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 1.1.1 в виде DFD-модели.

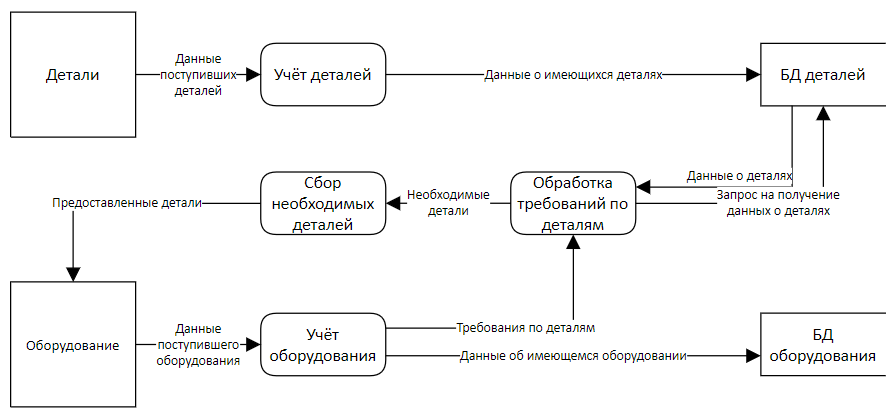


Рисунок 1.1.1. DFD-модель

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 1.1.2, представлена модель “Чёрный ящик” и её декомпозиция в нотации IDEF0.

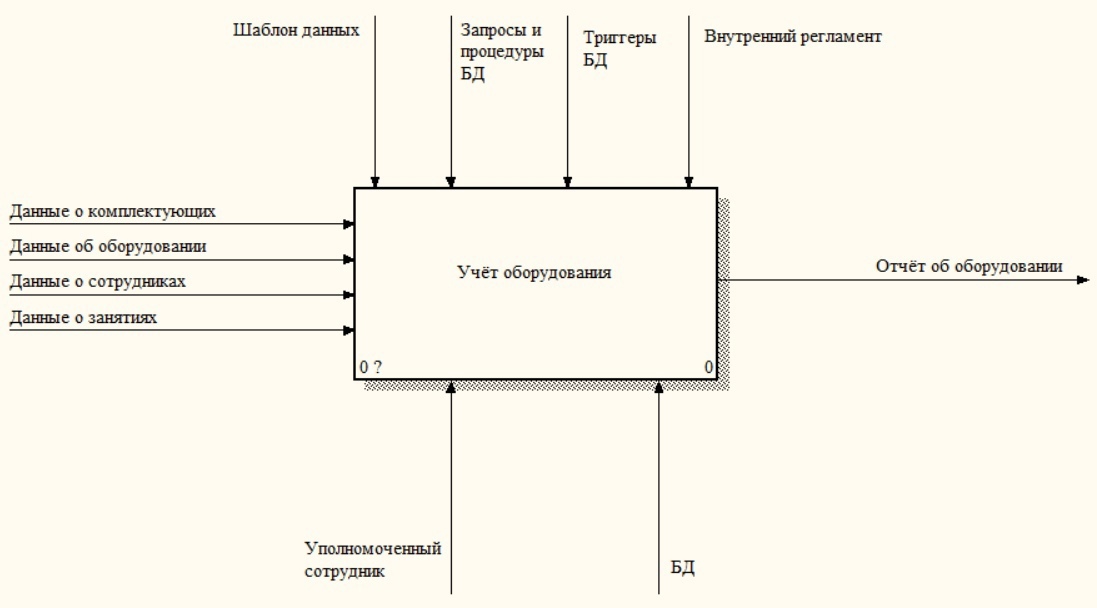


Рисунок 1.1.2. “Чёрный ящик”

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 1.1.3.

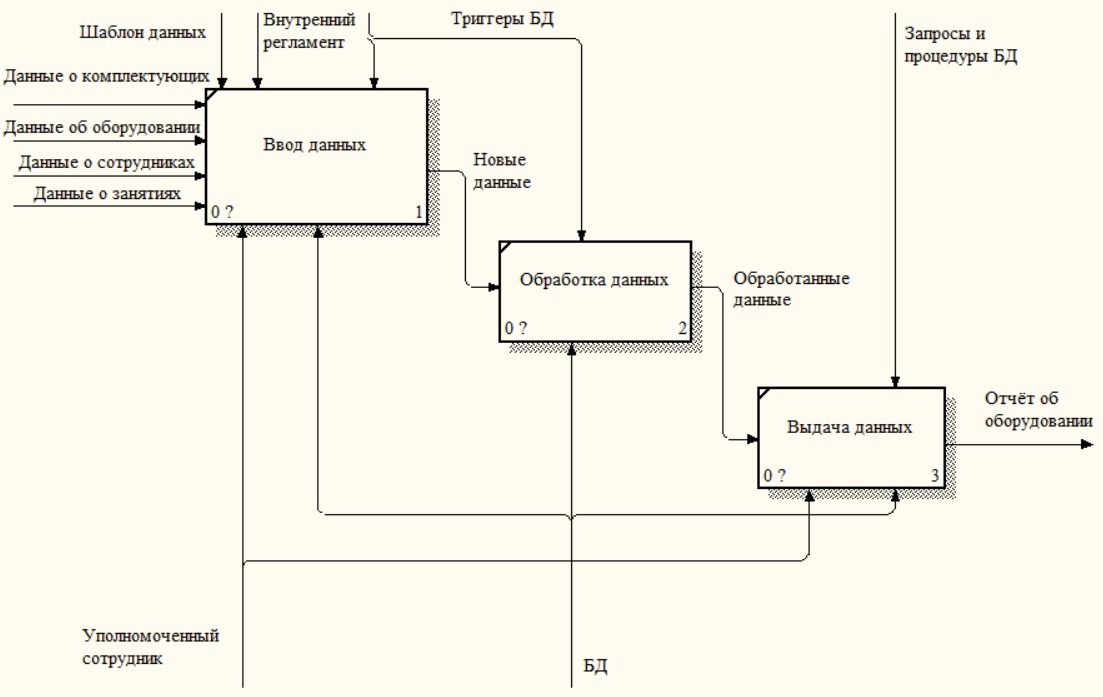


Рисунок 1.1.3. Декомпозиция модели “Чёрный ящик”

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 1.2.1.

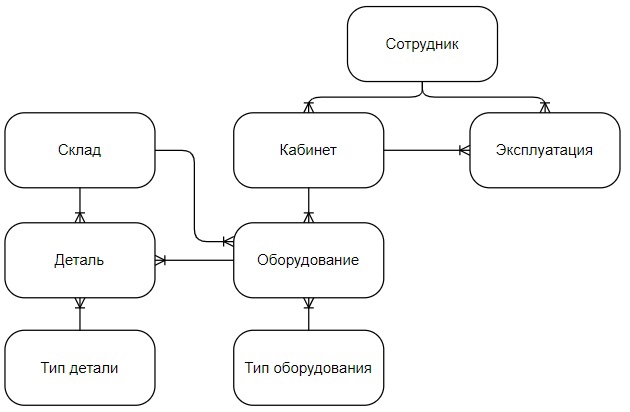


Рисунок 1.2.1. Концептуальная модель данных

Концептуальная модель данных отражает отношения между сущностями, указанными на диаграмме. В базе данных ведется учет сотрудников. Один сотрудник может быть материально ответственным за все кабинеты вычислительного центра. Другие сотрудники в лице преподавателей получают оборудование в распоряжение на срок, указанный в расписании учебных занятий заведения. Подразумевается, что эти сроки измеряются в занятиях и записываются в таблицу «Эксплуатация». Сотрудники, указанные в этой таблице, являются ответственными за эксплуатацию на указанный в этой же таблице срок. В одном кабинете может находиться большое количество оборудования, соответствующего определенному типу из заданных в таблице «Тип оборудования» (монитор, системный блок, смарт-доска и многие другие). В случае, если оборудование предусматривает наличие комплектующих (у принтера есть чернильный картридж, а у системного блока есть много комплектующих), их учет производится в таблице «Деталь». Каждая деталь имеет тип, притом все типы перечислены в таблице «Тип детали» и могут пополняться новыми наименованиями при необходимости. В случае, если оборудование или деталь находится на складе, соответствующее поле «warehouse» («склад») в этих записях должно быть заполнено и соответствовать номерам складов, указанных в таблице «Склад». Далее представлена логическая модель данных.

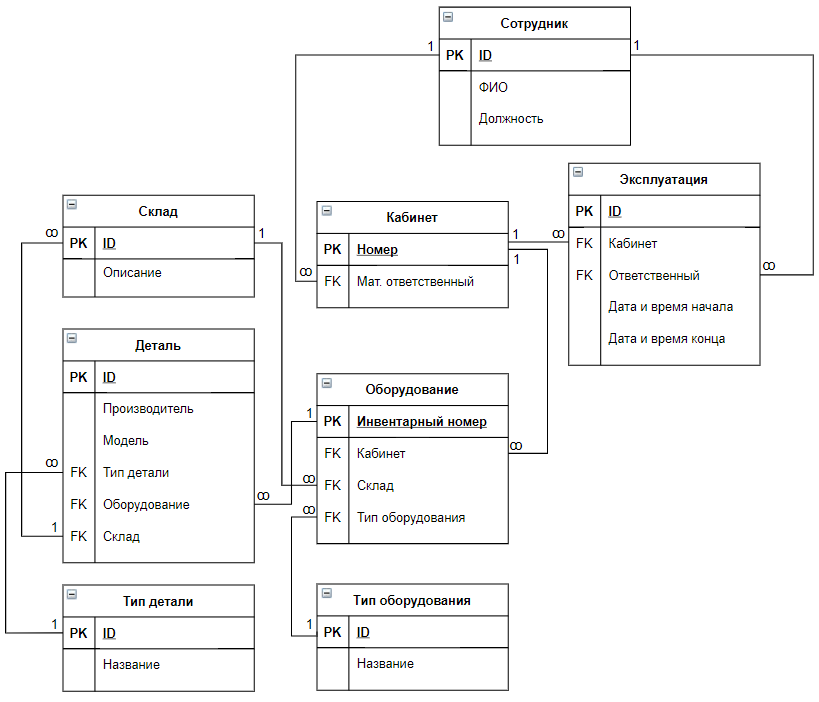


Рисунок 1.2.2. Логическая модель данных

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 1.2.3. В ней происходит описание сущностей, их названия и атрибуты, а также связи между ними.

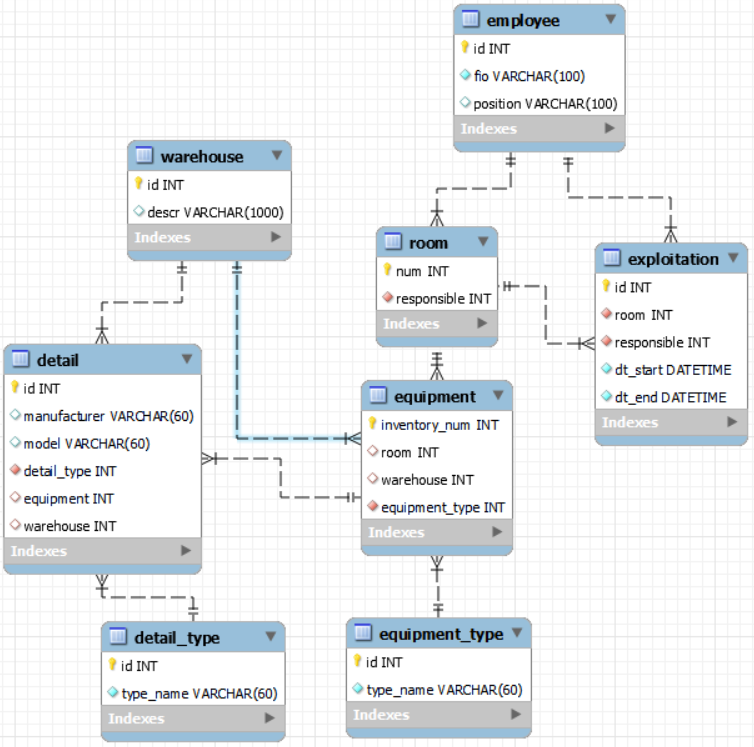


Рисунок 1.2.3. Физическая модель данных

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 программу 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.

# **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, 3),

(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 кабинетов и материально ответственных за эти кабинеты

SELECT Num, fio AS ‘Responsible’ FROM room r, employee e

WHERE r.responsible = e.id;

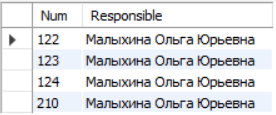


Рисунок 1.4.1. Результат выполнения запроса

1. Подсчет количества оборудования каждого типа

SELECT et.type\_name AS 'Equipment type', (SELECT count(\*) AS 'Count' FROM equipment e

WHERE et.id = e.equipment\_type) AS 'Count'

FROM equipment\_type et;

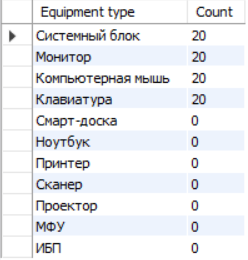


Рисунок 1.4.2. Результат выполнения запроса

1. Вывод журнала эксплуатации, отсортированного по времени начала эксплуатации.

SELECT Room, fio AS 'Name', dt\_start AS 'Start', dt\_end AS 'End'

FROM exploitation e

INNER JOIN employee em ON e.responsible = em.id

ORDER BY dt\_start;

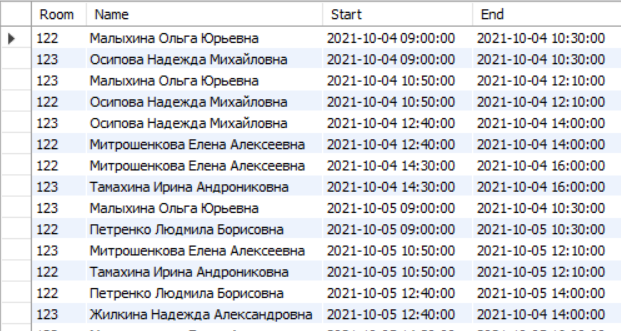


Рисунок 1.4.3. Часть результата запроса

1. Выборка кабинетов, в которых нет указанного типа оборудования

SET @t\_name = 'Монитор';

SELECT num from room WHERE num NOT IN (

SELECT DISTINCT room FROM equipment e

inner join equipment\_type et on et.id = e.equipment\_type

where e.equipment\_type = (select et.id from equipment\_type et where et.type\_name = @t\_name));



Рисунок 1.4.4. Результат выполнения выборки

1. Выборка всех сотрудников, проводивших занятия или взявших ответственность за эксплуатацию в указанном кабинете и в указанный день.

SET @classroom = 122, @dt = '2021-10-04';

SELECT DISTINCT em.id, em.fio FROM employee em

INNER JOIN exploitation ex ON ex.responsible = em.id

WHERE ex.room = @classroom AND cast(ex.dt\_start AS DATE) = @dt;

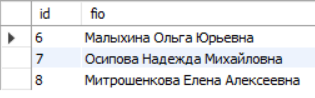


Рисунок 1.4.5. Результат выполнения выборки

Список триггеров:

1. Добавление кабинетов (номер должен быть трёхзначным)

delimiter //

CREATE TRIGGER Warehouse\_description\_warning

BEFORE INSERT ON warehouse FOR EACH ROW

BEGIN

if new.descr is null then

SET new.descr = 'ПРЕДУПРЕЖДЕНИЕ: необходимо добавить описание склада во избежание ошибок распределения оборудования.';

end if;

END //

delimiter ;

Пример работы триггера при выполнении команд ниже представлен на рисунке 1.4.6.

insert into warehouse () values ();

select \* from warehouse;

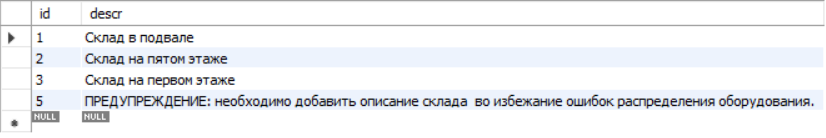


Рисунок 1.4.6. Результат выполнения запроса для триггера на добавление

1. Изменение времени эксплуатации

delimiter //

CREATE TRIGGER Exploitation\_dt\_error

BEFORE UPDATE ON exploitation FOR EACH ROW

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

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Время начала эксплуатации не может быть позже времени конца эксплуатации.';

end if; //

delimiter ;

Пример работы триггера при выполнении команд ниже представлен на рисунке 1.4.7.

update exploitation set dt\_start = '2022-10-04 09:00:00' where id = 1;

select \* from exploitation limit 1;

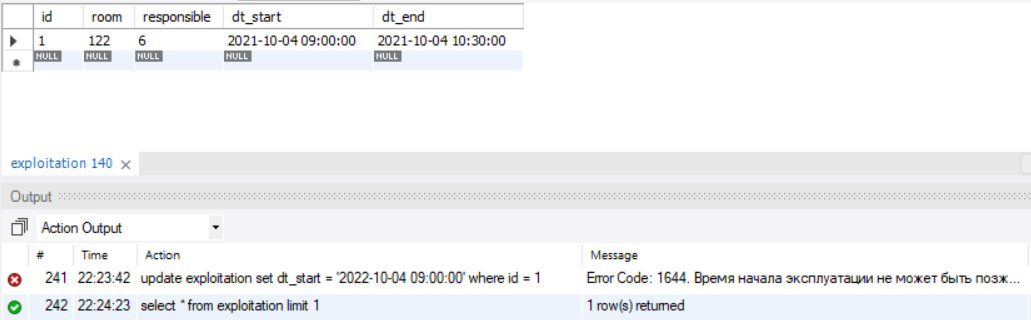


Рисунок 1.4.7. Результат выполнения запроса на изменение

1. Удаление деталей

delimiter //

CREATE TRIGGER Detail\_deletion\_error

BEFORE DELETE ON Detail FOR EACH ROW

if old.equipment is not null then

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Сначала деталь необходимо изъять из оборудования или воспользоваться процедурой

Detail\_dispose(detail\_id).';

end if; //

delimiter ;

Пример работы триггера при выполнении команд ниже представлен на рисунке 1.4.8.

insert into detail (detail\_type, equipment) values (2, 1);

delete from detail where id = 2;



Рисунок 1.4.8. Результат выполнения запроса

Хранимые процедуры:

1. Подсчет количества оборудования указанного типа в данном кабинете

delimiter //

CREATE PROCEDURE Count\_equipment\_in\_room (in roomNum int, in eq\_type int, out amount int)

BEGIN

select count(e.inventory\_num) into amount from equipment e

inner join equipment\_type et on et.id = e.equipment\_type

where et.id = eq\_type and e.room = roomNum;

END; //

delimiter ;

Результат процедуры выводится в переменную и может быть использована позже. Вывод после выполнения команд представлен на рисунке 1.4.9.

call Count\_equipment\_in\_room(123, 1, @amount);

select @amount;



Рисунок 1.4.9. Результат выполнения процедуры для подсчета оборудования

1. Процедура для списания детали

delimiter //

CREATE PROCEDURE Detail\_dispose (in detail\_id int)

BEGIN

declare e int;

select equipment into e from detail where id = detail\_id;

if e is null then

delete from detail where id = detail\_id;

else

update detail set equipment = null;

delete from detail where id = detail\_id;

end if;

END; //

delimiter ;

Перед выполнением указанных ниже команд в таблице деталей находилось две записи. Результат выполнение запросов указан на рисунке 1.4.10, по которому видно, что одна из деталей была успешно удалена.

call detail\_dispose(1);

select \* from detail;

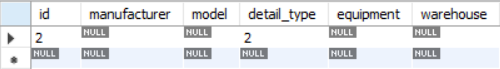


Рисунок 1.4.10. Результат выполнения процедуры количества оборудования по его типу

1. Замена картриджа в принтере и удаление использованного из базы данных

В базе данных одним из возможных типов деталей является чернильный картридж для принтера. Для удобства регистрации их замены в принтерах, была создана специальная процедура.

В рамках тестирования в таблицу equipment была добавлена запись о принтере, а в таблицу detail – картриджа, один из которых вставлен в принтер. Команды, используемые для этого, представлены ниже:

insert into equipment (inventory\_num, room, equipment\_type) values (101, 123, 7);

insert into detail (detail\_type, equipment) values (1, 101);

insert into detail (detail\_type, warehouse) values (1, 1);

Листинг создания процедуры смены картриджа:

delimiter //

CREATE PROCEDURE Change\_cartridge (in printerID int, in newCartridgeID int)

BEGIN

if not exists (select \* from detail where warehouse is not null and id = newCartridgeID) then

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Указанного чернильного картриджа нет на складе.';

end if;

call detail\_dispose((select id from detail where equipment = printerID));

update detail set equipment = printerID, warehouse = null;

END; //

delimiter ;

Ниже изображена таблица detail до применения процедуры:

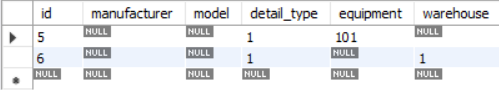


Рисунок 1.4.11. Результат выполнения процедуры выборки деталей по их типу

Результат выполнения процедуры командой ниже изображен на рисунке 1.4.12.

call change\_cartridge(101, 6);

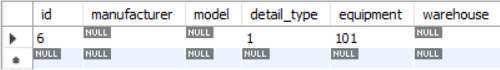


Рисунок 1.4.12. Таблица detail после выполнения процедуры

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';

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

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

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

flush privileges;

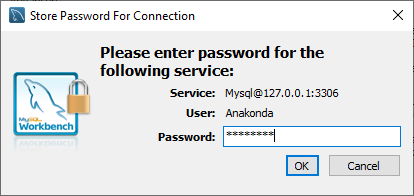


Рисунок 1.4.13. Окно аутентификации для пользователя Anakonda

use equipment;

select \* from warehouse;

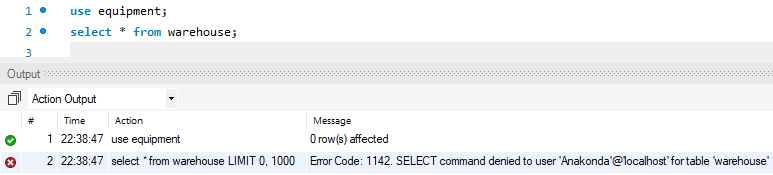


Рисунок 1.4.14. Результат выполнения запроса за пользователя Anakonda

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';

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';

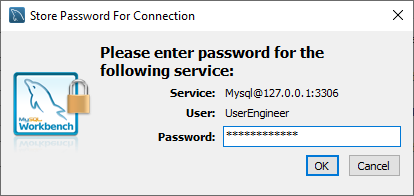


Рисунок 1.4.15. Окно аутентификации для пользователя UserEngineer

use equipment;

insert into room values (119, 6);

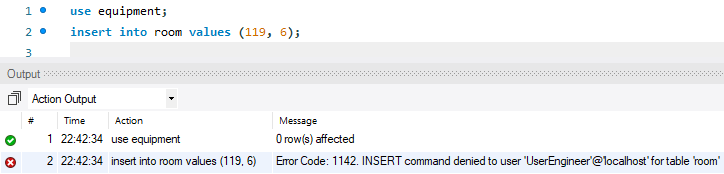


Рисунок 1.4.16. Результат выполнения запроса за пользователя UserEngineer

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';

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';

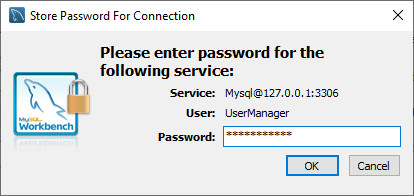


Рисунок 1.4.17. Окно аутентификации пользователя UserManager

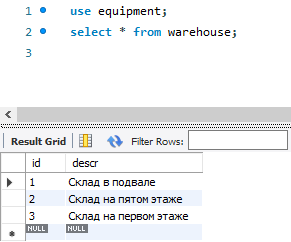


Рисунок 1.4.18. Результат выполнения запроса выборки за пользователя UserManager

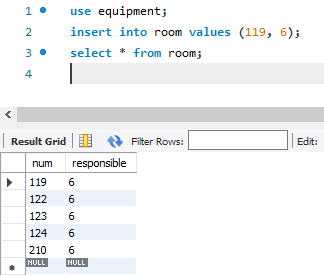


Рисунок 1.4.19. Результат выполнения запроса с добавлением за пользователя UserManager

# **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\_\_\_\_\_\_\_\_\_\_\_\_\_Симонян Павел Русланович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(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)

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

Должность Подпись Ф.И.О. руководителя практики

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

Руководитель практики от образовательного учреждения

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

Должность Подпись Ф.И.О. руководителя практики