Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизированных систем управления (АСУ)

**СИСТЕМА ИНТЕЛЛЕКТУАЛЬНОГО ПОИСКА. СПОСОБЫ УЛУЧШЕНИЯ МЕТОДОВ ПОИСКА LLM. МОДУЛЬ ОБРАБОТКИ И ФОРМАТИРОВАНИЯ ТАБЛИЦ**

Отчет

по результатам

производственной практики: преддипломная практика

|  |  |
| --- | --- |
|  | Обучающийся гр. 431-3  Е.П.Бекиш  (подпись) (И.О.Фамилия)    (дата) |
| (оценка)  М.П. | Руководитель практики от профильной организации:  Начальник отдела автоматизации ООО «РН-КрасноярскНИПИнефть»  (должность, ученая степень, звание)  А.С.Яковлюк  (подпись) (И.О.Фамилия)    (дата) |
| (оценка) | Руководитель практики от Университета:  Доцент кафедры АСУ, к.т.н.  А.А.Шелестов  (подпись) (И.О.Фамилия)    (дата) |

Томск 2025

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизированных систем управления (АСУ)

**УТВЕРЖДАЮ**

Зав. кафедрой АСУ

канд. техн. наук, доцент

В. В. Романенко

(подпись)

« » 20*25* г.

(дата)

ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

на производственную практику: преддипломная практика

студенту гр.431-3 факультета систем управления

Бекиш Егору Павловичу

1. Тема практики: Система интеллектуального поиска. Способы улучшения методов поиска LLM. Модуль обработки и форматирования таблиц.
2. Цель практики: подготовка к выполнению выпускной квалификационной работы по разработке системы интеллектуального поиска в корпоративных базах знаний в процессе технологической подготовки производства ООО «РН-КрасноярскНИПИнефть».
3. Сроки прохождения практики: 10.02.2025 – 17.05.2025

**Совместный рабочий график (план) проведения практики**

|  |  |  |
| --- | --- | --- |
| № п/п | Перечень заданий | Сроки выполнения |
| 1 | Прохождение техминимума и тех. безопасности на рабочем месте. | 10.02.2025 |
| 2 | Ознакомление со структурой предприятия, видами деятельности, процессами организации управления деятельностью предприятия | 14.02.2025 |
| 3 | Изучение целей и функций автоматизации технологических процессов, автоматизированных систем управления, используемых средств вычислительной техники в деятельности предприятия | 14.02.2025 |
| 4 | Изучение и освоение различных пакетов программ, применяемых на предприятии | 21.02.2025 |
| 5 | Постановка задачи автоматизации (описание предметной области, разработка требований к программному продукту) | 28.02.2025 |
| 6 | Обзор аналогов для решения задачи | 10.03.2025 |
| 7 | Проектирование модуля обработки и форматирования таблиц | 31.03.2025 |
| 8 | Реализация модуля обработки и форматирования таблиц | 21.04.2025 |
| 9 | Тестирование модуля обработки и форматирования таблиц | 28.04.2025 |
| 10 | Написание отчета по практике и создание презентации с основными результатами | 12.05.2025 |

Дата выдачи: 10.02.2025 г.

Руководитель практики от университета

|  |  |  |
| --- | --- | --- |
| Доцент кафедры АСУ, к.т.н. | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Подпись) | Шелестов А.А. |

Согласовано:

Руководитель практики от профильной организации

|  |  |  |
| --- | --- | --- |
| Начальник отдела автоматизации  (должность) | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Подпись)  М.П. | Яковлюк А.С.  (Ф.И.О.) |

Задание принял к исполнению 10.02.2025 г.

|  |  |  |
| --- | --- | --- |
| Студент гр.431-3 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Подпись) | Бекиш Е.П.  (Ф.И.О.) |

**Оглавление**

[Введение 5](#_Toc198050165)

[1 ПРЕДПРИЯТИЕ ООО «РН-КРАСНОЯРСКНИПИНЕФТЬ» 7](#_Toc198050166)

[1.1 Общая характеристика предприятия 7](#_Toc198050167)

[1.2 Научная деятельность специалиста BI 8](#_Toc198050168)

[2 ПОСТАНОВКА ЗАДАЧИ 10](#_Toc198050169)

[3 ОБЗОР АНАЛОГОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 11](#_Toc198050170)

[4 ПРОЕКТИРОВАНИЕ СИСТЕМЫ. 12](#_Toc198050171)

[4.1 Проектирование функций взаимодействия 12](#_Toc198050172)

[5 РАЗРАБОТКА СИСТЕМЫ 16](#_Toc198050173)

[5.1 Предварительная настройка проекта 16](#_Toc198050174)

[5.2 Реализация функций 16](#_Toc198050175)

[6 РЕЗУЛЬТАТЫ РАБОТЫ 17](#_Toc198050176)

[6.1 Этапы получения корректной таблицы 17](#_Toc198050177)

[Заключение 21](#_Toc198050178)

[Список использованных источников 22](#_Toc198050179)

# **Введение**

Производственная практика: «Преддипломная практика» проходила в компании ООО «РН-КрасноярскНИПИнефть».

Для исследований в ходе практики была выбрана тема «Система интеллектуального поиска. Способы улучшения поиска LLM. Модуль обработки и форматирования таблиц».

Языковые модели являются важной и активно развивающейся областью, которая за последние несколько лет привлекает большое внимание промышленности, научных и академических кругов [1].

В современном мире объемы информации с каждым днем растут все больше и больше, поэтому анализировать информацию становится труднее, как и выбрать более корректную. В организациях существуют собственные база знаний, которые имеют тоже не маленький объем. Поэтому, чтобы повысить собственное качество и работоспособность им бы хотелось избегать анализа лишнего материала, т.к. на нее тратится очень большего количество времени.

Кроме того, чтобы в дальнейшем обрабатывать предоставленные документы, необходимо их обработать для более точного и релевантного поиска, чтобы языковая модель могла это определить.

Целью практики является подготовка к выполнению выпускной квалификационной работы по созданию системы интеллектуального поиска в корпоративных базах знаний.

Основные задачи:

* ознакомление со структурой компании, видами деятельности, процессами организации управления деятельностью компании;
* изучение целей и функций автоматизации технологических процессов, автоматизированных систем управления, используемых средств вычислительной техники в действиях данной компании;
* изучение и освоение различных пакетов программ, применяемых в компании;
* постановка задачи автоматизации (описание предметной области, разработка требований к программному продукту);
* обзор аналогов для решения задачи;
* проектирование системы;
* реализация системы.

Объектом исследования является структура считывания и отображение объектов в документах.

Предмет исследования – методы и технологии считывания, обработки и форматирования объектов в документах.

В ходе исследования осуществлялся тематический поиск и критический анализ научно-технической информации, разработано программное обеспечение

# **ПРЕДПРИЯТИЕ ООО «РН-КРАСНОЯРСКНИПИНЕФТЬ»**

## **Общая характеристика предприятия**

ООО «РН-КрасноярскНИПИнефть» – это инжиниринговая компания, расположенная в Красноярском крае, г. Красноярск. Предприятие специализируется на научно-исследовательских и проектных работах в нефтегазовой отрасли, обеспечивая полный цикл услуг – от геологоразведки до проектирования и сопровождения разработки месторождений.

Основная деятельность компании связана с проведением комплексных исследований, проектированием объектов нефтедобычи и транспортировки углеводородов, а также разработкой технологических решений для повышения эффективности нефтегазовых проектов. Компания обладает значительным опытом в области моделирования месторождений, инженерных изысканий и экологического мониторинга, что позволяет ей участвовать в крупных проектах нефтегазового сектора России.

Ключевые направления работы включают:

* геологическое изучение и оценку запасов нефти и газа;
* проектирование объектов добычи, транспорта и переработки углеводородов;
* внедрение инновационных технологий для повышения нефтеотдачи пластов;
* экспертизу и сопровождение промышленной безопасности;
* разработку природоохранных мероприятий и экологический аудит.

Компания активно сотрудничает с ведущими нефтегазовыми корпорациями, такими как «Роснефть», и вносит значительный вклад в развитие энергетического комплекса Восточной Сибири. Внедрение современных технологий и научных разработок позволяет ООО «РН-КрасноярскНИПИнефть» оставаться ключевым игроком в области нефтегазового инжиниринга.

## **Научная деятельность специалиста BI**

Business Intelligence (BI) – это направление, связанное с сбором, обработкой, визуализацией и анализом данных для поддержки принятия бизнес-решений. BI-специалисты используют инструменты и технологии, которые помогают преобразовывать сырые данные в понятные отчёты, дашборды и аналитические модели, позволяя компаниям оптимизировать процессы, выявлять тенденции и повышать эффективность [2].

Наука о данных – это раздел информатики, изучающий проблемы анализа, обработки и представления данных в цифровой форме. Она объединяет методы по обработке данных в условиях больших объёмов и высокого уровня параллелизма, статистические методы, методы интеллектуального анализа данных и приложения искусственного интеллекта для работы с данными, а также методы проектирования и разработки баз данных. В данной сфере деятельности выделяют три основные роли: инженер данных, аналитик данных и специалист по науке о данных [3].

Чтобы вести бизнес, основываясь на данных, компании собирают информацию с сайтов, приложений, камер видеонаблюдения и датчиков на производстве и отдают их специалистам для анализа. Эти данные помогают прогнозировать спрос, изучать поведение клиентов, планировать эффективные рекламные кампании. Но прежде, чем работать с информацией, её нужно собрать, грамотно структурировать, где-то сохранить, а потом доставить до аналитика. Для этого нужна инфраструктура: хранилища, серверные мощности для анализа, инструменты для сбора, очистки и доставки данных.

Инженер данных – это специалист, который занимается построением и обслуживанием инфраструктуры для работы с данными, а также их предварительной обработкой. Инженер не участвует в анализе, но обеспечивает аналитиков нужными мощностями, инструментами и подготовленными данными [4].

Аналитик данных проводит статистический анализ, чтобы ответить на вопросы или решить проблемы. Для этого он получает данные, выявляет закономерности и формирует отчёты, которые помогают руководителям проекта или бизнеса принимать стратегические решения [5].

Специалист по науке о данных создаёт инструменты для решения задач бизнеса. Для этого он использует навыки анализа данных и построения моделей машинного и глубокого обучения. Специалист работает на стыке нескольких областей знания: статистики, программирования, машинного и глубокого обучения. Поэтому он не только умеет анализировать и визуализировать данные, но и строить модели на их основе, заниматься прогнозированием и оптимизацией бизнес-процессов.

Однако основной целевой аудиторией данной системы являются исследователи в области больших языковых моделей. Эти специалисты отвечают за разработку системы по нахождению ответа из релевантных документов.

В связи с этим становится ясной необходимость в инструменте, который бы позволял упростить процесс отбора, обработки и форматирование данных их документов, чтобы информация была релевантная.

# **ПОСТАНОВКА ЗАДАЧИ**

Целью разработки является создание модуля обработки и форматирования таблиц.

Цели модуля:

* разработать интуитивно понятные функции для получения данных на определенных стадиях;
* предоставить итоговый результат обработанных и форматированных таблиц.

Описание функциональности включали в себя следующие моменты:

* пользователь должен иметь доступ к документам;
* модуль должен корректно отображать обработанные, отформатированные таблицы.

# **3** **ОБЗОР АНАЛОГОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

В связи с тем, что те модули, которые существуют для получения данных из документов, не рассчитаны на полноценную обработку всех типов объектов, как таковых явных аналогов нет, но существуют модули, которые используют внутри себя языковые модели, которые не полноценно, но могут получить отформатированные таблицы.

PyMuPDF – это мощная и гибкая библиотека Python для работы с PDF-документами и другими форматами (XPS, EPUB, MOBI). Она предоставляет инструменты для извлечения текста, изображений и метаданных, а также для редактирования, аннотирования и преобразования документов. Библиотека отличается высокой производительностью и поддерживает интеграцию с современными ML-решениями, включая обработку документов для языковых моделей (LLM) [6].

MegaParse – мощная библиотека для извлечения и обработки данных из различных форматов файлов, включая PDF, DOCX, PPTX и другие. Она позволяет легко парсить структурированные и неструктурированные данные, подготавливая их для дальнейшего анализа или интеграции с ML-моделями и базами данных [7].

Markdrop – это Python-библиотека, предназначенная для удобной работы с разметкой текстовых данных в формате Markdown. Она предоставляет инструменты для парсинга, обработки и генерации Markdown–документов, что делает её полезной для разработчиков, технических писателей и аналитиков данных. Библиотека поддерживает базовые операции с Markdown и может быть полезна в проектах, где требуется программная работа с текстовыми данными в этом формате [8].

Таким образом, рассмотренные аналоги либо представляют из себя хорошие решения для анализа данных, но с недостаточным функционалом по работе с таблицами

# **4 ПРОЕКТИРОВАНИЕ СИСТЕМЫ**.

Проектирование системы играет ключевую роль в разработке программного обеспечения, обеспечивая эффективную и гибкую архитектуру системы.

Поскольку в разделе 2 были описаны общие требования системы, далее было выполнено проектирование классов с помощью UML, что в дальнейшем обеспечит понятность и поддерживаемость кода, а также облегчит добавление новых функций и модификацию существующих.

## **4.1 Проектирование функций взаимодействия**

В этом подразделе мы рассмотрим каждую функцию в отдельности, опишем его функциональность, свойства.

В результате проектирования классов, отвечающих за интерфейс системы, была получена диаграмма классов, представленная на рисунке 4.1

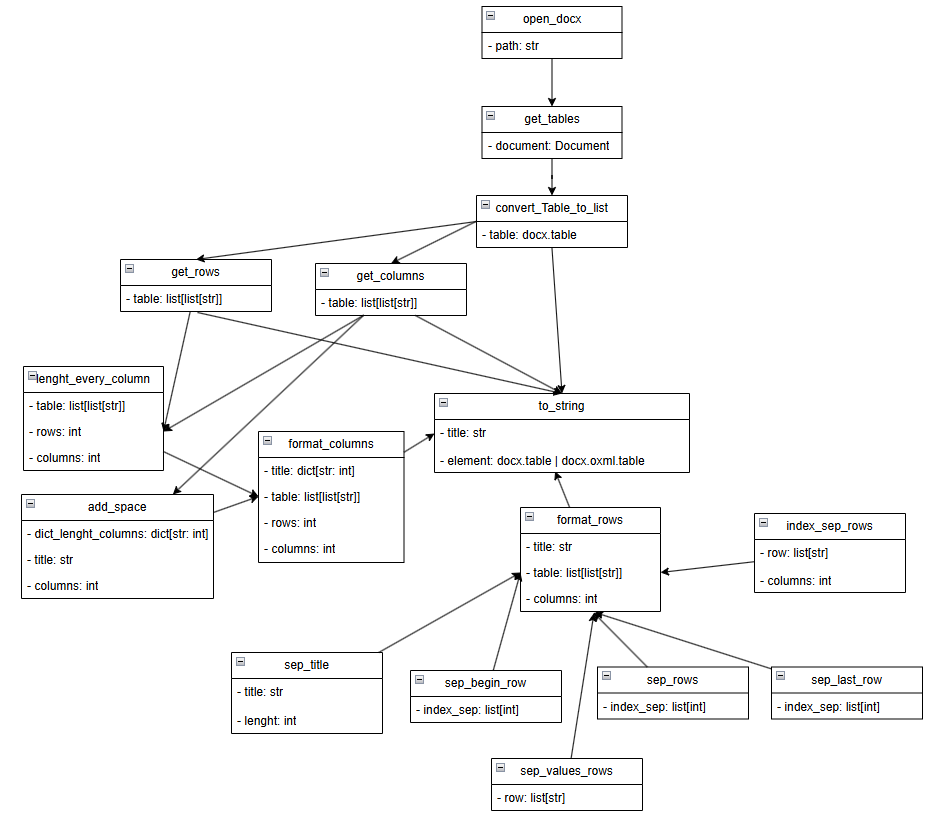


Рисунок 4.1 – Диаграмма классов функция взаимодействия

Более детальное описание функций приведено в таблице 4.1.

Таблица 4.1 – Описание функций взаимодействия

|  |  |
| --- | --- |
| Функция | Параметры |
| open\_docx(path: str) – открытие файла формата docx. | path – путь к файлу docx. |
| get\_tables(document: Document) – получение объектов таблиц. | document – объект открытого документа. |
| convert\_Table\_to\_list(table: docx.table) – преобразование объекта таблицы в список строк. | table – объект таблица. |
| get\_rows(table: list[list[str]]) – | table – таблица из списка строк. |

Продолжение таблицы 4.1

|  |  |
| --- | --- |
| получение кол-во строк в таблице. |  |
| get\_columns(table: list[list[str]]) – получение кол-во столбцов в таблице. | table – таблица из списка строк. |
| lenght\_every\_column(table: list[list[str]], rows: int, columns: int) – получение ширину столбцов. | table – таблица из списка строк;  rows – кол-во строк;  columns – кол-во столбцов. |
| add\_space(dict\_lenght\_columns: dict[str: int], title: str, columns: int) – дополнительные пробелы для ширины стобца. | dict\_lenght\_columns – ширина столбца;  title – заголовок таблицы;  columns – кол-во столбцов. |
| format\_columns(title: str, table: list[list[str], rows: int, columns: int) – форматирование столбцов. | title – заголовок таблицы;  table – таблица из списка строк;  rows – кол-во строк;  columns – кол-во столбцов. |
| sep\_values\_rows(row: list[str]) – форматирование ячеек строк таблицы. | row – строка из таблицы. |
| index\_sep\_rows(row: list[str], columns: int): – индексы для форматирование разделения ячеек | row – строка из таблицы;  columns – кол-во столбцов. |
| sep\_rows(index\_sep: list[int]) – форматирование строк таблицы. | index\_sep – индексы для форматирование разделения ячеек. |
| sep\_begin\_rows(index\_sep: list[int]) – форматирование первой строки таблицы. | index\_sep – индексы для форматирование разделения ячеек. |
| sep\_last\_rows(index\_sep: list[int]) ]) – форматирование последней строки | index\_sep – индексы для форматирование разделения ячеек. |

Окончание таблицы 4.1

|  |  |
| --- | --- |
| таблицы |  |
| sep\_title(index\_sep: list[int]) ]) – форматирование заголовка таблицы. | index\_sep – индексы для форматирование разделения ячеек. |
| format\_rows(title: str, table: list[list[str]], columns: int) – форматирование строк | title – заголовок таблицы;  table – таблица из списка строк;  columns – кол-во столбцов. |
| to\_string(title: str, element: docx.table | docx.oxml.table) – преобразование объекта таблицы в строку | title – заголовок таблицы;  element – объект таблицы. |

# **5 РАЗРАБОТКА СИСТЕМЫ**

## **5.1 Предварительная настройка проекта**

При разработке системы важно обеспечить правильную настройку рабочего окружения, которая позволит эффективно выполнять задачи разработки. В данном подразделе описывается процесс разворачивания проекта в операционной системе Window и Linux с использованием текстового редактора Visual Studio Code [9], менеджера зависимостей UV [10], фреймворка Python-docx [11].

В качестве интегрированной среды разработки (IDE) выбран Visual Studio Code, который является популярным инструментом для разработки программного обеспечения. После установки Visual Studio Code были скачаны расширения "Python" и "Mypy Type Checker" от Microsoft, чтобы обеспечить поддержку Python и удобное редактирование кода.

Для управления зависимостями Python проекта был выбран менеджер UV. Затем были добавлены необходимые зависимости посредством редактирования файла pyproject.toml с помощью команды uv add <package\_name>.

В качестве фреймворка для работы с документами был выбран Python-docx. Для его использования необходимо было установить библиотеку с помощью менеджера зависимостей UV.

## **5.2 Реализация функций**

Всего в проекте был реализовано два пакета: utils, main.

В пакете utils были созданы основные функции обработки и форматирования таблиц. Весь перечень был описан выше в таблице 4.1.

В пакете main был выполнен вызов данных функций для проверки полноценной работоспособности разработанного модуля.

# **6 РЕЗУЛЬТАТЫ РАБОТЫ**

## **6.1 Этапы получения корректной таблицы**

В самом начале необходимо открыть файл. На рисунке 6.1 показан результат открытия файла.



Рисунок 6.1 – Открытие файла docx

Как можем заметить, что у нас возвращается объект нашего документа, следовательно, он обладает следующими, для нас полезными, свойствами и атрибутами:

* получение только параграфов, т.е. абзацы, разделенные переносом на другую строку, где каждый параграф есть объект;
* получение только таблиц, где каждая таблица есть объект;
* получение всех элементов файла в формате xml.

Задача стоит в обработки таблиц, поэтому следует их получить. Результат представлен на рисунке 6.2.

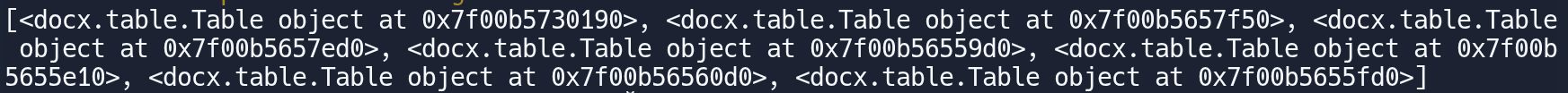


Рисунок 6.2 – Список объектов таблиц

Возьмем первую таблицу из данного списка и сравним как она выглядит в документе и после временной обработки. Результат представлен на рисунках 6.3 – 6.4.

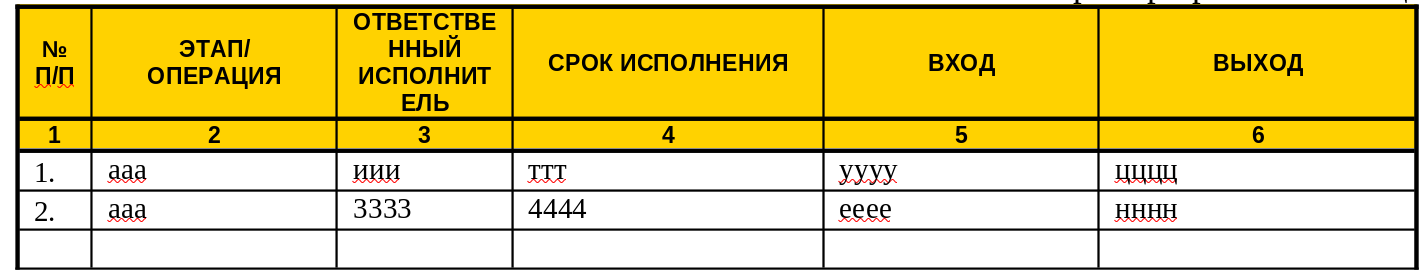


Рисунок 6.3 – Входная таблица, находящиеся в документе

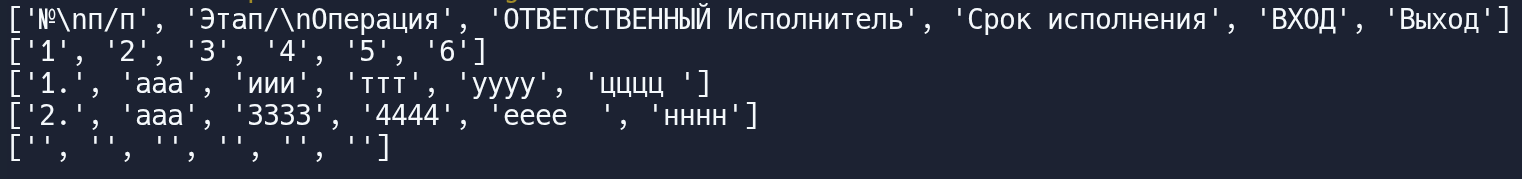


Рисунок 6.4 – Результат полученной преобразованной таблицы

Как можно было заметить, что при простой организации таблицы все обрабатывается корректно, но при разделении/объединение колонки/строки, происходит следующий результат, который показан на рисунках 6.5 – 6.8



Рисунок 6.5 – Входная таблица с объединением строк

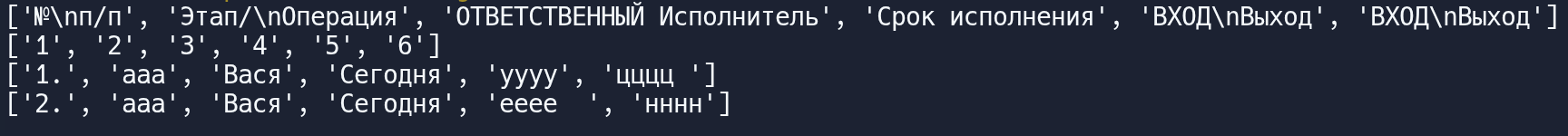


Рисунок 6.6 – Результат обработки с объединением строк



Рисунок 6.7 – Входная таблица с разделением столбцов

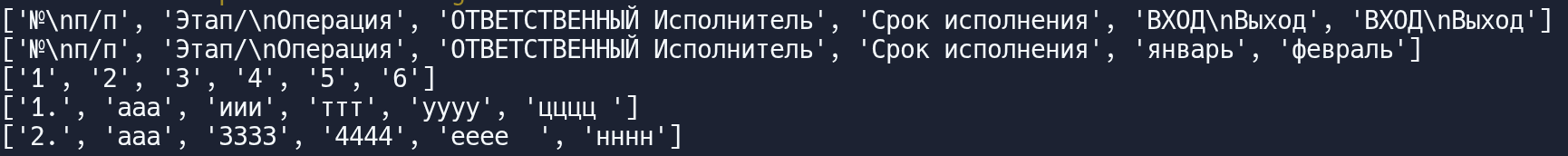


Рисунок 6.8 – Результат обработки с разделением столбцов

Из приведенных выше примеров наблюдается следующая зависимость – при объединении/разделение элементы повторяются, что и логично с точки зрения внутренней структуры xml. Это как раз хорошо повлияет на поставленную задачу.

Теперь, начнем поэтапно форматировать наши данные в таблице. Результат работы представлен на рисунке 6.9.

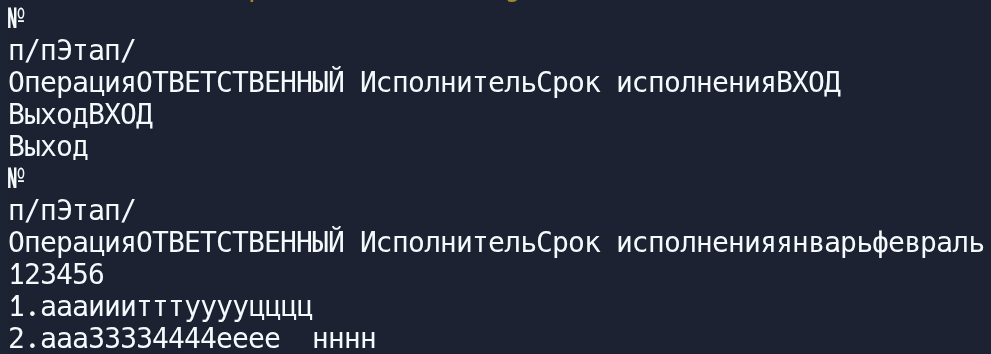


Рисунок 6.9 – Соединение элементов таблицы

На рисунке 6.9 видно, что какие-то строки и столбцы съехали, да и не понятно, как они между собой связаны, поэтому необходимо сделать определенную чистку данных, т.е. удалить лишние отступы, пробелы и т.п. и добавить некий разделитель, как это отображено на рисунке 6.10.

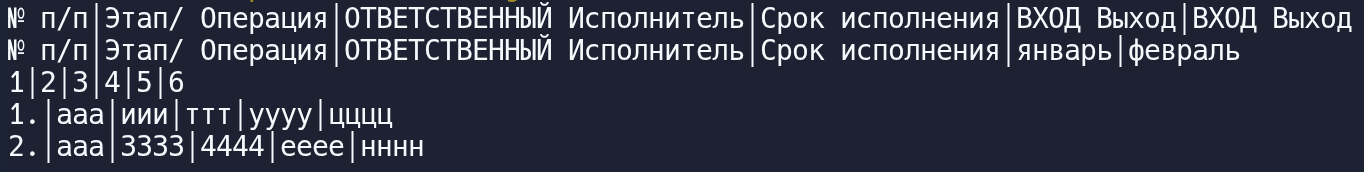


Рисунок 6.10 – Чистка данных для таблицы

Так на много лучше, но остается до сих пор не читабельно из-за разной ширины. Следовательно, для каждого столбца необходимо найти элемент, который имеет максимальную длину и выровнять по центу.

Получаем следующий вид таблицы (рисунок 6.11).

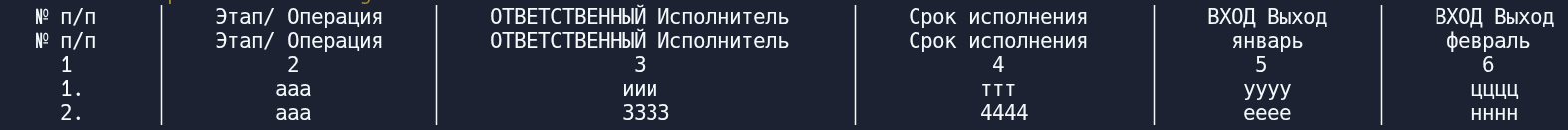


Рисунок 6.11 – Читабельный вид таблицы

Теперь, когда основная работа выполнена, немного добавим стилистики – разделим на ячейки. В результате получаем следующее (рисунок 6.12).

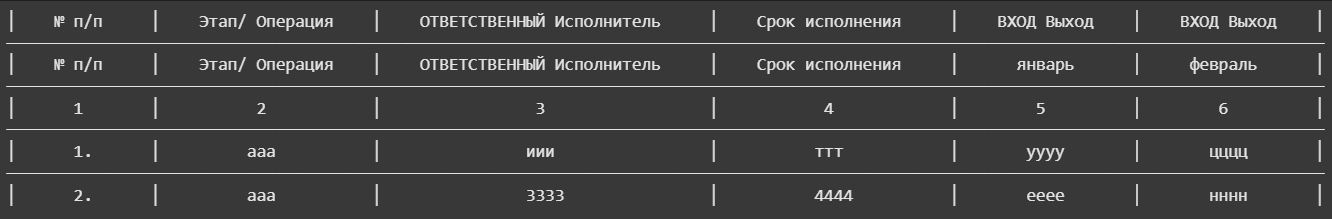


Рисунок 6.12 – Добавление разделителя

Но, как можно заметить линии между собой не соединены. Зная, про разделяющий символ между элементами, будем находить его индекс и при создании разделителя стоки на данное место будем помещать другой разделяющий символ. Получаем следующее (рисунок 6.13).

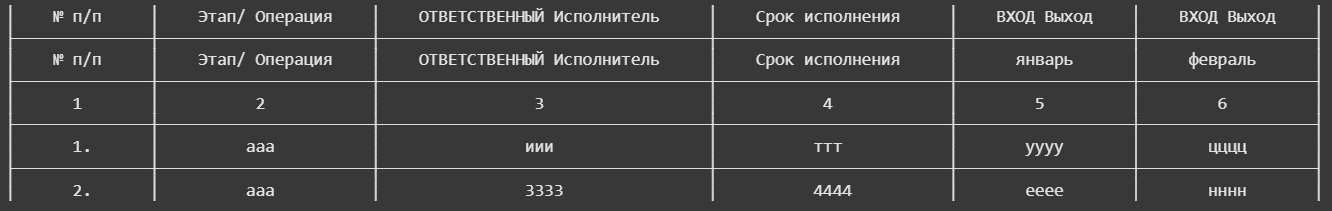


Рисунок 6.13 – Модификация разделителя

Остается добавить верхний и нижний разделитель, отличаются только в элементах разделения.

В результате получаем полноценную таблицу (рисунок 6.14).

Рисунок 6.14 – Полноценная читабельная таблица

И указав название таблицы получим следующее. (рисунок 6.15).

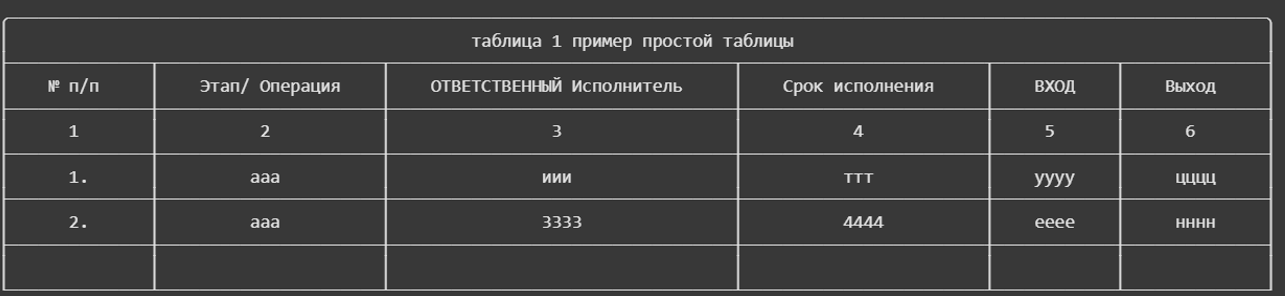


Рисунок 6.15 – Таблица с заголовком

# **Заключение**

В результате производственной практики, осуществленной в ООО "РН-КрасноярскНИПИнефть" и направленной на преддипломную работу, был достигнут значительный прогресс в изучении обработки и форматирования таблиц. Цель практики, заключавшаяся в подготовке к выпускной квалификационной работе, была достигнута благодаря осуществлению ряда задач, охватывающих различные аспекты исследований в области дата инжиниринга.

В ходе практики были рассмотрены и изучены структура компании, ее виды деятельности и процессы управления, что позволило получить полное представление о рабочей среде и условиях работы. Исследование целей и функций автоматизации технологических процессов, а также освоение различных пакетов программ, применяемых в компании, способствовали расширению знаний и навыков в области автоматизации и информационных технологий.

Особое внимание было уделено постановке задачи автоматизации, включающей в себя описание предметной области и разработку требований к программному продукту. Проведенный обзор аналогов для решения поставленной задачи позволил ознакомиться с существующими решениями на рынке и выделить основные преимущества и недостатки.

Наконец, реализация модуля обработки и форматирования таблиц была завершена, что позволило протестировать и оценить эффективность разработанных решений.

# **Список использованных источников**

1. Раскрывая секреты LLM: руководство по основным понятиям больших языковых моделей [Электронный ресурс]: сайт Habr. URL: <https://habr.com/ru/articles/768844/> (дата обращения: 01.05.2025).
2. Business Intelligence и бизнес-аналитика: стратегия, этапы, процессы и инструменты [Электронный ресурс]: сайт Habr. URL: <https://habr.com/ru/articles/714910/> (дата обращения: 01.05.2025).
3. Изучение data science c нуля: этапы и вехи [Электронный ресурс]: сайт Habr. URL: <https://habr.com/ru/articles/538562/> (дата обращения: 01.05.2025).
4. Кто такой Data Engineer [Электронный ресурс]: сайт Habr. URL: <https://habr.com/ru/articles/801971/> (дата обращения: 01.05.2025).
5. Виды аналитиков данных [Электронный ресурс]: сайт Habr. URL: <https://habr.com/ru/companies/yandex_praktikum/articles/852068/> (дата  обраще-ния: 01.05.2025).
6. PyMuPDF4LLM [Электронный ресурс]: сайт Pymupdf. URL: <https://pymupdf.readthedocs.io/en/latest/pymupdf4llm/>         (дата       обращения: 01.05.2025).
7. MegaParse [Электронный ресурс]: сайт GitHub. URL: <https://github.com/QuivrHQ/MegaParse> (дата обращения: 01.05.2025).
8. Markdrop [Электронный ресурс]: сайт PyPI. URL: <https://pypi.org/project/markdrop/> (дата обращения: 01.05.2025).
9. VS Code [Электронный ресурс]: сайт Visual Studio Code. URL: <https://code.visualstudio.com/> (дата обращения: 01.05.2025).
10. UV [Электронный ресурс]: сайт PyPI. URL: <https://pypi.org/project/uv/> (дата обращения: 01.05.2025).
11. Python-docx [Электронный ресурс]: сайт PyPI. URL: <https://pypi.org/project/python-docx/> (дата обращения: 01.05.2025).