления базами данных, в которых информация представлена в виде объектов, используется в объектно-ориентированном программировании. Объектные базы данных отличаются от реляционных баз данных, являющихся таблично-ориентированными. Объектно-реляционные базы данных являются гибридом обоих подходов. Объектные базы данных были рассмотрены в начале 1980-х годов;

Объектно-реляционная СУБД (ОРСУБД)  – реляционная СУБД (РСУБД), поддерживающая некоторые технологии, реализующие объектно-ориентированный подход: объекты, классы и наследование реализованы в структуре баз данных и языке запросов.Объектно-реляционными СУБД являются, например, широко известные Oracle Database, Informix, DB2, PostgreSQL.

Реляционная модель данных (РМД) – логическая модель данных, прикладная теория построения баз данных, которая является приложением к задачам обработки данных таких разделов математики как теории множеств и логика первого порядка.

Сетевая модель данных – логическая модель данных, являющаяся расширением иерархического подхода, строгая математическая теория, описывающая структурный аспект, аспект целостности и аспект обработки данных в сетевых базах данных. Функциональная модель данных использует такой подход для определения объекта. Вместо того чтобы представлять объект записью с определенным содержанием или же кортежем в дереве, функциональная модель сообщает, какие функции ( или операции) определены на этом объекте.

Представление объекта – это дело реализации, и оно определяется на более низком уровне абстракции.

Классификация по среде постоянного хранения:

Во вторичной памяти, или традиционная – средой постоянного хранения является периферийная энергонезависимая память (вторичная память), как правило, жёсткий диск. В оперативную память СУБД помещает лишь кэш и данные для текущей обработки.

1.2 Анализ рынка программного обеспечения СУБД

Информацией, хранящейся в базе данных (БД), может быть всё что угодно: каталог продукции, информация о клиентах, контент веб-сайта и др. Для обеспечения доступа к информации, хранящейся в базе данных, а также для управления ею, применяют систему управления базами данных (СУБД). СУБД — это комплекс языко­вых и программных средств, предназначенный для создания, ведения и совместного использования БД многими пользователями. Обычно СУБД различают по используемой модели данных. Так, СУБД, базирующиеся на использовании реляционной модели данных, называют ре­ляционными СУБД. Системы управления базами данных помогают отсортировать информацию, а также связать базы данных между собой, при этом предоставив отчет об изменениях и зарегистрированных событиях.

Несмотря на то, что все системы управления базами данных выполняют одну и ту же основную задачу (т.е дают возможность пользователям создавать, редактировать и получать доступ к информации, хранящейся в базах данных), сам процесс выполнения этой задачи варьируется в широких пределах. Кроме того, функции и возможности каждой СУБД могут существенно отличаться. Различные СУБД документированы по-разному: более или менее тщательно. По-разному предоставляется и техническая поддержка.

При сравнении различных популярных баз данных, следует учитывать, удобна ли для пользователя и масштабируема ли данная конкретная СУБД, а также убедиться, что она будет хорошо интегрироваться с другими продуктами, которые уже используются. Кроме того, во время выбора следует принять во внимание стоимость системы и поддержки, предоставляемой разработчиком.

Если речь идёт о выборе СУБД для предприятия, то следует принять во внимание возможность СУБД «расти» вместе с развитием организации. Малому бизнесу могут потребоваться только базовые функции и возможности, а также небольшое количество информации, размещаемой в БД. Но требования могут существенно расти с течением времени, а переход на другую СУБД может стать проблемой.

Существует несколько популярных СУБД, как платных, так и бесплатных, которые можно рекомендовать для применения в организации.

Я выбрал PostgreSQL, потому что является одним из нескольких бесплатных популярных вариантов СУБД, часто используется для ведения баз данных веб-сайтов. Это была одна из первых разработанных систем управления базами данных, поэтому в настоящее время она хорошо развита, и позволяет пользователям управлять как структурированными, так и неструктурированными данными. Может быть использован на большинстве основных платформ, включая Linux. Прекрасно справляется с задачами импорта информации из других типов баз данных с помощью собственного инструментария [9].

Движок БД может быть размещен в ряде сред, в том числе виртуальных, физических и облачных. Самая свежая версия, предлагает обработку больших объемов данных и увеличение числа одновременно работающих пользователей. Безопасность была улучшена благодаря поддержке DBMS\_SESSION.

**Достоинства:**

– является масштабируемым и способен обрабатывать терабайты данных,

– поддерживает формат json,

– существует множество предопределенных функций,

– доступен ряд интерфейсов.

**Недостатки:**

– документация туманна, поэтому, возможно, ответы на некоторые вопросы придется искать в интернете,

– конфигурация может смутить неподготовленного пользователя,

– скорость работы может падать во время проведения пакетных операций или выполнения запросов чтения.

Идеально подходит для организаций с ограниченным бюджетом, но квалифицированными специалистами, когда требуется возможность выбрать свой интерфейс и использовать json.

**2.1 Описание предметной области**

## Магазин «Автозапчасти» занимается продажей новых запасных частей к автомобилям отечественного и иностранного производства.

## Полное фирменное наименование магазина: магазин «Автозапчасти».

## Необходимо разработать инфологическую модель базы данных, позволяющую автоматизировать выполнение основных действий при продаже и всех учетных действий с запчастями.

## Для этого рассмотрим основные этапы, по которым запчасти проходят в магазине:

## 1. Закупаются запчасти, они поступают в магазин.

## 2. Для каждого наименования заводится отдельная строка, в которой указываются:

## 2.1. Код товара по каталогу

## 2.2 Название

## 2.3. Модель автомобиля, к которому подходит запчасть

## 2.5. Cтоимость

## 2.7 Группа. (1-двигатель, 2-система охлаждения, 3-система питания, 4-система зажигания, 5-трансмиссия, 6-подвеска, 7-рулевое управление, 8-тормозная система, 9-кузов, 10-электрооборудование.

## 3. При продаже формируется отдельный список заказов.

## 4. При возврате какой-либо запчасти, ее код, дата продажи, дата возврата вносится в отдельный список.

**2.2 Перечень сущностей**

Магазин автозапчастей содержит следующие сущности:

1. User
2. Order
3. Product
4. OrderProduct
5. Group
6. ProductGroup
   1. **Разработка базы данных для продажи запчастей авто.**

# **3.1Построение логической модели, схемы базы данных**

Логическая модель базы данных - это абстрактное представление структуры и связей данных, независимое от конкретной системы управления базами данных. Для построения логической модели базы данных используется нотация IDEF1X, которая позволяет изображать сущности, атрибуты и отношения между ними с помощью графических символов.

В логической модели базы данных ателье используются следующие сущности, атрибуты и домены:

1. User - это человек, который хочет приобрести подходящую запчасть на авто. Атрибуты и домены клиента:
   * Id - целочисленный идентификатор мастера, является первичным ключом.
   * Phone - строковый атрибут, содержащий номер телефона мастера в формате 11 цифр.
   * Patronomic – строковый атрибут, содержащий отчество клиента, не более 100 символов.
   * Surname - строковый атрибут, содержащий фамилию мастера, не более 100 символов.
   * Name - строковый атрибут, содержащий имя мастера, не более 100 символов.
2. Order - это таблица, где хранятся заказы клиентов. Атрибуты и домены заказа:
   * Id - целочисленный идентификатор заказа, является первичным ключом.
   * UserID - строковый атрибут, целочисленный идентификатор заказа клиента.
   * OrderDate - временной атрибут, содержащий дату и время начала выполнения заказа.
   * Status - логический атрибут, содержащий статус заказа.
3. Product - это товар (запчасть) которые есть в складе и которые продаются. Атрибуты и домены услуги:
   * Id - целочисленный идентификатор услуги, является первичным ключом.
   * Name - строковый атрибут, содержащий название продукта (запчасти), не более 250 символов.
   * Cost - целочисленный атрибут, содержащий цену продукта (запчасти) в рублях.
4. Group - это таблица в которой виды групп запчастей авто. Атрибуты и домены групп запчастей:
   * Id - целочисленный идентификатор заказа-услугис
   * Name - строковый атрибут, содержащий название группы продуктов (запчастей), не более 250 символов.
5. ProductGroup – это табоица где хранятся группа и продукта айди.
   * GroupID - целочисленный идентификатор группы – продукта, является первичным ключом.
   * ProductID - целочисленный идентификатор продукта.
6. OrderProduct – это таблица где хранятся номер заказа и номер запчати.
   * OrderID - целочисленный идентификатор заказа – продукта.
   * ProductID - целочисленный идентификатор продукта.

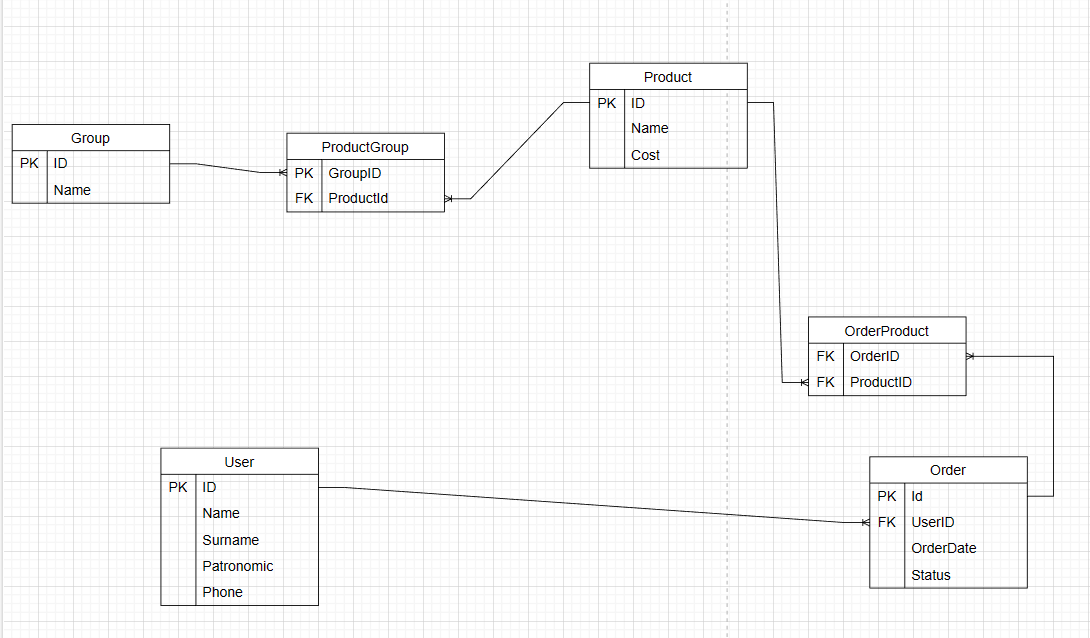


Рисунок 1 - ERD Диаграмма БД Автозапчасти

Между сущностями существуют следующие отношения:

1. Между клиентом и заказом существует отношение один-ко-многим, то есть один клиент может сделать несколько заказов.
2. Между заказом и заказом продукта существует отношение один-ко-многим, то есть один заказ может включать несколько продуктов (запчастей).
3. Между группой и продуктом существует отношение один-ко-многим, то есть одна группа может хранить несколько продуктов (запчастей).

## 4.1 Создание таблиц в Postgresql.

**create** **table** **Groups**

(

Id **INT** **PRIMARY** **KEY** **GENERATED** **ALWAYS** **AS** **IDENTITY**,

**Name** **varchar**(1000)

)

**create** **table** ProductGroup

(

GroupID **INT**,

ProductId **varchar**(1000)

)

**create** **table** Product

(

IDProduct **INT** **PRIMARY** **KEY** **GENERATED** **ALWAYS** **AS** **IDENTITY**,

Product\_name **varchar**(1000),

**Cost** **int**

)

**create** **table** OrderProduct

(

ID **INT** **PRIMARY** **KEY** **GENERATED** **ALWAYS** **AS** **identity**,

OrderID **INT**,

ProductID **int**

)

**create** **table** Orders

(

id\_Order **INT** **PRIMARY** **KEY** **GENERATED** **ALWAYS** **AS** **identity**,

user\_id **INT**,

order\_date **timestamp**,

status **varchar**(100)

)

**create** **table** Users

(

IDClienta **INT** **PRIMARY** **KEY** **GENERATED** **ALWAYS** **AS** **IDENTITY**,

Surname **varchar**(100),

**Name** **varchar**(100),

Patronymic **varchar**(100),

Telephone **varchar**(11)

)

# Написание запросов

Запрос №1. Пропишем запрос для кол-ва зарегистрированных клиентов в базе.

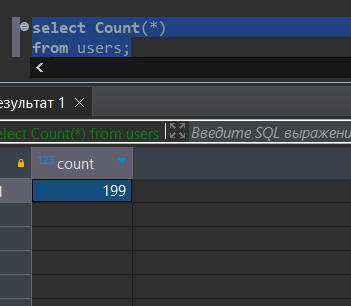


Рисунок №1 запрос 1

Запрос №2 Пропишем запрос для кол-ва заказов в базе.

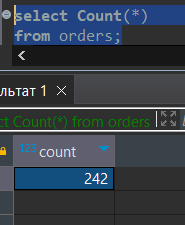


Рисунок №2 запрос 2

# **ЗАКЛЮЧЕНИЕ**

Была разработана база данных для упрощения работы с информацией, внесением и поиском данных санаторно-курортного комплекса.

В настоящее время создано большое количество СУБД, имеющих приблизительно одинаковые возможности. Все они позволяют создавать файлы БД, редактировать их, обновляя записи, удаляя ненужные, добавляя новые. Созданные файлы БД можно упорядочивать по значению определенного индексного реквизита, выполнять поиск информации в базе, формировать отчеты заданной формы. Кроме того, очень важной является функция изменения структуры уже созданного файла базы данных.

Технология баз данных является ведущим направлением организации внутримашинного информационного обеспечения. Развитие технологии баз данных определяется рядом факторов: ростом информационных потребностей пользователей, требованиями эффективного доступа к информации, появлением новых видов массовой памяти, увеличением ее объемов, новыми средствам и возможностями в области коммуникаций и многим другим.

Сейчас базы данных используются почти во всех сферах человеческой деятельности. Так как с помощью электронных баз данных работа персонала организации становится быстрой, качественной и удобной.

Использование баз данных и информационных систем – это составная часть функционирования различных преуспевающих организаций и деятельности современного человека. В связи с этим большую актуальность приобретает освоение принципа построения и эффективного применения соответствующих технологий и программных продуктов.

Поставленная цель курсовой работы достигнута. Разработана база данных «Фирма по продаже запчастей», с помощью которой мы можем создавать:

– таблицы для сохранения данных,

– запросы для поиска и извлечения только необходимых данных,

– формы для просмотра, добавления и изменения данных в таблицах,

– отчеты для анализа и печати данных в определенном формате.

Данная база нормализирована до третьей нормальной формы. Сделаны связи между объектами, использовался мастер подстановок, с помощью которых облегчается заполнение самой базы данных. Разработаны запросы, макросы, отчеты и формы. Многофункциональный интерфейс базы данных помогает упростить работу сотрудникам.