Министерство науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный технический университет имени Н.Э. Баумана»

(национальный исследовательский университет)

Московский техникум космического приборостроения

**ОТЧЕТ О ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

по профессиональному модулю ПМ.11 Разработка, администрирование и защита баз данных

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

Код, специальность 09.02.07 Информационные системы и программирование\_\_\_\_\_\_\_\_\_\_\_\_\_

Место прохождения практики \_\_\_\_\_\_\_\_\_\_\_\_\_\_МТКП МГТУ им. Н. Э. Баумана\_\_\_\_\_\_\_\_\_\_\_\_

(полное название организации)

Выполнил студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Симонян Павел Русланович\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия, имя, отчество)

Курс \_\_\_\_\_\_3\_\_\_\_\_\_\_ Группа \_\_\_\_\_\_\_\_ТИП-51\_\_\_\_\_\_\_\_\_

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата приема зачета « \_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.

Руководитель практики \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (фамилия, имя, отчество)

Москва 2021 г.

# **ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ**

Студента 3 курса Московского техникума космического приборостроения МГТУ

им. Н. Э. Баумана, специальность 09.02.07 Информационные системы и программирование

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Симонян Павел Русланович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия, имя, отчество)

на период производственной практики по ПМ.11 Разработка, администрирование и защита

баз данных

Студент во время практики с 27 октября 2021 г. по 9 ноября 2021 г.

должен:

Знать:

1. Основные положения теории баз данных, хранилищ данных, баз знаний;
2. Основные принципы структуризации и нормализации базы данных;
3. Основные принципы построения концептуальной, логической и физической модели данных;
4. Методы описания схем баз данных в современных системах управления базами данных;
5. Структуры данных систем управления базами данных, общий подход к организации представлений, таблиц, индексов и кластеров; методы организации целостности данных;
6. Способы контроля доступа к данным и управления привилегиями;
7. Основные методы и средства защиты данных в базах данных

Уметь:

1. Работать с современными CASE-средствами проектирования баз данных;
2. Проектировать логическую и физическую схемы базы данных;
3. Создавать хранимые процедуры и триггеры на базах данных;
4. Применять стандартные методы для защиты объектов базы данных;
5. Выполнять стандартные процедуры резервного копирования и мониторинга выполнения этой процедуры;
6. Выполнять процедуру восстановления базы данных и вести мониторинг выполнения этой процедуры;
7. Обеспечивать информационную безопасность на уровне базы данных

Получить практический опыт:

В работе с объектами базы данных в конкретной системе управления базами данных; использовании стандартных методов защиты объектов базы данных; работе с документами отраслевой направленности.

Руководитель практики \_\_\_\_\_\_\_\_\_\_\_\_\_ Осипова Н.М.

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

ОГЛАВЛЕНИЕ

[ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ 2](#_Toc86445133)

[ВВЕДЕНИЕ 5](#_Toc86445134)

[ОСНОВНАЯ ЧАСТЬ 6](#_Toc86445135)

[АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 7](#_Toc86445136)

[РАЗРАБОТКА СТРУКТУРЫ БАЗЫ ДАННЫХ 8](#_Toc86445137)

[ВЫБОР СРЕДСТВ РАЗРАБОТКИ БД 9](#_Toc86445138)

[РЕАЛИЗАЦИЯ БАЗЫ ДАННЫХ В КОНКРЕТНОЙ СУБД 10](#_Toc86445139)

[ЗАКЛЮЧЕНИЕ 11](#_Toc86445140)

[ПРИЛОЖЕНИЕ 12](#_Toc86445141)

[СПИСОК ЛИТЕРАТУРЫ 13](#_Toc86445142)

[ДНЕВНИК ПРАКТИКИ 14](#_Toc86445143)

[АТТЕСТАЦИОННЫЙ ЛИСТ 15](#_Toc86445144)

# **ВВЕДЕНИЕ**

В настоящее время в подразделениях УИ-ВЦ МТКП происходит обновление и поставка нового оборудования, учёт которого необходим для структурированного анализа рабочих единиц техники и составления расписаний занятий. База данных (БД) необходима как централизованная система хранения данных для последующих манипуляций и анализа работы. Разработанная БД позволяет на начальном уровне с помощью запросов и процедур делать выводы о проделанной работе в техникуме, эксплуатации техники и оценке дальнейших перспектив. Целью поставленной задачи является сбор, упорядочивание и своевременная выдача информации, которая в дальнейшем может быть автоматизирована. Разработка данной БД происходит на поверхностном уровне, что позволяет в последующих этапах доработки расширить и усилить охват данной предметной области (ПрО). Основание для разработки строится ещё как техническое задание по производственной практике, проходящей в техникуме.

# **1 ОСНОВНАЯ ЧАСТЬ**

# **1.1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**

БД учёта оборудования в подразделениях УИ-ВЦ МТКП – это совокупность сущностей, взаимодействующих между собой. Главным субъектом данной информационной системы (ИС) является уполномоченный(-ые) сотрудник(-и), вводящие новые данные в базу и проводящие анализ работы на основе запросов и процедур.

Разработанная БД должна обеспечивать четко согласованную работу между всеми объектами, проверять наличие ошибочных данных, уведомляя пользователей об этом, и отслеживать списанные оборудования.

Уполномоченный пользователь обязан вносить новые данные в базу. К таким данным относятся:

* Каждая поставка нового оборудования: установка серийного номера на определённый тип оборудования в соответствующей таблице, указание кабинета, к которому будет прикреплено устройство и тип самого устройства;
* Учёт деталей: указание производителя, модели, типа детали и оборудования, в котором оно находится, или же номер склада, на котором оно находится;
* Учёт эксплуатации кабинетов: номер самого кабинета, ответственный во время пользования и даты со временем начала и конца пользования;
* Данные по сотрудникам: их ФИО и должности;
* Кабинеты: материально ответственный;
* Типы деталей и оборудования с указанием названия;
* Склад: техническое описание того, что конкретно находится в том или ином кабинете-хранилище;

Сущность комплектующих имеет атрибуты склада и оборудования, в котором они находятся. Наличие номера комплектующего подразумевает отсутствие конкретно этой детали на складе, т.е. значение поля со складом равно NULL. В противном случае будет указан номер склада (кабинета), в котором она находится, а в поле с оборудованием будет уже NULL.

Более подробное описание движения данных по БД представлено ниже, на рисунке 1 в виде DFD-модели.

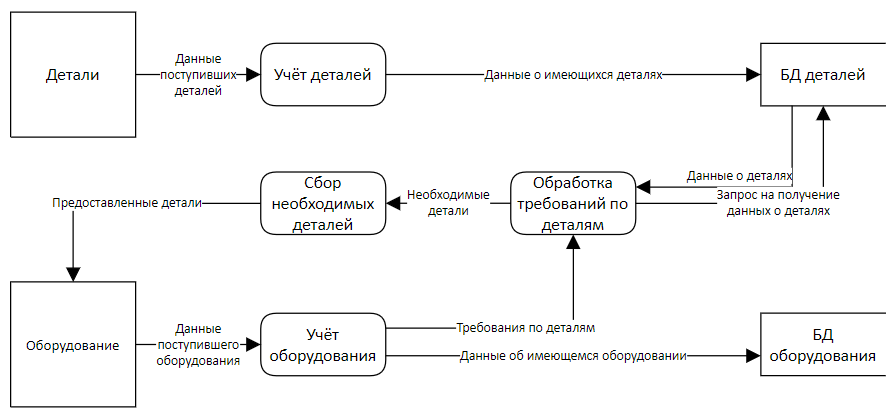


Рисунок 1.1.1. DFD-модель

Новые детали (комплектующие) приносят в ИС данные о себе, которые заносит уполномоченный сотрудник в БД. Эти данные попадают в свой учёт, обращающийся в БД деталей. Информация о деталях может быть использована в обработке требований, т.е. в проверке на совместимость, где идёт запрос на получение данных. Далее уже совместимые комплектующие собираются воедино в оборудование, в качестве которого может выступать как уже собранная техника (мониторы, клавиатуры), так и квинтэссенция им – ПК, вбирающий в себя конкретно в данной БД системный блок, в котором находятся детали.

Данные об оборудовании идут на учёт, где может быть новое регулирование требованиями по деталям, обращаясь к обработке требований; а также информация об имеющемся оборудовании структурированно хранится в БД оборудования.

Ниже, на рисунке 2, представлена модель “Чёрный ящик” и её декомпозиция в нотации IDEF0.

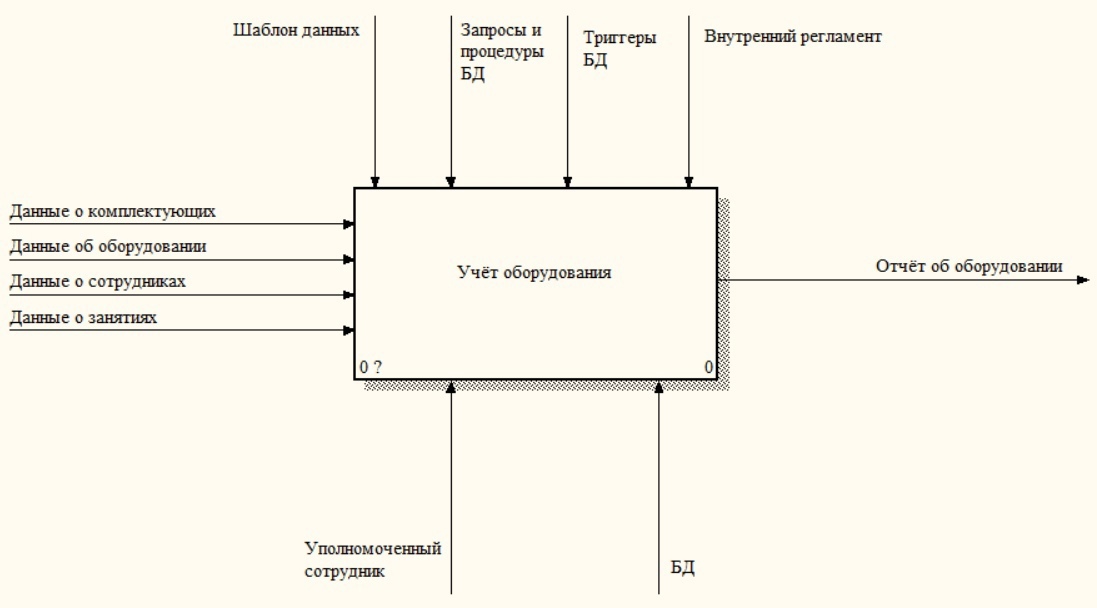


Рисунок 1.1.2. “Чёрный ящик”

Данная модель используется для того, чтобы в общем виде представить процесс работы с БД. На вход ИС идут данные о комплектующих, оборудовании, сотрудниках и занятиях. Приём и работа с данными происходит посредством шаблона данных, внутреннего регламента, запросов, процедур и триггеров БД. Исполнительным лицом является уполномоченный сотрудник, а сами действия происходят непосредственно в самой БД. После всех операций на выходе мы получаем отчет об оборудовании. Подробное описание процесса Учёта оборудования представлено ниже, на рисунке 3.

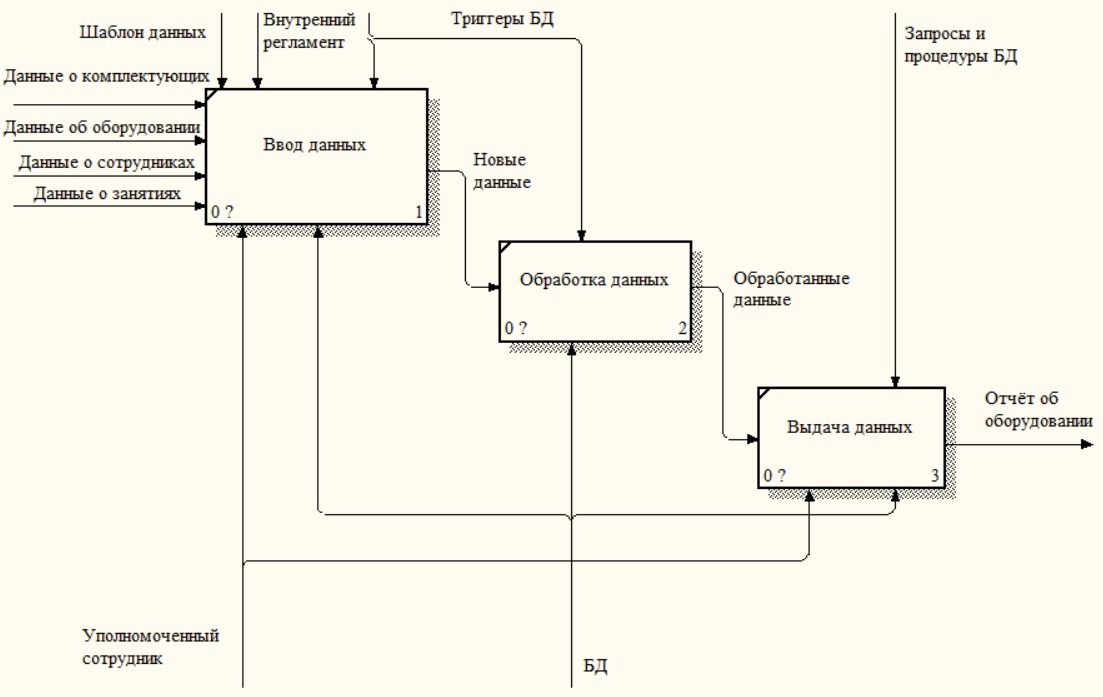


Рисунок 1.1.3. Декомпозиция модели “Чёрный ящик”

Набор действий, описанный на рисунке 3 насчитывает три последовательно идущих процесса - Ввод данных; Обработка данных; Выдача данных. На вход идут ранее описанные данные, которые записываются сотрудником в БД. Данные записываются в определённом формате, что обеспечивает Шаблон данных. К примеру можно отнести запись модели блоков питания, в конце названия которой пишется число, определяющее мощность данного блока питания в ваттах. Внутренний регламент используется как для новых записей о сотрудниках, так и для оборудования с комплектующими, описывающий внутреннюю политику использования иностранной или отечественной техники. Триггеры позволяют вводить данные без ошибок, чтобы при дальнейшей работе не возникало казусов.

Новые данные оказываются на поступление в блок Обработки данных, который регулируют все те же триггеры, но уже для изменения или удаления записей из БД.

Обработанные данные используются в Выдаче данных. Данное действие выполняется посредством ввода сотрудником необходимых запросов и процедур. Выданные данные уже можно использовать для печати, т.к. находятся в уже структурированном виде, понятном для прочтения.

# **1.2 РАЗРАБОТКА СТРУКТУРЫ БАЗЫ ДАННЫХ**

Разработка структуры БД происходит посредством последовательного составления моделей и таблиц, описывающие вид и способ реализации (написания) БД на языке SQL.

Первой на очереди идёт концептуальная модель данных, представленная на рисунке 3

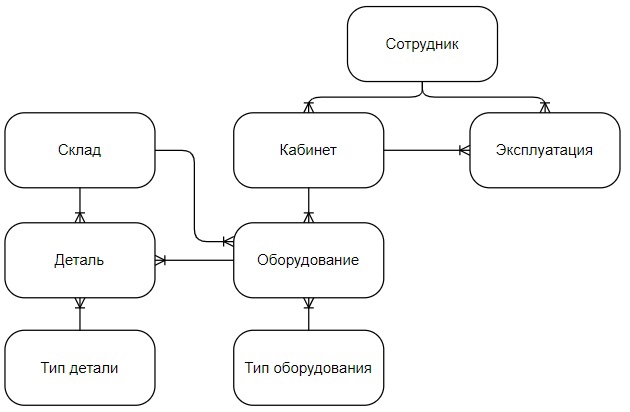


Рисунок 1.2.1. Концептуальная модель данных

Представленные между собой сущности отвечают за одноименные таблицы в третьей нормальной форме БД: один сотрудник может быть указан в нескольких кабинетах для своей работы, учёт которой ведётся в Эксплуатации. В одном кабинете может находиться много оборудования, уже в котором несколько деталей. На одном складе хранятся множество деталей и оборудования. Один тип деталей или оборудования может относиться к ряду деталей и оборудования соответственно. Ниже, на рисунке 4 представлена логическая модель данных с описанием всех атрибутов и их связей, а также внешних и первичных ключей.

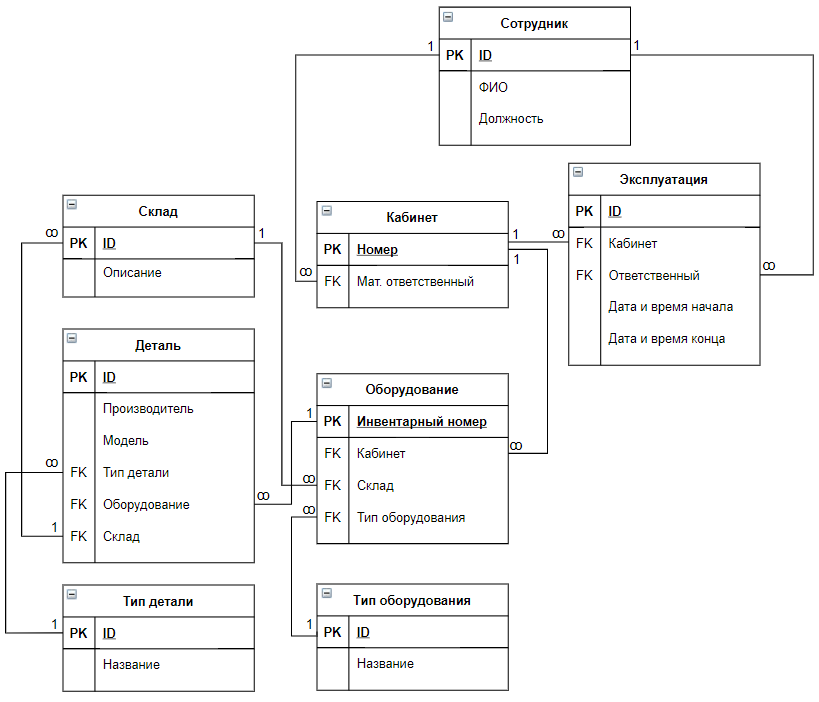


Рисунок 1.2.2. Логическая модель данных

Логическая модель описывает каждую сущность, её атрибуты и их взаимодействия между собой. С помощью этой модели удобно находить связи для выборки необходимых данных из БД, переходя от одной таблицы к другой, обращаясь к внешним ключам.

Ниже представлена физическая модель данных на рисунке 5. В ней происходит описание сущностей, их названия и атрибуты, а также связи между ними.

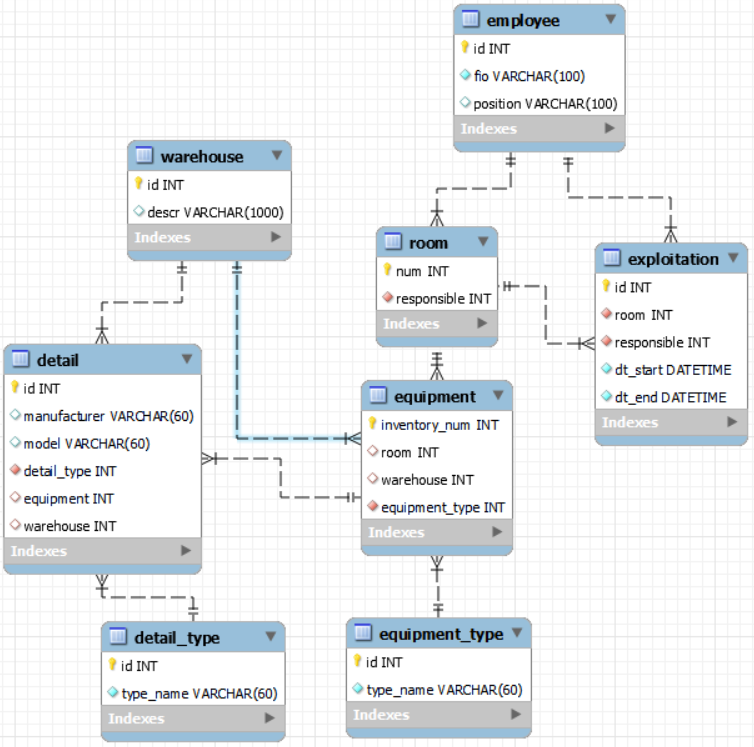


Рисунок 1.2.3. Физическая модель данных

Далее, уже в таблицах, идёт более подробное описание таблиц: название, идентификатор, тип данных, размер и описание

Таблица 1.2.1 – Структура сущности Employee

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Тип данных | Размер | Описание |
| ID | Числовой | - | ID сотрудника |
| Fio | Текстовый | 100 | ФИО сотрудника |
| Position | Текстовый | 100 | Должность |

Таблица 1.2.2 – Структура сущности Room

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Тип данных | Размер | Описание |
| Num | Числовой | - | Номер кабинета |
| Responsible | Числовой | - | Материально ответственный |

Таблица 1.2.3 – Структура сущности Equipment\_type

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Тип данных | Размер | Описание |
| ID | Числовой | - | ID типа оборудования |
| Type\_name | Текстовый | 60 | Название типа оборудования |

Таблица 1.2.4 – Структура сущности Exploitation

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Тип данных | Размер | Описание |
| ID | Числовой | - | ID эксплуатации (занятия) |
| Room | Числовой | - | Номер кабинета |
| Responsible | Числовой | - | Ответственный за эксплуатацию |
| Dt\_start | Дата и время | Авто | Дата и время начала |
| Dt\_end | Дата и время | Авто | Дата и время окончания |

Таблица 1.2.5 – Структура сущности Equipment

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Тип данных | Размер | Описание |
| Inventory\_num | Численный | - | Инвентарный номер оборудования |
| Room | Числовой | - | Номер кабинета (если оборудование в кабинете) |
| Warehouse | Числовой | - | Номер склада (если на складе) |
| Equipment\_type | Числовой | - | ID типа оборудования |
| Position | Текстовый | 80 | Должность |
| Aviary | Числовой | - | Закрепленный вольер (если есть) |

Таблица 1.2.6 – Структура сущности Detail

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Тип данных | Размер | Описание |
| ID | Числовой | - | ID детали или комплектующего |
| Manufacturer | Текстовый | 60 | Производитель |
| Model | Текстовый | 60 | Модель |
| Detail\_type | Числовой | - | ID типа детали |
| Equipment | Числовой | - | ID оборудования (если размещено в оборудовании) |
| Warehouse | Числовой | - | ID склада (если размещено на складе) |

Таблица 1.2.7 – Структура сущности Detail\_type

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Тип данных | Размер | Описание |
| ID | Числовой | - | ID типа детали |
| Type\_name | Текстовый | 60 | Название типа детали |

Таблица 1.2.8 – Структура сущности Warehouse

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Тип данных | Размер | Описание |
| ID | Числовой | - | ID склада |
| Descr | Текстовый | 1000 | Описание схемы прохода |

Запросы по созданию БД приведены в разделе Реализация базы данных в конкретной СУБД, описанный ниже.

# **1.3 ВЫБОР СРЕДСТВ РАЗРАБОТКИ БД**

В практической работе было решено использовать среду разработки и систему управления БД (СУБД) MySQL Workbench как программа с удобным интерфейсом и понятным выводом данных. Обучение по дисциплине технологии разработки и защиты БД (ТРиЗБД) подразумевает охват нескольких СУБД, но в данный момент времени активно изучается MySQL Workbench. Для тестовых запусков и проверок запросов, триггеров и процедур использовался другой бесплатный программный инструмент со своим web-интерфейсом – phpMyAdmin. Подобного рода параллельная разработка позволила минимизировать возможные риски и проблемы в новой для меня СУБД MySQL Workbench. Обе программы поддерживают работу MySQL, отличаясь друг от друга лишь интерфейсом и удобством редактирования полей.

Была также рассмотрена СУБД Microsoft SQL Server Management Studio 18, но по опыту работы, удобству представления и редактирования данных было решено оставить этот вариант.

# **1.4 РЕАЛИЗАЦИЯ БАЗЫ ДАННЫХ В КОНКРЕТНОЙ СУБД**

После подробного описания атрибутов к сущностям мы можем приступать к написанию самой БД на языке SQL через следующие запросы:

Запрос на создание:

CREATE DATABASE Equipment;

USE Equipment;

Запросы на создание таблиц:

CREATE TABLE Employee(

id INT auto\_increment NOT NULL,

PRIMARY KEY (id),

fio VARCHAR(100) NOT NULL,

position VARCHAR(100));

CREATE TABLE Room(

num INT NOT NULL,

PRIMARY KEY (num),

responsible int NOT NULL,

FOREIGN KEY (responsible) REFERENCES Employee(id));

CREATE TABLE Exploitation(

id INT auto\_increment NOT NULL,

PRIMARY KEY (id),

room INT NOT NULL,

FOREIGN KEY (room) REFERENCES Room(num),

responsible int NOT NULL,

FOREIGN KEY (responsible) REFERENCES Employee(id),

dt\_start DATETIME NOT NULL,

dt\_end DATETIME NOT NULL);

CREATE TABLE Warehouse(

id INT auto\_increment NOT NULL,

PRIMARY KEY (id),

descr VARCHAR(1000));

CREATE TABLE Equipment\_type(

id INT auto\_increment NOT NULL,

PRIMARY KEY (id),

type\_name VARCHAR(60) NOT NULL);

CREATE TABLE Equipment(

inventory\_num INT NOT NULL,

PRIMARY KEY (inventory\_num),

room INT,

FOREIGN KEY (room) REFERENCES Room(num),

warehouse INT,

FOREIGN KEY (warehouse) REFERENCES Warehouse(id),

equipment\_type INT NOT NULL,

FOREIGN KEY (equipment\_type) REFERENCES Equipment\_type(id));

CREATE TABLE Detail\_type(

id INT auto\_increment NOT NULL,

PRIMARY KEY (id),

type\_name VARCHAR(60) NOT NULL);

CREATE TABLE Detail(

id INT auto\_increment NOT NULL,

PRIMARY KEY(id),

manufacturer VARCHAR(60),

model VARCHAR(60),

detail\_type INT NOT NULL,

FOREIGN KEY (detail\_type) REFERENCES Detail\_type(id),

equipment INT,

FOREIGN KEY (equipment) REFERENCES Equipment(inventory\_num),

warehouse INT,

FOREIGN KEY (warehouse) REFERENCES Warehouse(id));

Запросы на заполнение таблиц:

USE equipment;

INSERT INTO employee (fio, position) values ('Малыхина Ольга Юрьевна', 'Заведующая учебно-вычислительным центромб преподаватель'),

('Осипова Надежда Михайловна', 'Преподаватель'), ('Митрошенкова Елена Алексеевна', 'Преподаватель'),

('Петренко Людмила Борисовна', 'Преподаватель'), ('Тамахина Ирина Андрониковна', 'Преподаватель'),

('Жилкина Надежда Александровна', 'Преподаватель');

INSERT INTO room values (122, 6), (123, 6),

(124, 6), (210, 6);

# ' YYYY-MM-DD hh:mm:ss '

INSERT INTO exploitation (room, responsible, dt\_start, dt\_end) values

(122, 6, '2021-10-4 9:00:00', '2021-10-4 10:30:00'), (122, 7, '2021-10-4 10:50:00', '2021-10-4 12:10:00'),

(122, 8, '2021-10-4 12:40:00', '2021-10-4 14:00:00'), (122, 8, '2021-10-4 14:30:00', '2021-10-4 16:00:00'),

(122, 9, '2021-10-5 9:00:00', '2021-10-5 10:30:00'), (122, 10, '2021-10-5 10:50:00', '2021-10-5 12:10:00'),

(122, 9, '2021-10-5 12:40:00', '2021-10-5 14:00:00'), (122, 8, '2021-10-5 14:30:00', '2021-10-5 16:00:00'),

(123, 7, '2021-10-4 9:00:00', '2021-10-4 10:30:00'), (123, 6, '2021-10-4 10:50:00', '2021-10-4 12:10:00'),

(123, 7, '2021-10-4 12:40:00', '2021-10-4 14:00:00'), (123, 10, '2021-10-4 14:30:00', '2021-10-4 16:00:00'),

(123, 6, '2021-10-5 9:00:00', '2021-10-4 10:30:00'), (123, 8, '2021-10-5 10:50:00', '2021-10-5 12:10:00'),

(123, 11, '2021-10-5 12:40:00', '2021-10-4 14:00:00'), (123, 11, '2021-10-5 14:30:00', '2021-10-5 16:00:00');

INSERT INTO equipment\_type (type\_name) values ('Системный блок'),

('Монитор'), ('Компьютерная мышь'), ('Клавиатура'), ('Смарт-доска'),

('Ноутбук'), ('Принтер'), ('Сканер'), ('Проектор'), ('МФУ'), #Многофункциональное устройство

('ИБП'); #Источник бесперебойного питания

INSERT INTO warehouse (descr) values('Склад в подвале'),

('Склад на пятом этаже'), ('Склад на первом этаже');

INSERT INTO detail\_type (type\_name) values ('Картридж для принтера'),

('Видеокарта'), ('Материнская плата'), ('Процессор'), ('Кулер'),

('HDD'), ('SSD'), ('Блок питания');

#Системные блоки

INSERT INTO equipment values (1, 122, NULL, 1),

(2, 122, NULL, 1), (3, 122, NULL, 1), (4, 122, NULL, 1),

(5, 122, NULL, 1), (6, 122, NULL, 1), (7, 122, NULL, 1),

(8, 122, NULL, 1), (9, 122, NULL, 1), (10, 122, NULL, 1);

INSERT INTO equipment values (11, 123, NULL, 1),

(12, 123, NULL, 1), (13, 123, NULL, 1), (14, 123, NULL, 1),

(15, 123, NULL, 1), (16, 123, NULL, 1), (17, 123, NULL, 1),

(18, 123, NULL, 1), (19, 123, NULL, 1), (20, 123, NULL, 1);

#Мониторы

INSERT INTO equipment values (21, 122, NULL, 2),

(22, 122, NULL, 2), (23, 122, NULL, 2), (24, 122, NULL, 2),

(25, 122, NULL, 2), (26, 122, NULL, 2), (27, 122, NULL, 2),

(28, 122, NULL, 2), (29, 122, NULL, 2), (30, 122, NULL, 2);

INSERT INTO equipment values (31, 123, NULL, 2),

(32, 123, NULL, 2), (33, 123, NULL, 2), (34, 123, NULL, 2),

(35, 123, NULL, 2), (36, 123, NULL, 2), (37, 123, NULL, 2),

(38, 123, NULL, 2), (39, 123, NULL, 2), (40, 123, NULL, 2);

#Компьютерные мыши

INSERT INTO equipment values (41, 122, NULL, 3),

(42, 122, NULL, 3), (43, 122, NULL, 3), (44, 122, NULL, 3),

(45, 122, NULL, 3), (46, 122, NULL, 3), (47, 122, NULL, 3),

(48, 122, NULL, 3), (49, 122, NULL, 3), (50, 122, NULL, 3);

INSERT INTO equipment values (51, 123, NULL, 2),

(52, 123, NULL, 3), (53, 123, NULL, 3), (54, 123, NULL, 3),

(55, 123, NULL, 3), (56, 123, NULL, 3), (57, 123, NULL, 3),

(58, 123, NULL, 3), (59, 123, NULL, 3), (60, 123, NULL, 3);

#Клавиатуры

INSERT INTO equipment values (61, 122, NULL, 4),

(62, 122, NULL, 4), (63, 122, NULL, 4), (64, 122, NULL, 4),

(65, 122, NULL, 4), (66, 122, NULL, 4), (67, 122, NULL, 4),

(68, 122, NULL, 4), (69, 122, NULL, 4), (70, 122, NULL, 4);

INSERT INTO equipment values (71, 123, NULL, 4),

(72, 123, NULL, 4), (73, 123, NULL, 4), (74, 123, NULL, 4),

(75, 123, NULL, 4), (76, 123, NULL, 4), (77, 123, NULL, 4),

(78, 123, NULL, 4), (79, 123, NULL, 4), (80, 123, NULL, 4);

#Материнские платы

INSERT INTO detail (manufacturer, model, detail\_type, equipment) values

('GIGABYTE', 'GA-A320M-H', 3, 1), ('GIGABYTE', 'GA-A320M-H', 3, 2), ('GIGABYTE', 'GA-A320M-H', 3, 3),

('GIGABYTE', 'GA-A320M-H', 3, 4), ('GIGABYTE', 'GA-A320M-H', 3, 5), ('GIGABYTE', 'GA-A320M-H', 3, 6),

('GIGABYTE', 'GA-A320M-H', 3, 7), ('GIGABYTE', 'GA-A320M-H', 3, 8), ('GIGABYTE', 'GA-A320M-H', 3, 9),

('GIGABYTE', 'GA-A320M-H', 3, 10), ('GIGABYTE', 'GA-A320M-H', 3, 11), ('GIGABYTE', 'GA-A320M-H', 3, 12),

('GIGABYTE', 'GA-A320M-H', 3, 13), ('GIGABYTE', 'GA-A320M-H', 3, 14), ('GIGABYTE', 'GA-A320M-H', 3, 15),

('GIGABYTE', 'GA-A320M-H', 3, 16), ('GIGABYTE', 'GA-A320M-H', 3, 17), ('GIGABYTE', 'GA-A320M-H', 3, 18),

('GIGABYTE', 'GA-A320M-H', 3, 19), ('GIGABYTE', 'GA-A320M-H', 3, 20);

#Процессор

INSERT INTO detail (manufacturer, model, detail\_type, equipment) values

('Intel', 'Core i5 9400F', 4, 1), ('Intel', 'Core i5 9400F', 4, 2), ('Intel', 'Core i5 9400F', 4, 3),

('Intel', 'Core i5 9400F', 4, 4), ('Intel', 'Core i5 9400F', 4, 5), ('Intel', 'Core i5 9400F', 4, 6),

('Intel', 'Core i5 9400F', 4, 7), ('Intel', 'Core i5 9400F', 4, 8), ('Intel', 'Core i5 9400F', 4, 9),

('Intel', 'Core i5 9400F', 4, 10), ('Intel', 'Core i5 9400F', 4, 11), ('Intel', 'Core i5 9400F', 4, 12),

('Intel', 'Core i5 9400F', 4, 13), ('Intel', 'Core i5 9400F', 4, 14), ('Intel', 'Core i5 9400F', 4, 15),

('Intel', 'Core i5 9400F', 4, 16), ('Intel', 'Core i5 9400F', 4, 17), ('Intel', 'Core i5 9400F', 4, 18),

('Intel', 'Core i5 9400F', 4, 19), ('Intel', 'Core i5 9400F', 4, 20);

#Видеокарта

INSERT INTO detail (manufacturer, model, detail\_type, equipment) values

('MSI NVIDIA', 'GeForce GT 730', 2, 1), ('MSI NVIDIA', 'GeForce GT 730', 2, 2), ('MSI NVIDIA', 'GeForce GT 730', 2, 3),

('MSI NVIDIA', 'GeForce GT 730', 2, 4), ('MSI NVIDIA', 'GeForce GT 730', 2, 5), ('MSI NVIDIA', 'GeForce GT 730', 2, 6),

('MSI NVIDIA', 'GeForce GT 730', 2, 7), ('MSI NVIDIA', 'GeForce GT 730', 2, 8), ('MSI NVIDIA', 'GeForce GT 730', 2, 9),

('MSI NVIDIA', 'GeForce GT 730', 2, 10), ('MSI NVIDIA', 'GeForce GT 730', 2, 11), ('MSI NVIDIA', 'GeForce GT 730', 2, 12),

('MSI NVIDIA', 'GeForce GT 730', 2, 13), ('MSI NVIDIA', 'GeForce GT 730', 2, 14), ('MSI NVIDIA', 'GeForce GT 730', 2, 15),

('MSI NVIDIA', 'GeForce GT 730', 2, 16), ('MSI NVIDIA', 'GeForce GT 730', 2, 17), ('MSI NVIDIA', 'GeForce GT 730', 2, 18),

('MSI NVIDIA', 'GeForce GT 730', 2, 19), ('MSI NVIDIA', 'GeForce GT 730', 2, 20);

#HDD

INSERT INTO detail (manufacturer, model, detail\_type, equipment) values

('WD', '500GB WD5000AZLX', 6, 1), ('WD', '500GB WD5000AZLX', 6, 2), ('WD', '500GB WD5000AZLX', 6, 3),

('WD', '500GB WD5000AZLX', 6, 4), ('WD', '500GB WD5000AZLX', 6, 5), ('WD', '500GB WD5000AZLX', 6, 6),

('WD', '500GB WD5000AZLX', 6, 7), ('WD', '500GB WD5000AZLX', 6, 8), ('WD', '500GB WD5000AZLX', 6, 9),

('WD', '500GB WD5000AZLX', 6, 10), ('WD', '500GB WD5000AZLX', 6, 11), ('WD', '500GB WD5000AZLX', 6, 12),

('WD', '500GB WD5000AZLX', 6, 13), ('WD', '500GB WD5000AZLX', 6, 14), ('WD', '500GB WD5000AZLX', 6, 15),

('WD', '500GB WD5000AZLX', 6, 16), ('WD', '500GB WD5000AZLX', 6, 17), ('WD', '500GB WD5000AZLX', 6, 18),

('WD', '500GB WD5000AZLX', 6, 19), ('WD', '500GB WD5000AZLX', 6, 20);

#Блок питания

INSERT INTO detail (manufacturer, model, detail\_type, equipment) values

('Aerocool', 'VX PLUS 700', 8, 1), ('Aerocool', 'VX PLUS 700', 8, 2), ('Aerocool', 'VX PLUS 700', 8, 3),

('Aerocool', 'VX PLUS 700', 8, 4), ('Aerocool', 'VX PLUS 700', 8, 5), ('Aerocool', 'VX PLUS 700', 8, 6),

('Aerocool', 'VX PLUS 700', 8, 7), ('Aerocool', 'VX PLUS 700', 8, 8), ('Aerocool', 'VX PLUS 700', 8, 9),

('Aerocool', 'VX PLUS 700', 8, 10), ('Aerocool', 'VX PLUS 700', 8, 11), ('Aerocool', 'VX PLUS 700', 8, 12),

('Aerocool', 'VX PLUS 700', 8, 13), ('Aerocool', 'VX PLUS 700', 8, 14), ('Aerocool', 'VX PLUS 700', 8, 15),

('Aerocool', 'VX PLUS 700', 8, 16), ('Aerocool', 'VX PLUS 700', 8, 17), ('Aerocool', 'VX PLUS 700', 8, 18),

('Aerocool', 'VX PLUS 700', 8, 19), ('Aerocool', 'VX PLUS 700', 8, 20);

Список запросов для БД:

1. Выборка всех сотрудников

SELECT id, fio AS 'ФИО', position AS 'Должность'

FROM employee;

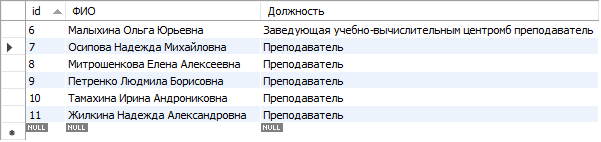


Рисунок 1.4.1. Результат выполнения выборки всех сотрудников

1. Выборка кабинета и ответственного за него сотрудника

SELECT rm.num AS 'Кабинет', ee.fio AS 'Материально ответственный'

FROM room rm INNER JOIN employee ee

ON rm.responsible = ee.id;

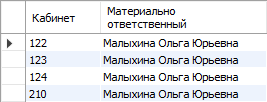


Рисунок 1.4.2. Результат выполнения выборки кабинета и его мат. ответственного

1. Выборка кабинетов, в которых есть мониторы

SELECT num AS 'Кабинеты', COUNT(et.equipment\_type) AS 'Кол-во мониторов'

FROM equipment et INNER JOIN room rm

ON et.room = rm.num

WHERE et.equipment\_type = 2

GROUP BY num;

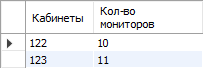


Рисунок 1.4.3. Результат выполнения выборки кабинетов, в которых есть мониторы

1. Выборка кабинета, в котором наибольшее количество системных блоков

SELECT room AS 'Кабинет', max\_value AS 'Кол-во СБ'

FROM (SELECT room, COUNT(equipment\_type) as max\_value

FROM equipment

WHERE equipment\_type = 1

GROUP BY room) t1

ORDER BY max\_value DESC LIMIT 1;



Рисунок 1.4.4. Результат выполнения выборки запроса 4

1. Все задействованные на данный момент видеокарты

SELECT id, manufacturer AS 'Производитель', model AS 'Модель', equipment AS 'Системный блок'

FROM detail

WHERE warehouse is null AND equipment is not null AND detail\_type = 2;

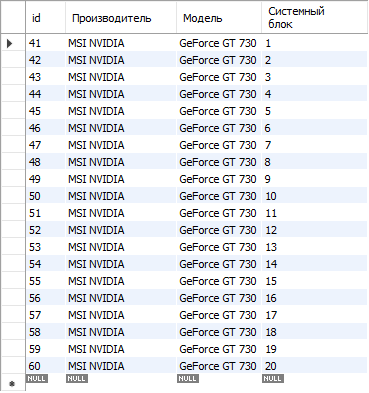


Рисунок 1.4.5. Результат выполнения выборки на все задействованные видеокарты

Список триггеров:

1. Добавление кабинетов (номер должен быть трёхзначным)

use equipment;

delimiter //

create trigger room\_insert

before insert on room

for each row

begin

if new.num not like '\_\_\_' then

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Помещение должно иметь трёхзначный номер!';

END IF;

end//

delimiter ;

insert into room values (119, 6);



Рисунок 1.4.6. Результат выполнения запроса для триггера на добавление

1. Изменение времени эксплуатации

use equipment;

delimiter //

create trigger exploitation\_update

before update on exploitation

for each row

begin

if new.dt\_start > new.dt\_end then

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Дата начала эксплуатации не может быть позже даты окончания эксплуатации';

END IF;

end//

delimiter ;

insert into exploitation values (1001, 122, 6, '2021-10-4 9:00:00', '2021-10-4 10:30:00');

update exploitation set dt\_start = '2021-10-4 10:30:00', dt\_end = '2021-10-4 9:00:00' where id = 1001;

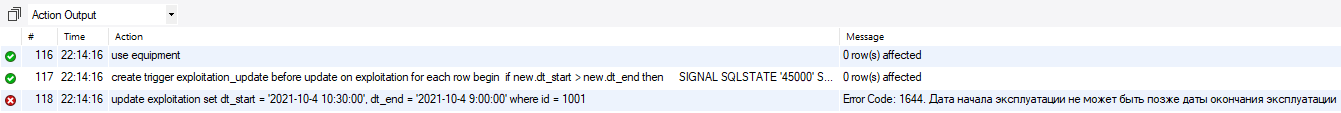


Рисунок 1.4.7. Результат выполнения запроса для триггера на изменение

1. Списание оборудования и его деталей

use equipment;

drop trigger equipment\_delete;

delimiter //

create trigger equipment\_delete

before delete on equipment

for each row

begin

delete from detail where equipment = old.inventory\_num;

end//

delimiter ;

insert into equipment values (1000, 122, NULL, 1);

insert into detail (manufacturer, model, detail\_type, equipment) values ('GIGABYTE', 'GA-A320M-H', 3, 1000);

delete from equipment where inventory\_num = 1000;

select \* from detail;

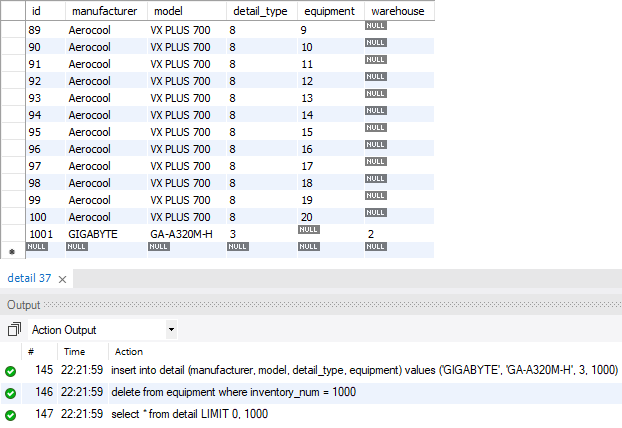


Рисунок 1.4.8. Результат выполнения запроса для триггера списания

Хранимые процедуры:

1. Выборка оборудования по типу. Результат выполнения представлен на рисунке

use equipment;

DELIMITER //

create procedure `equipment\_by\_type` (in eqtype varchar(60))

begin

select Equipment.inventory\_num as 'Инвентарный номер', Equipment.room as 'Номер помещения', Equipment.warehouse as 'Номер склада'

from Equipment\_type inner join Equipment on

Equipment\_type.id = Equipment.equipment\_type

where Equipment\_type.type\_name = eqtype;

end//

call equipment\_by\_type('Системный блок');

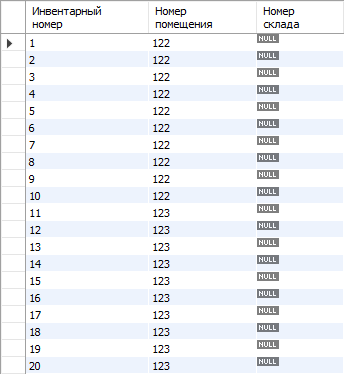


Рисунок 1.4.9. Результат выполнения процедуры выборки оборудования по типу

1. Вывод количества оборудования по его типу

use equipment;

DELIMITER //

create procedure `number\_of\_equipment\_by\_type` (in eqtype varchar(60))

begin

select Equipment\_type.type\_name as 'Наименование оборудования',

count(Equipment.inventory\_num) as 'Количество оборудования'

from Equipment inner join Equipment\_type on

Equipment\_type.id = Equipment.equipment\_type

where Equipment\_type.type\_name = eqtype

group by Equipment.equipment\_type ;

end//

call number\_of\_equipment\_by\_type('Системный блок');



Рисунок 1.4.10. Результат выполнения процедуры количества оборудования по его типу

1. Детали по их типу

use equipment;

drop procedure details\_by\_type;

DELIMITER //

create procedure `details\_by\_type` (in dttype varchar(60))

begin

select Detail.id as 'ID детали', Detail.manufacturer as 'Производитель', Detail.model as 'Модель',

Detail.equipment as 'Инвентарный номер оборудования', Detail.warehouse as 'Номер склада'

from Detail\_type inner join Detail on

Detail\_type.id = Detail.detail\_type

where Detail\_type.type\_name = dttype;

end//

call details\_by\_type('Видеокарта');

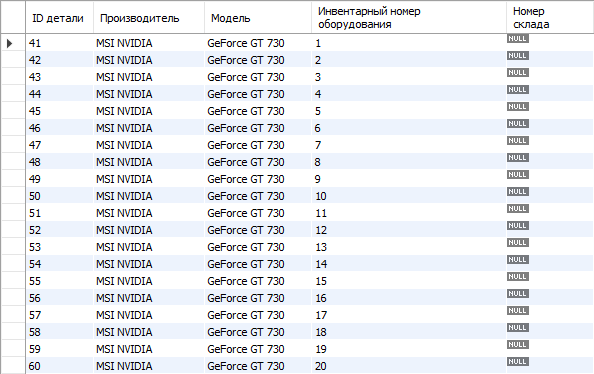


Рисунок 1.4.11. Результат выполнения процедуры выборки деталей по их типу

Пользователи в данной БД определены на трёх уровнях. К ним также относятся три уровня доступа по ролям:

1. Учитель – имеет доступ к чтению информации о занятиях, доступном оборудовании и их характеристиках для планирования образовательной деятельности. Учителя не имеют доступа к информации о складах.
2. Инженер – для обновления информации о техническом обеспечении, ему предоставлено право на редактирование таблиц Equipment (оборудование) и Detail (детали / комплектующие). Остальные таблицы инженер редактировать не может, но имеет доступ к чтению.
3. Заведующий – имеет доступ к редактированию любой таблицы, так как отвечает за передачу ответственности за кабинеты с оборудованием, должен иметь возможность вносить правки в описание складов, информацию о сотрудниках, кабинетах, типах деталей и оборудования, а также их составе в случае, когда это необходимо. Заведующий имеет роль администратора базы данных, но не имеет возможности редактировать ее структуру.

Определение прав пользователей:

1. Роль “Учитель”

use equipment;

CREATE ROLE 'Teacher'@'localhost';

GRANT SELECT ON equipment.equipment TO 'Teacher'@'localhost';

GRANT SELECT ON equipment.detail TO 'Teacher'@'localhost';

GRANT SELECT ON equipment.detail\_type TO 'Teacher'@'localhost';

GRANT SELECT ON equipment.employee TO 'Teacher'@'localhost';

GRANT SELECT ON equipment.equipment\_type TO 'Teacher'@'localhost';

GRANT SELECT ON equipment.room TO 'Teacher'@'localhost';

GRANT SELECT ON equipment.exploitation TO 'Teacher'@'localhost';

select \* from mysql.default\_roles;

select current\_role()

CREATE USER 'Anakonda'@'localhost' IDENTIFIED BY 'Anakonda';

GRANT 'Teacher'@'localhost' TO 'Anakonda'@'localhost';

SET DEFAULT ROLE 'Teacher'@'localhost' TO 'Anakonda'@'localhost';

flush privileges;

show grants for 'Anakonda'@'localhost';

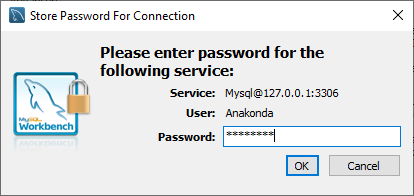


Рисунок 1.4.12. Окно аутентификации для пользователя Anakonda

use equipment;

select \* from warehouse;

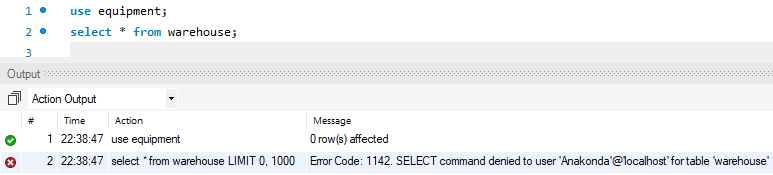


Рисунок 1.4.13. Результат выполнения запроса за пользователя Anakonda

1. Роль “Инженер”

use equipment;

CREATE ROLE 'Engineer'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.equipment TO 'Engineer'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.detail TO 'Engineer'@'localhost';

GRANT SELECT ON equipment.warehouse TO 'Engineer'@'localhost';

GRANT SELECT ON equipment.detail\_type TO 'Engineer'@'localhost';

GRANT SELECT ON equipment.employee TO 'Engineer'@'localhost';

GRANT SELECT ON equipment.equipment\_type TO 'Engineer'@'localhost';

GRANT SELECT ON equipment.room TO 'Engineer'@'localhost';

GRANT SELECT ON equipment.exploitation TO 'Engineer'@'localhost';

select \* from mysql.default\_roles;

select current\_role()

CREATE USER 'UserEngineer'@'localhost' IDENTIFIED BY 'UserEngineer';

GRANT 'Engineer'@'localhost' TO 'UserEngineer'@'localhost';

SET DEFAULT ROLE 'Engineer'@'localhost' TO 'UserEngineer'@'localhost';

flush privileges;

show grants for 'UserEngineer'@'localhost';

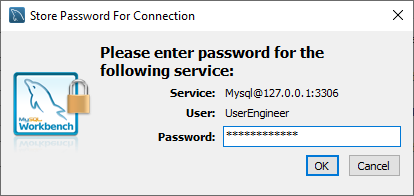


Рисунок 1.4.14. Окно аутентификации для пользователя UserEngineer

use equipment;

insert into room values (119, 6);

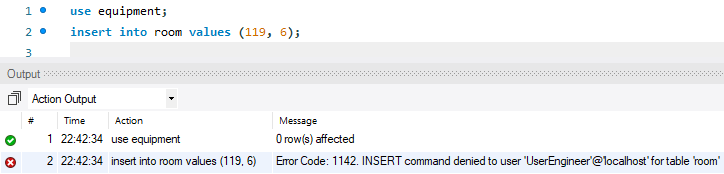


Рисунок 1.4.15. Результат выполнения запроса за пользователя UserEngineer

1. Роль “Заведующий”

use equipment;

CREATE ROLE 'Manager'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.equipment TO 'Manager'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.detail TO 'Manager'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.warehouse TO 'Manager'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.detail\_type TO 'Manager'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.employee TO 'Manager'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.equipment\_type TO 'Manager'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.room TO 'Manager'@'localhost';

GRANT SELECT, DELETE, UPDATE, INSERT ON equipment.exploitation TO 'Manager'@'localhost';

select \* from mysql.default\_roles;

select current\_role()

CREATE USER 'UserManager'@'localhost' IDENTIFIED BY 'UserManager';

GRANT 'Manager'@'localhost' TO 'UserManager'@'localhost';

SET DEFAULT ROLE 'Manager'@'localhost' TO 'UserManager'@'localhost';

flush privileges;

show grants for 'UserManager'@'localhost';

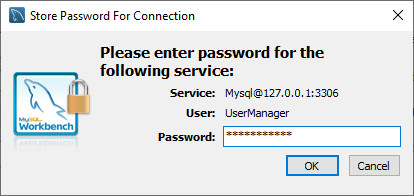


Рисунок 1.4.16. Окно аутентификации пользователя UserManager

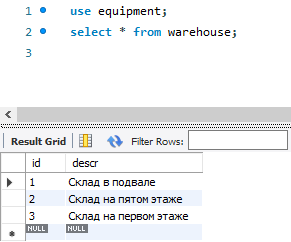


Рисунок 1.4.17. Результат выполнения запроса выборки за пользователя UserManager

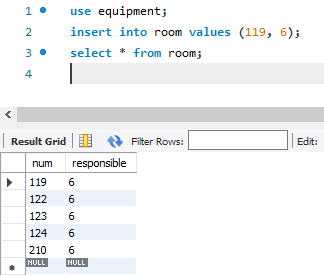


Рисунок 1.4.18. Результат выполнения запроса с добавлением за пользователя UserManager

# **2. ЗАКЛЮЧЕНИЕ**

По итогу проделанной работы была создана БД по учёту оборудования в подразделениях УИ-ВЦ МТКП. Закреплены навыки анализа предметной области, проектирования и создания БД, определение запросов, процедур, триггеров и прав пользователей. Освоена работа в СУБД MySQL Workbench.

# **СПИСОК ЛИТЕРАТУРЫ**

1. Карвин, Билл Программирование баз данных SQL. Типичные ошибки и их устранение / Билл Карвин. - М.: Рид Групп, 2018. - 336 c
2. Дунаев, В. В. Базы данных. Язык SQL для студента / В.В. Дунаев. - М.: БХВ-Петербург, 2017. - 288 c.
3. Кригель, А. SQL. Библия пользователя / А. Кригель. - М.: Диалектика / Вильямс, 2016. - 0 c.

# **АТТЕСТАЦИОННЫЙ ЛИСТ**

**ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

студента Московского техникума космического приборостроения МГТУ им.Н.Э.Баумана\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Ф.И.О. студента)

Группа **ТИП-51**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Специальность **09.02.07 Информационные системы и программирование Квалификация - Программист** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(код, наименование специальности)

прошел (ла) производственную практику **ПП 11 ПМ.Разработка, администрирование и защита баз данных \_\_\_\_\_\_\_\_\_**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(наименование практики)

по профессиональному модулю **ПП 11 ПМ.11 Разработка, администрирование и защита баз данных** в объеме 72 часов\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(наименование профессионального модуля)

с «27» октября 2021 года по «09» ноября 2021 года

на предприятии (организации) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (юридический адрес предприятия (организации)

**Виды и качество работ в период производственной практики**

|  |  |
| --- | --- |
| Виды работ, выполненные студентом во время практики, согласно программы производственной практики | Результат (по 5-ти бальной шкале) |
| Разработка, администрирование и защита баз данных и соответствующими ему компетенциями и необходимыми умениями и опытом практической работы по профессии |  |

**В ходе производственной практики студентом освоены следующие профессиональные компетенции**

|  |  |
| --- | --- |
| Код и название профессиональной компетенции | Результат освоения (освоена/не освоена) |
| ПК 11.1.Осуществлять сбор, обработку и анализ информации для проектирования баз данных. | **Освоена** |
| ПК 11.2.Проектировать базу данных на основе анализа предметной области | **Освоена** |
| ПК 11.3.Разрабатывать объекты базы данных в соответствии с результатами анализа предметной области. | **Освоена** |
| ПК 11.4.Реализовывать базу данных в конкретной системе управления базами данных. | **Освоена** |
| ПК 11.5.Администрировать базы данных. | **Освоена** |
| ПК 11.6. Защищать информацию в базе данных с использованием технологии защиты информации. | **Освоена** |

**Характеристика профессиональной деятельности обучающегося во время производственной практики:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Рекомендуемая оценка по практике\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Руководитель практики от предприятия (организации)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.М. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Должность Подпись Ф.И.О. руководителя практики

**Итоговая оценка по практике\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Руководитель практики от образовательного учреждения

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

Должность Подпись Ф.И.О. руководителя практики