

Министерство цифрового развития, связи и массовых коммуникаций РФ  
Уральский технический институт связи и информатики (филиал) ФГБОУ ВО  
"Сибирский государственный университет телекоммуникаций и  
информатики" в г. Екатеринбурге (УрТИСИ СибГУТИ)



Уральский технический  
институт связи  
и информатики

## ОТЧЕТ

По дисциплине «Сетевое программирование»  
Практическое занятие №4  
«Работа с базами данных»

Выполнила:

студентка группы ПЕ-216

Морос Е.Е.

Проверил:

преподаватель

Бурумбаев Д.И.

## 1 Цель работы:

1.3 Научиться работать с базами данных;

1.4 Закрепить знания по теме «Основы работы с базами данных».

## 2 Перечень оборудования:

2.1 Персональный компьютер;

2.2 СУБД;

2.3 Visual Studio Code.

## 3.Ход работы:

Glide — это платформа для создания мобильных и веб-приложений без необходимости программирования, которая активно использует концепцию работы с базами данных. В центре каждого приложения Glide находится база данных, которая определяет, какие данные обрабатывает и отображает приложение, как эти данные структурированы и как они меняются при взаимодействии пользователя с интерфейсом.

В Glide данные приложения могут храниться в различных источниках, среди которых основными являются Google Sheets, Glide Tables и иногда Airtable или сторонние интеграции через API. Наиболее часто используется связка Glide Tables и/или Google Sheets.

Когда вы создаёте новое приложение в Glide, вас приглашают выбрать или создать источник данных. Это может быть уже существующий Google Sheet либо новая, специально созданная для этого приложения Glide Table. Glide Table — это встроенная реляционная (табличная) база данных, которую хранит и управляет сама платформа Glide. Она работает схожим образом с обычными таблицами (sheet), но доступна только внутри Glide, быстрее реагирует на изменения, а также предоставляет расширенные типы данных и логику вычислений.

Структура таблицы включает строки и столбцы. Столбцы определяют поля данных (например, имя, дата, изображение, ссылка), а строки — конкретные записи (элементы). Столбцы могут иметь разные типы: текст, число, дата, изображение, адрес, e-mail, логика (boolean), кнопки, вычисляемые значения и многие другие, включая формулы (например, автоматические вычисления по ряду столбцов). Glide поддерживает и специальные колонки, которые не просто хранят данные, а выполняют вычисления (например, Rollup — агрегирует значения из связанных записей; Relation — создает связь между таблицами на основе сопоставления значений в столбцах и т.д.).

В Glide возможно создавать реляционные базы: разные таблицы могут быть связаны между собой через так называемые Relation столбцы. Это работает следующим образом: в одной таблице есть, например, столбец с идентификатором пользователя, во второй — список деталей заказа с этим же идентификатором, и через Relation вы получаете список всех заказов, принадлежащих конкретному пользователю. Такие связи позволяют строить сложные структуры данных, использовать дочерние таблицы, агрегировать, фильтровать и автоматически рассчитывать данные по связанным объектам.

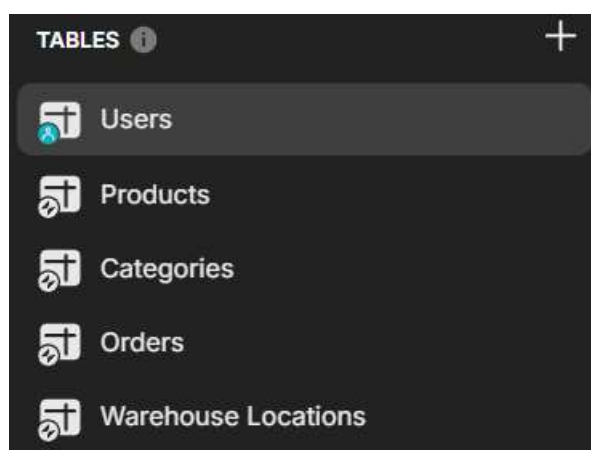
В приложении Glide у каждой таблицы есть два типа обновлений: те, которые делает пользователь через интерфейс приложения, и те, которые происходят в источнике данных непосредственно (например, изменение Google Sheet вручную или через API). Glide синхронизирует эти изменения в обе стороны: пользовательский ввод (создание, редактирование, удаление данных) мгновенно отображается в базе, а если изменить таблицу напрямую, эти изменения отобразятся в приложении. Для Glide Tables эта синхронизация практически мгновенная, для Google Sheets может быть небольшая задержка из-за ограничений Google API.

Важным компонентом является настройка доступа: Glide позволяет ограничивать доступ к данным по ролям и условиям: кто и какие данные может видеть или редактировать. Это реализуется через специальные фильтры, правила доступа, а также защищённые колонки. Glide Table, в отличие от Google Sheets, остаётся полностью частной, если только не дать кому-то к ней явный доступ.

Glide поддерживает вычисляемые столбцы (Computed Columns), различные формулы, условия, интеграции с внешними сервисами по API, а также простую автоматизацию процессов через Actions и Workflows — например, при добавлении новой строки можно запускать автоматическую рассылку письма, создавать связанные записи или выполнять массовые действия с таблицей.

Можно массово загружать данные из CSV, создавать новые таблицы прямо в редакторе, копировать и перемещать записи между таблицами, экспортировать результаты в различные форматы, а также отслеживать статистику изменений.

В итоге, база данных в Glide — это не просто набор таблиц, а мощная, гибкая реляционная система, глубоко интегрированная с логикой приложения и его пользовательским интерфейсом. Пользователь взаимодействует с данными «поверхностно», но под капотом происходит мгновенная синхронизация, логическая обработка, вычисления и автоматизация. Разработчик, используя визуальные инструменты Glide, может проектировать связи между таблицами, выстраивать сложные правила отображения данных, управлять правами и интегрировать внешние источники — всё это без необходимости писать программный код. Glide таким образом превращает даже самого обычного пользователя в «архитектора данных», делая базы данных визуальными, гибкими и легко настраиваемыми под любые задачи.



## Рисунок 1 - Glide Table

В самом Glide таблицы выполняют функцию базы данных приложения. Каждая строка — это отдельная запись (например, задание, пользователь, товар), а каждая колонка — отдельное свойство или атрибут этих записей (например, имя, дата, стоимость, статус). Таблицы позволяют организовать всю необходимую информацию и задавать связи между разными сущностями.

Таблицы бывают двух типов — «Glide Tables» (внутренние таблицы платформы) и внешние таблицы, чаще всего на базе Google Sheets. Glide Tables обладают большей скоростью и удобной синхронизацией, а Google Sheets хороши для интеграции с внешними сервисами и привычны большинству пользователей. Они поддерживают автоматическую синхронизацию данных: изменения, внесённые с мобильного приложения, моментально отображаются в таблице, и наоборот.

Важной частью работы с таблицами в Glide является создание и настройка столбцов. Типы данных могут быть самыми разными: текст, число, дата, изображение, e-mail, ссылка — для каждого типа есть свои интерфейсные и логические возможности. Например, колонка типа «Date/Time» позволяет создать календарный интерфейс, а колонка типа «Image» позволяет отображать картинки, загружаемые пользователем.

Кроме базовых колонок, Glide предлагает «вычисляемые» колонки — например, для подсчёта итогов, конкатенации текста, преобразования формата даты, автоматических флагов и формул. Встроенный редактор формул Glide Data Editor напоминает функции Excel, но адаптирован под работу с приложениями и мобильными интерфейсами.

Glide поддерживает так называемые отношения между таблицами («Relations» или «Links»). Это позволяет создавать сложные системы данных: например, задачник, где одна таблица — пользователи, другая — задачи, и задачи связаны с пользователями по специфичному идентификатору. Через такие связи можно автоматически строить списки, фильтровать записи, присваивать и отображать связанные данные в интерфейсе (например, расшифровывать ID в знакомое имя пользователя или показывать товары, относящиеся к определенной категории).

Все вводимые данные (например, в формах или анкетах) попадают непосредственно в таблицы. Через таблицы определяется и логика фильтрации данных. Glide, основываясь на структуре таблицы, позволяет быстро вводить фильтры, сортировки, ограничения по просмотру — например, разрешать пользователю видеть или менять только свои записи (через специальную колонку e-mail), или показывать задачи только с определённым статусом (через колонку-статус).

Также на уровне таблиц задаются «actions» — автоматические действия, которые происходят при изменении данных или нажатии на кнопки. Например, если запись добавили — отправить уведомление, сменить статус, пересчитать итоговую сумму или автоматически создать связанные записи.

При создании приложения на базе таблицы, Glide автоматически предлагает различные стили представления данных — списки, карточки, галереи, календари, карты и многое другое. Каждый стиль отображения зависит от типа столбцов в таблице: например, если у вас есть колонка-местоположение — можно отобразить

записи на карте; если есть картинки — сделать галерею. При этом каждое изменение в таблице сразу отображается в приложении.

Таблицы в Glide поддерживают совместное редактирование, что удобно для командной работы. Можно задавать права доступа на уровне столбцов и строк: ограничивать видимость полей, запрещать редактирование определённых данных определённым категориям пользователей. Данные в таблицах надёжно защищены, синхронизируются с облаком, поддерживается история изменений.

Glide легко интегрируется с сервисами автоматизации (Zapier, Make/Integromat), что позволяет на основе событий в таблице инициировать внешние процессы: рассылки, уведомления, операции в других сервисах.

Для разработки приложения автоматизации складов обуви было принято решение создать 5 таблиц Glide Tables, которые отвечают за разные аспекты и функции приложения.



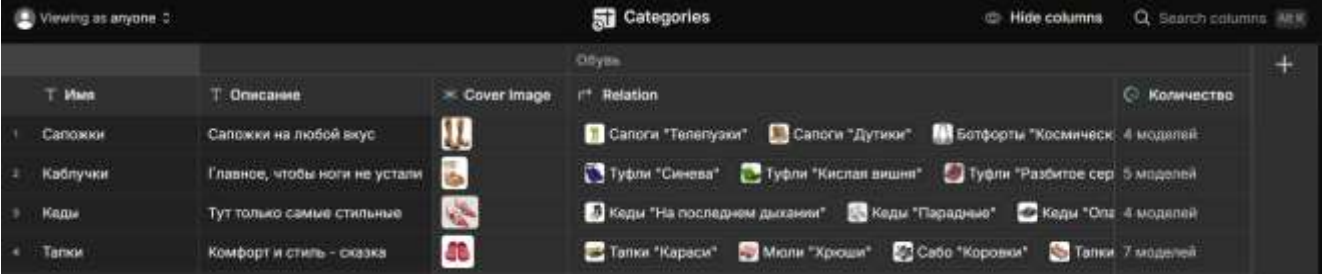
Имя	Email	Фото	Роль
Альма	alma@stelarartech.com	[Profile Picture]	Директор
Элис	elise@stelarartech.com	[Profile Picture]	Менеджер
Хьюго	hugo@quantumdynamics.com	[Profile Picture]	Грузин
Иван	ivan@quantumdynamics.com	[Profile Picture]	Логист
Фатима	fatima@stebulainnovations.com	[Profile Picture]	Аналитик
Лиза	moaros@yandex.ru	[Profile Picture]	Талисман

Рисунок 3.2 – Таблица Users



Имя	Описание	Цена	Склад
Салопы "Телепушки"	Салопы	9 900,00	1000 шт.
Туфли "Синява"	Туфли	9 900,00	1000 шт.
Туфли "Кислая вишня"	Туфли	9 900,00	1000 шт.
Туфли "Разбитое сердце"	Туфли	9 900,00	1000 шт.
Кеды "На последнем дыхании"	Кеды	9 900,00	1000 шт.
Кеды "Парадные"	Кеды	9 900,00	1000 шт.
Кеды "Опа"	Кеды	9 900,00	1000 шт.
Тапки "Караси"	Тапки	9 900,00	1000 шт.
Тапки "Хрюши"	Тапки	9 900,00	1000 шт.
Тапки "Коровки"	Тапки	9 900,00	1000 шт.

Рисунок 3.3 – Таблица Products



Имя	Описание	Cover Image	Relation	Количество
Салопы	Салопы на любой вкус	[Cover Image]	Салопы "Телепушки", Салопы "Дутики", Ботфорты "Космическ"	4 модели
Каблукы	Главное, чтобы ноги не устали	[Cover Image]	Туфли "Синява", Туфли "Кислая вишня", Туфли "Разбитое сердце"	5 моделей
Кеды	Тут только самые стильные	[Cover Image]	Кеды "На последнем дыхании", Кеды "Парадные", Кеды "Опа"	4 модели
Тапки	Комфорт и стиль – сказка	[Cover Image]	Тапки "Караси", Тапки "Хрюши", Тапки "Коровки"	7 моделей

Рисунок 3.4 – Таблица Categories



Дата	Описание	Цена	Статус
2023-10-10	Заказ на доставку	1000	В процессе
2023-10-11	Заказ на доставку	1000	В процессе
2023-10-12	Заказ на доставку	1000	В процессе
2023-10-13	Заказ на доставку	1000	В процессе
2023-10-14	Заказ на доставку	1000	В процессе
2023-10-15	Заказ на доставку	1000	В процессе
2023-10-16	Заказ на доставку	1000	В процессе
2023-10-17	Заказ на доставку	1000	В процессе
2023-10-18	Заказ на доставку	1000	В процессе
2023-10-19	Заказ на доставку	1000	В процессе

Рисунок 3.5 – Таблица Orders

Склады	Склады	Фото	Одежда
Имя	Адрес		Relation Count
Склад № 1	Ул. Ленина, дом 12, кв. 34, Москва, 109012		Сапоги "Телепузики" 4 Туфли "Кислая вы" 4
Центральный склад	Проспект Мира, дом 45, кв. 17, Санкт-Петербург, 191186		Туфли "Синява" 4 Моли "Хрюши" 4
Логистический центр "СН"	Ул. Советская, дом 89, кв. 5, Екатеринбург, 620075		Тапки "Караси" 4 Сабо "Коровки" 4
Склад временного хранения	Ул. Красная, дом 23, кв. 108, Краснодар, 350063		Кеды "На последнем дыхании" 4 Ботфорты 4
Магистральный склад обуви	Ул. Челюскинцев, дом 56, кв. 22, Новосибирск, 630132		Сапоги "Дутики" 4 Кеды "Парадные" 4

Рисунок 3.6 – Таблица Warehouse Locations

## 4 Контрольные вопросы

### 4.1. Пояснить термин «реляционная база данных».

Реляционная база данных — это база данных, организация и управление данными в которой реализованы на основе реляционной модели. В реляционной модели данные хранятся в виде таблиц (отношений), состоящих из строк (записей) и столбцов (полей). Между таблицами могут быть установлены связи через ключевые поля.

### 4.2. Для чего необходима система разграничения прав пользователей?

Система разграничения прав пользователей предназначена для обеспечения безопасности данных, предотвращения несанкционированного доступа, модификации или удаления информации из базы данных. Она позволяет разделить пользователей по ролям и определить, кто и какие действия может выполнять с данными и объектами базы данных.

### 4.3. Какие основные привилегии разрешают доступ к БД?

Основные привилегии, разрешающие доступ к базе данных, включают:

SELECT (просмотр данных)

INSERT (вставка новых записей)

UPDATE (изменение существующих записей)

DELETE (удаление записей)

CREATE (создание новых объектов, например, таблиц)

DROP (удаление объектов)

ALTER (изменение структуры объектов)

EXECUTE (выполнение процедур и функций)

### 4.4. Для чего применяются ключевые поля?

Ключевые поля используются для уникальной идентификации каждой строки в таблице (например, Primary Key) и для создания связей между таблицами (Foreign Key), что обеспечивает целостность и уникальность данных в реляционной базе данных.

### 4.5. Сколько ключевых полей и полей ссылок может содержать таблица?

Первичный ключ (Primary Key) может быть только один, но может состоять из нескольких столбцов (составной ключ).

Внешних ключей (Foreign Keys) в одной таблице может быть несколько — ограничений по количеству, как правило, нет (зависит от СУБД), главное, чтобы структура была логически обоснованной.

UQ-ключ (уникальный ключ, уникальный индекс, unique key) — это ограничение (constraint) в реляционной базе данных, которое гарантирует уникальность значений в одном или нескольких указанных столбцах таблицы.

4.6. Каким способом можно поменять структуру таблицы без применения команды ALTER?

Структуру таблицы можно поменять, создав новую таблицу с нужной структурой, перенесением данных из старой таблицы (с помощью операторов INSERT INTO ... SELECT ...), затем удалением старой таблицы и, по необходимости, переименованием новой таблицы в исходное имя.

4.7. В чем различие операторов INSERT и LOAD DATA?

INSERT используется для добавления одной или нескольких строк данных в таблицу (может быть ручным или пакетным, но относительно медленным).

LOAD DATA (например, LOAD DATA INFILE) — команда быстрой загрузки больших объемов данных из внешнего файла непосредственно в таблицу. Применяется для массового ввода данных и работает быстрее, чем многократный вызов INSERT.