Министерство образования Иркутской области

Государственное бюджетное профессиональное

образовательное учреждение Иркутской области

«Иркутский авиационный техникум»

(ГБПОУИО «ИАТ»)

ПП.09.02.07-1.24.202.18

ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ

ПМ.02 Осуществление интеграции программных модулей

|  |  |  |
| --- | --- | --- |
| Руководитель от предприятия:  М.П. | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (А.А.Тазеева) |
| Руководитель от техникума: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (С.Н.Касьяненко) |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Ю.А.Сологаев) |

Иркутск 2024

Содержание

[Введение 3](#_Toc163248889)

[1 Создать проект на GitHub. 4](#_Toc163248890)

[2 Техническое задание 5](#_Toc163248892)

[Приложение А – Техническое задание 6](#_Toc163248893)

[2.1 Загрузка файла технического задания 9](#_Toc163248895)

[3 Модуль приема данных извне MySQL 10](#_Toc163248896)

[4 Выполнение отладки 11](#_Toc163248897)

[5 Тестовые наборы данных 12](#_Toc163248898)

[6 Настройка Roslyn 13](#_Toc163248899)

[Заключение 14](#_Toc163248900)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 15](#_Toc163248901)

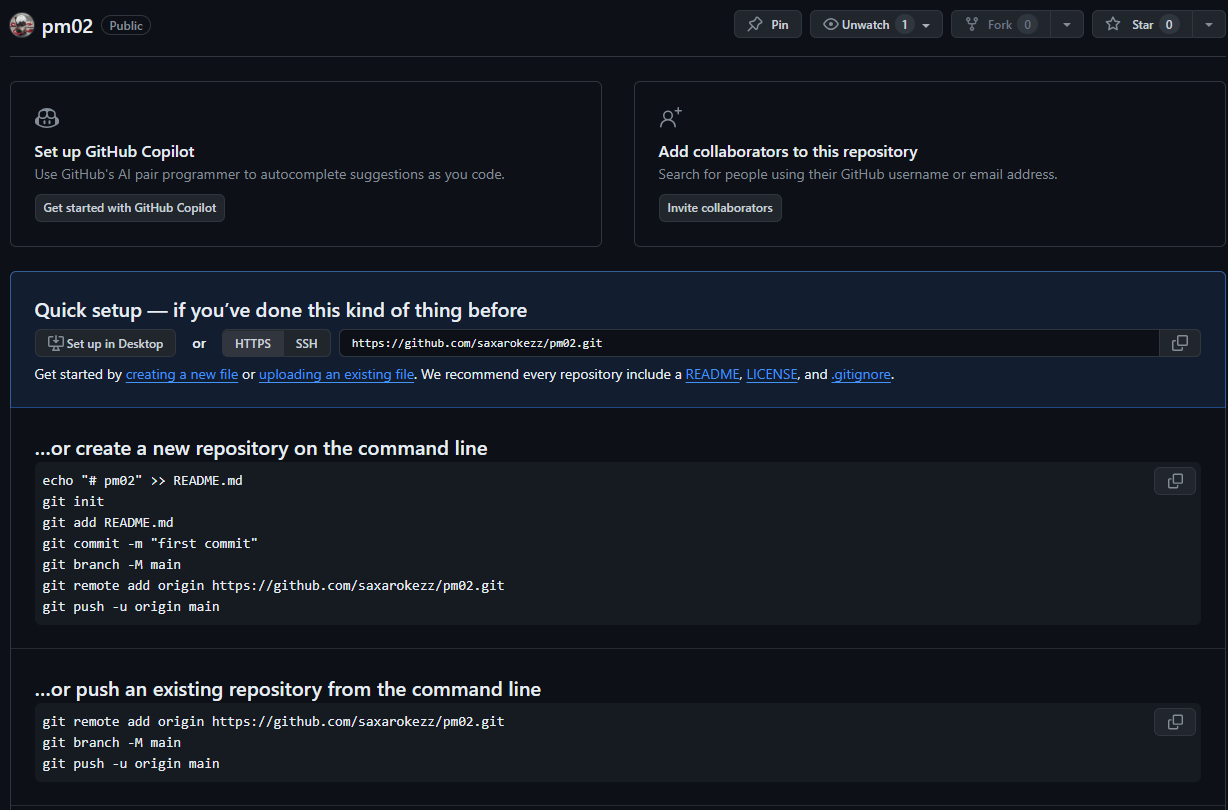
# **Введение**

Производственная практика необходима для того, чтобы студенты могли применить свои знания, полученные в процессе обучения, на практике. Она также помогает студентам понять, как работает компания, и дает им возможность узнать больше о своей будущей профессии. Кроме того, производственная практика может помочь студентам определиться с будущей карьерой и выбрать наиболее подходящую для них сферу деятельности.

Целью производственной практики являются следующие задачи:

1. Приобрести необходимые навыки самостоятельной работы по анализу и умению работать с данными;
2. Обеспечить сбор и обработку необходимого статистического материала для выполнения практических заданий.
3. **Создать проект на GitHub.**

После входа в систему переходим в меню New Repositories. Выбрать тип репозитория и ввести имя репозитория.



1. Техническое задание

1. Наименование проекта и цель создания базы данных.

2. Общие требования к базе данных.

3. Описание структуры данных.

4. Требования к целостности данных.

5. Требования к документации.

Приложение А – Техническое задание

Министерство образования Иркутской области

Государственное бюджетное профессиональное образовательное учреждение

Иркутской области

«Иркутский авиационный техникум»

(ГБПОУИО «ИАТ»)

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

РАЗРАБОТКА БАЗЫ ДАННЫХ

|  |  |  |
| --- | --- | --- |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (С.Н. Касьяненко) |
|  |  |  |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Ю.А. Сологаев) |

Иркутск 2024

**1 Наименование проекта и цель создания базы данных**

Наименование работы: Информационная система «Компьютерные сети».

Исполнитель: студент Иркутского авиационного техникума, группы БД-20-2, Сологаев Ю.А.

Разработка приложения проходит в рамках производственной практики для ГБПОУИО «ИАТ».

Сроки разработки с 14.03.2024 по 04.04.2024 года.

**2 Общие требования к базе данных**

* База данных должна хранить информацию о клиентах, заказах, продуктах и администраторах.
* База данных должна содержать информацию данных клиента (логин, пароль).
* База данных должна иметь возможность поиска по названию.
* База данных должна поддерживать целостность данных и обеспечивать безопасность доступа.
* База данных должна обеспечивать возможность добавления новых продуктов, а также изменения информации о существующих.

1. **Описание структуры данных**

База данных «Компьютерная техника» может состоять из следующих таблиц:

* + Пользователь. (Код, Телефон, Пароль, Логин, Возраст, Роль).
  + Номер. (Код номера, название, цена, статус, категория).
  + Роль. (Код роли, название)

То есть таблицы: «users», «numbers», «roles».

1. **Требования к целостности данных**

Требования к целостности данных таковы:

* + 1. Уникальность данных: каждая запись в базе данных должна быть уникальной, чтобы избежать дублирования информации.
    2. Согласованность данных: данные в базе данных должны быть согласованными и точными, чтобы избежать противоречий и ошибок.
    3. Ограничения целостности: использование ограничений целостности, таких как уникальные ключи, внешние ключи и проверочные ограничения, для обеспечения правильности данных.
    4. Актуальность данных: данные должны быть актуальными и обновленными, чтобы предоставлять точную информацию пользователям.
    5. Безопасность данных: обеспечение защиты данных от несанкционированного доступа, изменений и удалений.
    6. Резервное копирование данных: регулярное создание резервных копий данных для предотвращения потери информации в случае сбоев или аварий.

**5 Требования к документации**

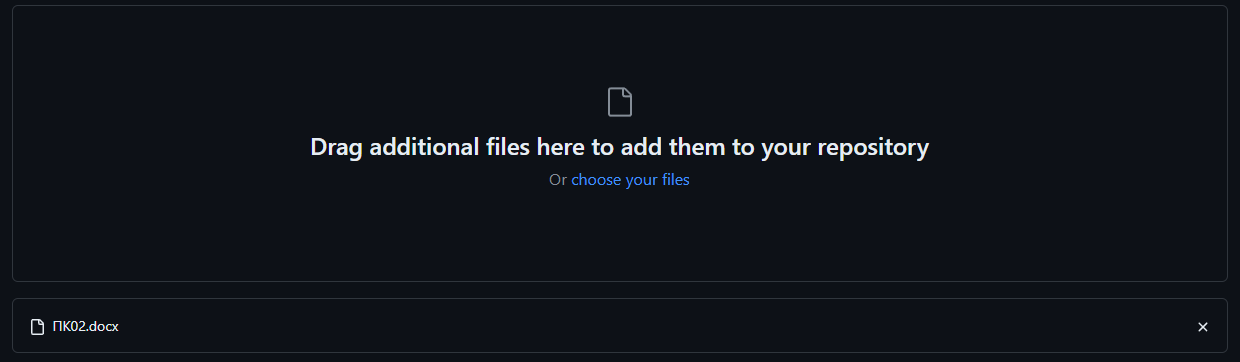
Основным документам, регламентирующими использование базы данных является руководство пользователя.

Основным документам, регламентирующими разработку базы данных является техническое задание.

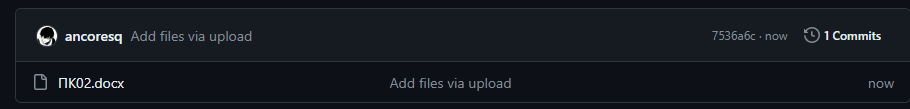
# **2.1 Загрузка файла технического задания**

Для загрузки файла на GitHub необходимо зайти в созданный проект. Далее нажать на сслыку «[uploading](https://github.com/nastyahor/qwe/upload) an existing file.»





Нажимаем на кнопку «Commit changes».

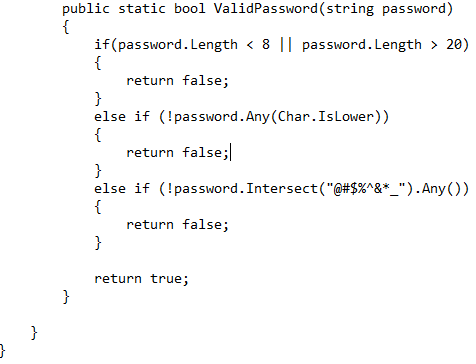


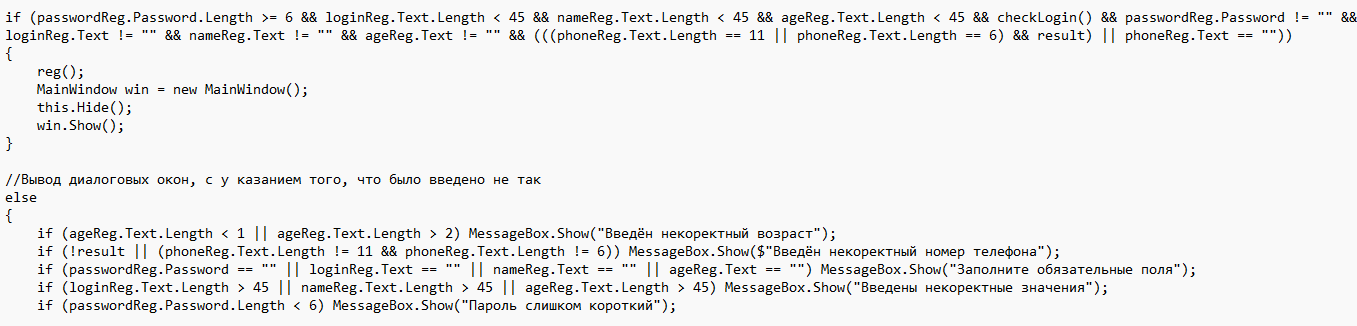
# **3 Модуль приема данных извне MySQL**

Для создания модуля,который проверяет их на корректность, а затем записывает их в базу данных MySQL, вам понадобится использовать для обработки данных и выполнения запросов к базе данных.

Код, который может быть использован для создания такого модуля:





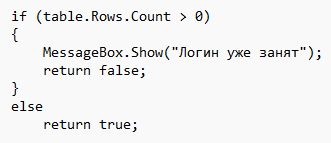


Этот код демонстрирует базовый пример модуля для приема данных извне, их проверки и записи в базу данных.

# **4 Выполнение отладки**

Отладка программного кода - это процесс поиска и устранения ошибок (багов), которые могут привести к неправильной работе программы. Во время разработки программы или при ее исполнении могут возникать различные проблемы, такие как неправильное поведение, неверные результаты или даже сбои в работе программы.

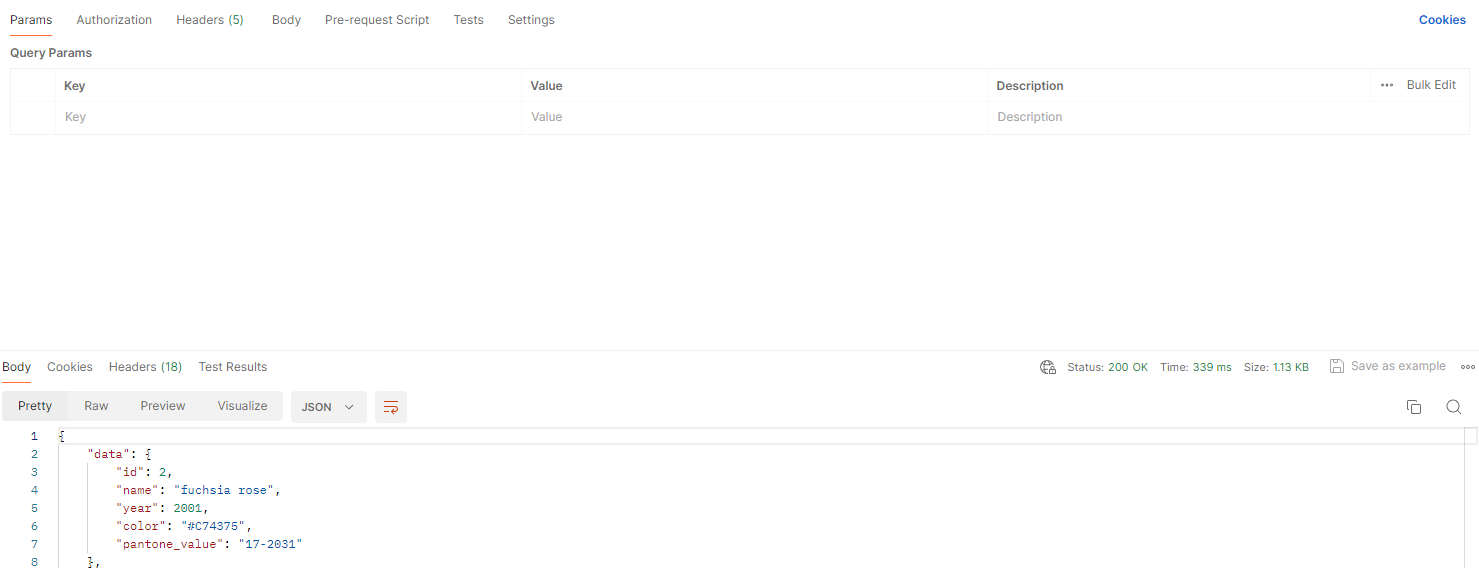
C помощью данной откладки пользователь видит сообщение о том , что какие-то данные были введены некорректно. Например:



Отладка является важной частью процесса разработки программного обеспечения и помогает создавать качественные и стабильные программы. Она требует тщательности, внимательности и систематичного подхода для успешного нахождения и устранения ошибок в коде.

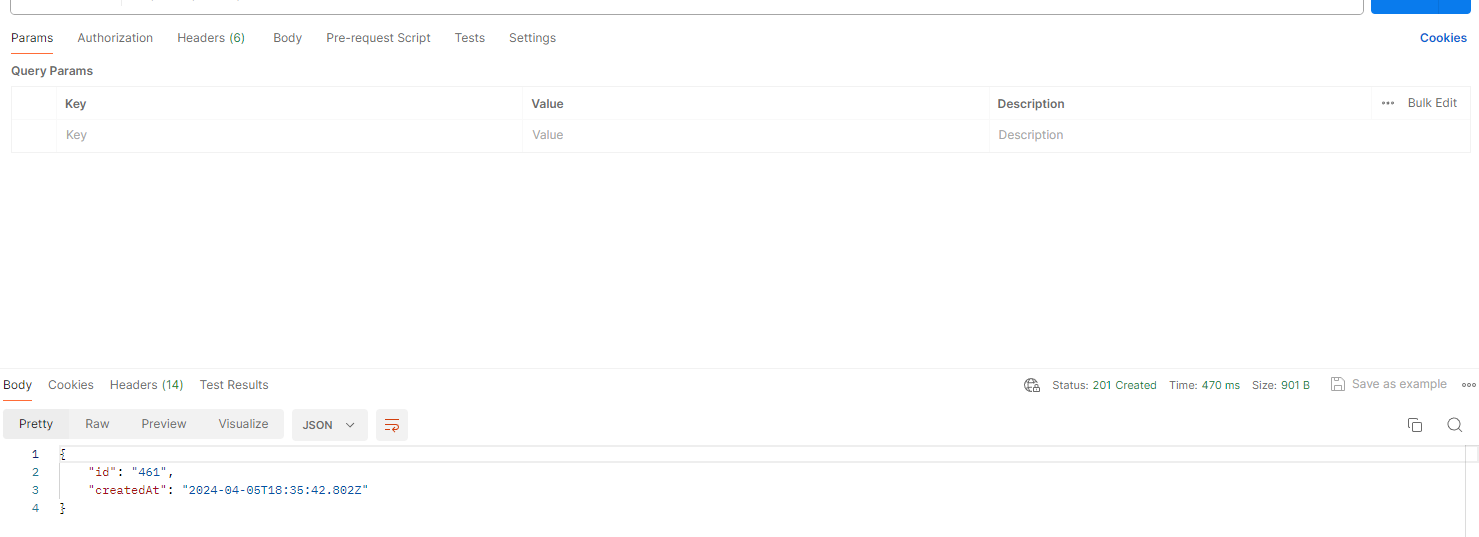
# **5 Тестовые наборы данных**

С помощью Postman вы можете создавать и отправлять различные типы запросов (GET, POST, PUT, DELETE и т.д.) к API, указывать параметры запроса, заголовки, тело запроса и многое другое. Postman также позволяет сохранять запросы, создавать коллекции запросов, автоматизировать тестирование API и генерировать документацию.  
 GET - это один из методов запроса, который используется для получения данных с сервера. Он отличается от других методов, таких как POST, PUT и DELETE, тем, что не изменяет данные на сервере, а только получает их. GET-запросы обычно используются для получения информации, просмотра страниц или загрузки файлов.



Для передачи конфиденциальных данных рекомендуется использовать метод POST.

Так же изображен результат запроса POST в формате JSON.



# **Настройка Roslyn**

Roslyn - это компилятор языка C# в платформу .NET, который позволяет анализировать, транспилировать и проверять исходный код на наличие ошибок и несоответствий. Он также позволяет создавать программы на C# без необходимости компилировать код вручную. Установка:

Скачать и установить компилятор Roslyn с официального сайта: https://roslyn.io/. В Visual Studio перейти в меню “Инструменты” > “Расширения и обновления” и найти “Roslyn”. Установить необходимые расширения. После установки перезапустить Visual Studio.

Для настройки Roslyn вам необходимо создать новый проект в Visual Studio и выбрать соответствующую версию .NET Core. Затем вы можете добавить необходимые пакеты NuGet и настроить компилятор в файле проекта.

# **Заключение**

В ходе производственной практики были получены практические навыки в области разработки программного обеспечения. Были изучены основные этапы разработки программных продуктов, а также методы и инструменты для их создания.  
 Были проведены работы по тестированию программного обеспечения, анализу требований и проектированию баз данных. Также были изучены методы оптимизации производительности программных систем и обеспечения их безопасности.

Производственная практика позволила получить опыт работы в команде, научиться эффективно взаимодействовать с другими специалистами и решать проблемы.  
 В процессе выполнения практических заданий были закреплены навыки:

* Создания проекта на GitHub.
* Составления технического задания на разработку Базы данных в соответствии с ГОСТ.
* Написание модуля приема данных извне
* Выполнение отладки любым из доступных инструментов.
* Подготовка тестовых наборов данных для запросов GET и POST с помощью Postman.
* Настройка Roslyn.

Таким образом, производственная практика является важным этапом в процессе обучения и позволяет студентам получить необходимые навыки и знания для успешной работы в сфере информационных технологий.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. https://web.postman.co/workspace
2. https://github.com/ancoresq/nn
3. https://learn.microsoft.com/ru-ru/visualstudio/code-quality/use-roslyn-analyzers?view=vs-2022
4. https://learn.microsoft.com/ru-ru/visualstudio/get-started/csharp/tutorial-debugger?view=vs-2022
5. https://help.sprinthost.ru/services/remotedb