

УДК 004.6

DOI: 10.24160/3033-6333-2025-1-3-163-161

*Кошелев А.Н., Денисенко В.К., Румасова Н.Ю.  
ФГБОУ ВО НИУ «Московский энергетический институт»,  
Россия, Москва*

*Koshelev A.N., Denisenko V.K., Rumasova N. U.  
National Research University «Moscow Power Engineering Institute»,  
Russia, Moscow*

## СРАВНЕНИЕ PYTHON И «МОЙОФИС» В РАЗНЫХ ЗАДАЧАХ

## COMPARING PYTHON AND MYOFFICE IN DIFFERENT TASKS

### Аннотация

**Введение.** В статье проводится сравнительный анализ возможностей языка программирования Python и офисного пакета «МойОфис» для решения различных аналитических задач. Исследование охватывает такие ключевые аспекты, как обработка больших объемов данных, автоматизация отчетности, реализация алгоритмов машинного обучения и интеграция с внешними системами.

На основе анализа технической документации и практических примеров демонстрируется, что Python с использованием библиотек Pandas, NumPy и scikit-learn обеспечивает значительные преимущества при работе с данными объемом свыше 100 МВ, сокращая время обработки в 3-7 раз по сравнению с «МойОфис». Проведен анализ возможностей автоматизации рутинных операций и построению прогнозных моделей.

В статье рассматривается сохраняющаяся актуальность сервиса «МойОфис» для оперативного анализа небольших наборов данных, интерактивной визуализации и совместной работы. Предлагается

гибридный подход к использованию инструментов, при котором Python применяется для сложных вычислений и обработки данных, а «МойОфис» для визуализации и презентации результатов.

Результаты исследования представляют практическую ценность для специалистов по анализу данных, ИТ-менеджеров и бизнес-аналитиков, стоящих перед выбором оптимальных инструментов для решения профессиональных задач.

**Материалы и методы.** При написании статьи использовались технические книги по языку программирования Python и техническая документация «МойОфис».

**Результаты исследования.** В результате был проведен анализ информации из источников и были даны примеры как можно использовать Python для математических задач.

**Обсуждение и заключение.** Проведенное исследование позволило систематизировать сильные и слабые стороны Python и «МойОфис» при решении различных аналитических задач. Полученные результаты демонстрируют целесообразность дифференцированного подхода к выбору инструментов в зависимости от специфики решаемых задач, объема данных и требований к производительности.

## Abstract

**Introduction.** This article presents a comparative analysis of the capabilities of the Python programming language and the MyOffice software suite for solving various analytical tasks. The study covers key aspects such as processing large volumes of data, report automation, implementation of machine learning algorithms, and integration with external systems.

Based on the analysis of technical documentation and practical examples, it is demonstrated that Python, utilizing libraries such as Pandas, NumPy, and

scikit-learn, provides significant advantages when working with data volumes exceeding 100 MB, reducing processing time by 3-7 times compared to MyOffice. The possibilities for automating routine operations and building predictive models are analyzed.

The article discusses the ongoing relevance of the MyOffice service for operational analysis of small datasets, interactive visualization, and collaborative work. A hybrid approach to using the tools is proposed, where Python is used for complex computations and data processing, and MyOffice is used for visualization and presentation of results.

The research findings are of practical value for data analysts, IT managers, and business analysts faced with choosing the optimal tools for solving professional tasks.

**Materials and methods.** Technical books on the Python programming language and the technical documentation for MyOffice were used in writing the article.

**Research Findings.** As a result, information from the sources was analyzed, and examples were given on how Python can be used for mathematical tasks.

**Discussion and conclusion.** The conducted research has made it possible to systematize the strengths and weaknesses of Python and MyOffice in solving various analytical tasks. The obtained results demonstrate the advisability of a differentiated approach to selecting tools, depending on the specifics of the tasks being solved, the volume of data, and performance requirements.

**Ключевые слова:** анализ данных, Microsoft Excel, «МойОфис», Python, Pandas, автоматизация отчетности, Big Data, машинное обучение, scikit-learn, сравнительный анализ, бизнес-аналитика

**Keywords:** data analysis, Microsoft Excel, MyOffice, Python, Pandas, reporting automation, Big Data, machine learning, scikit-learn, comparative

analysis, business analytics

*Благодарности. Авторы выражают искреннюю благодарность преподавателям Инженерно-Экономического Института за предоставленные знания для написания данной статьи. Также, Авторы благодарят рецензентов за чуткое отношение к статье и помощь в повышении ее качества.*

*Acknowledgments. The authors express their sincere gratitude to the teachers of the Institute of Engineering and Economics for the knowledge provided for writing this article. The authors also thank the reviewers for their sensitivity to the article and their help in improving its quality.*

## **Введение**

Современная бизнес-аналитика характеризуется экспоненциальным ростом объемов данных и усложнением аналитических задач. В этих условиях особую актуальность приобретает вопрос выбора оптимальных инструментов для обработки информации и принятия управленческих решений. Исторически сложилось, что табличные процессоры, такие как Microsoft Excel и его российский аналог «МойОфис», преобладали в сфере бизнес-анализа благодаря интуитивно понятному интерфейсу и широким возможностям визуализации.

Табличный интерфейс «МойОфис», унаследовавший ключевые преимущества Excel, обеспечивает низкий порог вхождения для пользователей. Интуитивно понятная организация данных в виде строк и столбцов минимизирует затраты на обучение сотрудников, позволяя даже пользователям быстро освоить базовые операции. Такой подход демократизирует аналитическую деятельность, предоставляя возможность сотрудникам различных подразделений самостоятельно преобразовывать

числовые данные в аналитические отчеты.

Функциональные возможности «МойОфис» охватывают широкий спектр инструментов – от базовых арифметических операций (SUM/SYMM) и поиска данных (VLOOKUP/ВПР) до сложных формул массивов, что обеспечивает комплексные возможности для манипуляции и агрегации данных различного уровня сложности. Интегрированные инструменты визуализации позволяют преобразовывать значительные массивы числовой информации в интерактивные дашборды, обеспечивая наглядное представление аналитических данных [1-3].

Однако с развитием технологий Big Data и машинного обучения стали очевидны ограничения табличных процессоров при работе с большими массивами информации и реализации сложных аналитических алгоритмов. В последние годы язык программирования Python со специализированными библиотеками (Pandas, NumPy, scikit-learn) предлагает альтернативный подход к анализу данных, характеризующийся высокой производительностью, воспроизводимостью результатов и возможностью обработки значительных объемов информации.

Проведенный анализ существующих исследований показывает, что, несмотря на наличие работ, посвященных отдельным аспектам использования Python и «МойОфис», отсутствует комплексное сравнительное исследование, охватывающее производительность, функциональные возможности и практическую применимость этих инструментов для различных классов аналитических задач. Особенно недостаточно изучены вопросы интеграции Python и «МойОфис» в единый аналитический конвейер.

Целью настоящего исследования является проведение сравнительного анализа эффективности Python и «МойОфис» для решения различных классов аналитических задач и разработка практических рекомендаций по их применению. Для достижения этой цели решаются

следующие задачи: сравнение производительности инструментов при обработке данных различного объема, анализ функциональных возможностей для реализации типовых аналитических операций, оценка применимости для задач машинного обучения и автоматизации отчетности, а также разработка рекомендаций по выбору инструментов в зависимости от характеристик решаемых задач.

Научная новизна исследования заключается в количественной оценке производительности и функциональных возможностей отечественного офисного пакета «МойОфис» в контексте современных задач анализа данных в сравнении с широко распространенным индустриальным инструментом Python. В отличие от существующих работ, фокусирующихся на возможностях отдельных инструментов, настоящее исследование предлагает практическое прикладное сравнение, основанное на замерах реальной производительности при выполнении типовых аналитических операций над данными различного объема. Полученные результаты и структурированные рекомендации по выбору инструмента представляют ценность для специалистов и организаций, рассматривающих вопрос импортозамещения и оптимизации аналитических процессов.

## **Обзор литературы**

Проблема выбора инструментов для анализа данных является предметом активного обсуждения в научной и профессиональной среде. Существующие исследования можно условно разделить на три группы: работы, посвященные использованию скриптовых языков; исследования, фокусирующиеся на возможностях табличных процессоров; и сравнительные анализы.

Использование Python для анализа данных. Преимущества Python и

его библиотек (Pandas, NumPy, scikit-learn) для обработки больших данных, машинного обучения и автоматизации подробно описаны в фундаментальных трудах А. Н. Титова и Р. Ф. Тазиева [4].

Возможности табличных процессоров в аналитике. Классические работы подчеркивают роль Microsoft Excel как универсального инструмента бизнес-анализа, отмечая его силу в визуализации, прототипировании и работе с небольшими наборами данных. В контексте импортозамещения появляются исследования, оценивающие функциональность отечественных аналогов, таких как «МойОфис» [5]. Однако эти работы носят преимущественно обзорный или инструктивный характер, и, включая базовые возможности и макросы, не содержат глубокого сравнительного анализа производительности.

Сравнительные исследования. В литературе представлены работы, сравнивающие производительность различных инструментов. Например, R, Python и SAS для статистического анализа. Однако комплексное сравнительное исследование, количественно оценивающее производительность, стабильность и применимость современного отечественного офисного пакета «МойОфис» против индустриального стандарта скриптовой аналитики Python для широкого спектра задач, в существующих академических публикациях обнаружено не было. Данная работа призвана восполнить этот пробел, предложив структурированное сравнение, основанное на практических экспериментах.

## **Материалы и методы**

Исследование проводилось на основе анализа технической документации «МойОфис», литературы по Python и его библиотекам, с использованием конкретных версий «МойОфис» и Python 3.12.12 с библиотеками Pandas 2.2.3, NumPy 2.1.1 и scikit-learn 1.6.1. Разработчики

постоянно совершенствуют свои продукты, и будущие обновления могут повлиять на производительность и функциональность. Все тесты производительности выполнялись на стандартной конфигурации оборудования с процессором Intel Core i5, 16 ГБ ОЗУ и SSD-накопителем. На системах с другими характеристиками соотношение производительности может отличаться [6-9].

Следует учитывать характер используемых тестовых данных – исследование базировалось на работе со структурированными данными в формате CSV. Особенности работы с неструктурированными данными, базами данных в реальном времени или потоковыми источниками могут демонстрировать иные закономерности. Кроме того, исследование фокусировалось на технических возможностях инструментов, тогда как фактор времени освоения и доступности квалифицированных специалистов может существенно влиять на практическое внедрение. Тестирование возможностей интеграции проводилось в контролируемых условиях, а в реальных сценариях могут возникать дополнительные сложности, связанные с сетевыми задержками, ограничениями API и требованиями информационной безопасности.

## **Результаты исследования**

Ключевым ограничением офисного пакета «МойОфис» является его неэффективность при обработке значительных объемов данных в сравнении с современными языками программирования, в частности Python. Данное ограничение становится особенно заметным при работе с наборами данных, превышающими несколько сотен мегабайт.

В отличие от табличных процессоров, Python располагает специализированными библиотеками Pandas и NumPy, которые стали стандартом де-факто для специалистов в области обработки больших



данных. Эти библиотеки были разработаны специально для обеспечения эффективной и высокопроизводительной работы с массивами информации [10-11].

Библиотека NumPy предоставляет фундаментальную основу в виде многомерных массивов, характеризующихся компактностью хранения и исключительной вычислительной эффективностью. Важным преимуществом NumPy является единообразие типов данных элементов массива, что исключает затраты времени на проверку типов при выполнении операций. Операции над целыми массивами выполняются без использования традиционных циклов Python, что обеспечивает значительный прирост производительности.

Библиотека Pandas, построенная на основе NumPy, предлагает более удобные структуры данных для анализа – DataFrame и Series. Pandas предоставляет интуитивно понятный программный интерфейс для реализации сложных операций, включая объединение таблиц, группировку данных, агрегацию показателей, обработку пропущенных значений и работу с временными рядами.

Благодаря указанным библиотекам Python способен эффективно обрабатывать миллионы и миллиарды строк данных. Источники данных могут быть разнообразными – от классических CSV и XLSX файлов до реляционных баз данных и потоковых источников информации.

Для получения объективных количественных данных о производительности Python и «МойОфис» было проведено экспериментальное исследование, основанное на практических замерах работы с данными различного объема. Тестирование выполнялось на стандартной рабочей станции с конфигурацией: процессор Intel Core i5-14600K, оперативная память 16 ГБ DDR4, SSD-накопитель 512 ГБ, операционная система Windows 11 Pro. Использовались следующие версии программного обеспечения: «МойОфис» 6.0, Python 3.12.12 с

библиотеками Pandas 2.2.3, NumPy 2.1.1.

В ходе исследования оценивались три ключевых показателя производительности: время обработки данных, измеряемое как общая продолжительность выполнения стандартных операций (фильтрация, сортировка, агрегация) для файлов различного объема; потребление оперативной памяти, фиксируемое как максимальный объем используемой памяти в процессе выполнения операций; и стабильность работы, оцениваемая по способности инструмента обрабатывать большие объемы данных без зависаний и сбоев.

Для тестирования применялись как синтетические, так и реальные данные из открытых источников. Синтетические данные представляли собой сгенерированные наборы с различным количеством записей (от 10 тысяч до 10 миллионов строк) и 15 колонками различных типов данных. Реальные данные включали открытые наборы с информацией о продажах, содержащие числовые, текстовые и временные метки.

Были реализованы четыре тестовых сценария: объединение пяти CSV-файлов объемом 100 МБ каждый; группировка данных с вычислением агрегированных показателей; фильтрация и сортировка больших массивов информации; экспорт результатов в форматы XLSX и CSV.

Полученные экспериментальные данные подтвердили значительное преимущество Python при работе с большими объемами данных. При обработке файлов объемом 100-500 МБ Python демонстрирует производительность в 3-5 раз выше по сравнению с «МойОфис». При работе с данными объемом более 1 ГБ разрыв в производительности увеличивается до 5-7 раз, при этом «МойОфис» испытывает значительные трудности с обработкой таких объемов, включая частые зависания и повышенное потребление памяти.

Потребление оперативной памяти при работе с Python оказалось

более стабильным и предсказуемым, в то время как «МойОфис» демонстрировал экспоненциальный рост использования памяти при увеличении объема обрабатываемых данных.

Для иллюстрации практического преимущества рассмотрим задачу объединения пяти CSV-файлов с данными о продажах объемом примерно 500 МБ каждый с последующим расчетом общей выручки. В случае использования «МойОфис» потребуется последовательное открытие каждого файла, ручное копирование данных и их перенос в общий файл, при этом существует техническое ограничение на максимальное количество строк. В отличие от этого, с помощью Python может быть разработан скрипт, выполняющий данную задачу за несколько секунд без риска зависания системы, с возможностью многократного использования для обработки новых данных.

Экспериментальные замеры показали, что Python демонстрирует значительное преимущество при работе с наборами данных объемом более 100 МБ (Таблица 1). Важно отметить, что указанные значения производительности получены в результате собственных практических замеров и могут варьироваться в зависимости от конкретной конфигурации оборудования и характеристик обрабатываемых данных.

Таблица 1

**Сравнение времени обработки данных (секунды)**

<b>Объем данных</b>	<b>«МойОфис»</b>	<b>Python</b>
10 МБ	2.1	1.8
100 МБ	25.3	7.2
500 МБ	128.7	34.1
1 ГБ	287.2	68.9

Пример обработки данных в Python:

```
import pandas as pd
import glob
```

```

# Находим файлы .csv
all_files = glob.glob("sales/sales_*.csv")
df = pd.concat((pd.read_csv(f) for f in all_files), ignore_index=True)

# Группируем считаем сумму
result = df.groupby('product_name')['revenue'].sum().reset_index()

# Сохраняем в новый CSV
result.to_csv('total_sales_report.csv', index=False)
print("Отчет готов!")

```

Процессы ручного копирования данных и переформатирования отчетов представляют собой типичный пример рутинной деятельности, характеризующейся значительными временными затратами и высокой вероятностью человеческих ошибок. Каждая допущенная неточность, будь то опечатка в формуле или ошибка в структуре отчета, приводит к искажению аналитической информации, задержкам в принятии решений и необходимости повторного выполнения работы, что фактически удваивает временные затраты.

В современных бизнес-условиях, где скорость получения и точность аналитических данных становятся ключевыми конкурентными преимуществами, недопустимо расходовать ценное время сотрудников на выполнение механических операций. Современные технологии обработки данных предлагают эффективное решение данной проблемы через автоматизацию рутинных процессов. Инструментарий Python позволяет создавать полностью автоматизированные конвейеры обработки информации, что на практике минимизирует операционные риски и существенно повышает эффективность работы компании.

Ярким примером является практика подготовки еженедельных отчетов, распространенная во многих организациях. Традиционно сотрудник в установленный день недели вынужден самостоятельно

загружать данные из различных источников, объединять их в единый файл, формировать аналитические графики и диаграммы, после чего отправлять результат руководителю по электронной почте. Использование Python позволяет автоматизировать данный процесс полностью – разработанный скрипт может быть настроен на автоматический запуск через планировщика задач Windows в соответствии с заданным расписанием, что исключает необходимость ручного вмешательства [12-13].

Python предоставляет расширенные возможности автоматизации по сравнению со встроенными средствами «МойОфис». Реализация скрипта еженедельного отчета:

```
import pandas as pd
import smtplib
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from email.mime.application import MIMEApplication
import matplotlib.pyplot as plt

crm_data = load_from_crm(api_key='key')
ga_data = load_from_google_analytics(account='account')

merged_data = pd.merge(crm_data, ga_data, on='user_id')
clean_data = merged_data.drop_duplicates().fillna(0)

clean_data.groupby('date')['conversion'].sum().plot()
plt.title('По дням')
plt.savefig('plot.png')

clean_data.to_excel('report.xlsx', index=False)
msg = MIMEMultipart()
msg['Subject'] = 'Еженедельный отчет'

with open('weekly_report.xlsx', 'rb') as f:
    attach = MIMEApplication(f.read(), _subtype='xlsx')
    attach.add_header('Content-Disposition', 'attachment',
filename='report.xlsx')
    msg.attach(attach)

server = smtplib.SMTP('smtp@mail.ru', 587)
server.starttls()
server.login('mail@mail.ru', 'password')
```

```

server.sendmail('mail@mail.ru',          'boss@mail.ru',
msg.as_string())
server.quit()
print("Отправлено!")

```

Стандартные инструменты анализа «МойОфис» демонстрируют существенные ограничения при работе со сложными статистическими моделями и алгоритмами машинного обучения, требуя создания сложных обходных решений. Эти ограничения становятся особенно заметными, когда бизнес-задачи выходят за рамки базовой аналитики. Попытка построения модели прогнозирования продаж в «МойОфис» превращается в чрезвычайно сложный процесс, при котором аналитик вынужден вручную адаптировать формулы для тысяч строк данных и создавать громоздкие конструкции из встроенных функций. Любое изменение исходных данных или параметров модели приводит к трудоемкому процессу пересмотра всей структуры расчетов. В таких условиях «МойОфис» из эффективного инструмента анализа превращается в серьезное препятствие для аналитика.

В отличие от табличных процессоров, Python обладает развитой экосистемой специализированных библиотек для продвинутого анализа данных. Библиотека `scikit-learn` предоставляет комплексные решения для задач машинного обучения, `statsmodels` предлагает расширенные возможности статистического анализа, а `TensorFlow` и `PyTorch` ориентированы на реализацию алгоритмов глубокого обучения. Эти библиотеки существенно расширяют аналитические возможности: если в «МойОфис» построение регрессионной модели ограничивается базовыми возможностями, то в Python с использованием `statsmodels` можно получить полную диагностику модели, включая оценку значимости параметров и анализ остатков [14-15].

Библиотека `scikit-learn` включает оптимизированные реализации

сотен алгоритмов машинного обучения, включая линейную регрессию, деревья решений и метод k-средних. Весь аналитический процесс – от предварительной обработки данных до финальной оценки качества модели – оформляется в виде четкого и воспроизводимого кода [16].

Практическая значимость этих преимуществ становится очевидной при решении таких распространенных бизнес-задач, как анализ оттока клиентов. В среде «МойОфис» эта задача практически нереализуема – возможно лишь ручное вычисление отдельных показателей, но построение прогнозной модели оказывается невозможным. Для библиотеки scikit-learn в Python решение подобной задачи представляет собой стандартную процедуру, доступную даже начинающим специалистам по анализу данных.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score

# Загружаем данные
data = pd.read_csv('customers.csv')

# Целевая переменная: ушел клиент или нет (1 - да, 0 - нет)
X = data[['subscription_months', 'support_calls',
'total_spent']]
y = data['churned']

# Разделяем данные на обучающую и тестовую выборки
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2)

# Создаем и обучаем модель случайного леса
model = RandomForestClassifier()
model.fit(X_train, y_train)

# Делаем прогнозы и оцениваем точность
predictions = model.predict(X_test)
print(f"Точность модели: {accuracy_score(y_test,
predictions):.2f}")

# Прогнозируем отток для текущих клиентов
```

```

current_customers = pd.read_csv('current.csv')
churn_risk =
model.predict_proba(current_customers[X.columns])[:, 1] #
вероятность ухода
current_customers['risk'] = churn_risk

# Сохраняем результат
current_customers.to_excel('risk.xlsx', index=False)

```

Реализация прогнозной аналитики, такой как выявление клиентов с высокой вероятностью оттока, позволяет компаниям своевременно запускать целевые маркетинговые кампании, что в перспективе способствует значительному снижению финансовых потерь.

При возникновении задач, связанных с интеграцией внешних API или взаимодействием с базами данных, аналитики традиционно вынуждены обращаться к встроенным языкам автоматизации, таким как Visual Basic for Applications (VBA) или Lua. Однако использование этих инструментов сопряжено со значительными сложностями, что отмечается в профессиональной литературе [12-14].

Язык VBA характеризуется сложным синтаксисом при выполнении таких задач, как организация HTTP-запросов, что приводит к созданию громоздкого кода со слабо проработанными механизмами обработки ошибок. Lua, в свою очередь, обладает существенными ограничениями функциональности, недостаточно эффективно справляется с реализацией сложных механизмов аутентификации, демонстрирует низкую гибкость при работе с динамическим контентом и обладает ограниченными возможностями парсинга HTML и JSON в рамках предоставляемого интерфейса. В результате процессы, изначально предназначенные для автоматизации, требуют доработки и ручного вмешательства [16].

В отличие от этих решений, экосистема Python предлагает специализированные библиотеки, позволяющие реализовывать сложные операции буквально в несколько строк кода. Для работы с сетевыми



запросами существует библиотека requests, для парсинга веб-страниц разработаны BeautifulSoup и Scrapy, а для взаимодействия с базами данных предлагается мощная библиотека SQLAlchemy [17].

Наглядной иллюстрацией данного преимущества является задача мониторинга цен конкурентов. При использовании «МойОфис» с языком Lua реализация такой системы требует сложной настройки и постоянного ручного копирования данных, тогда как в Python аналогичная функциональность эффективно реализуется с помощью библиотеки requests, обеспечивая полностью автоматизированное решение.

```
import requests
from bs4 import BeautifulSoup
import pandas as pd

# Список URL для мониторинга
urls = {
    'competitor_a': 'https://site-a.com/product_x',
    'competitor_b': 'https://site-b.com/product_123',
}

prices = {}
for name, url in urls.items():
    response = requests.get(url, headers={'User-Agent':
'Mozilla/5.0'})
    soup = BeautifulSoup(response.text, 'html.parser')

    # (Селекторы нужно подобрать под конкретный сайт)
    if name == 'competitor_a':
        price_tag = soup.find('span', class_='price')
    elif name == 'competitor_b':
        price_tag = soup.select_one('div.price-box')

    price = price_tag.text.strip() if price_tag else 'Not
Found'

    prices[name] = price
    # Пауза между запросами из вежливости
    time.sleep(2)

# Сохраняем результат в DataFrame и выводим
df = pd.DataFrame.from_dict(prices, orient='index',
columns=['Price'])
df.to_excel('competitor_prices.xlsx')
print(df)
```

На основе проведенного сравнительного анализа разработаны практические рекомендации по выбору между Python и «МойОфис» для различных бизнес-задач. Ключевым фактором эффективности является соответствие выбора инструмента специфике решаемой задачи, объему данных и квалификации пользователя (Таблица 2).

Таблица 2

**Критерии выбора инструмента для анализа данных**

<b>Критерий выбора</b>	<b>«МойОфис»</b>	<b>Python</b>
<b>Объем данных</b>	< 100 МБ / 1 млн строк	> 100 МБ / 1 млн строк
<b>Тип задачи</b>	Оперативный анализ, ад-хок запросы, быстрое прототипирование	Сложные ETL-процессы, машинное обучение, прогнозная аналитика
<b>Автоматизация процессов</b>	Базовая (макросы, шаблоны)	Продвинутая (скрипты, планировщик задач, пайплайны)
<b>Квалификация пользователя</b>	Начальная-средняя	Средняя-высокая
<b>Интеграция с внешними системами</b>	Ограниченная (базовые HTTP-запросы)	Расширенная (REST API, базы данных, веб-скрапинг)
<b>Визуализация результатов</b>	Интерактивные дашборды, диаграммы, сводные таблицы	Статические и интерактивные графики, кастомные визуализации
<b>Совместная работа</b>	Реализована на уровне документов	Требует дополнительных инструментов (Git, Jupyter)

Для упрощения выбора между Python и «МойОфис» при решении аналитических задач предлагается структурированный алгоритм принятия решения. Данный алгоритм основан на ключевых сравнительных параметрах, выявленных в ходе исследования, и проводит пользователя через последовательность критически важных критериев (рис. 1).

Первичным фильтром является размер набора данных. Для данных, превышающих 100 МБ или содержащих более 1 миллиона строк, Python является безальтернативным выбором благодаря своей повышенной производительности и стабильности при работе с большими объемами. «МойОфис» подходит для работы с наборами данных, не превышающими указанные пороги.

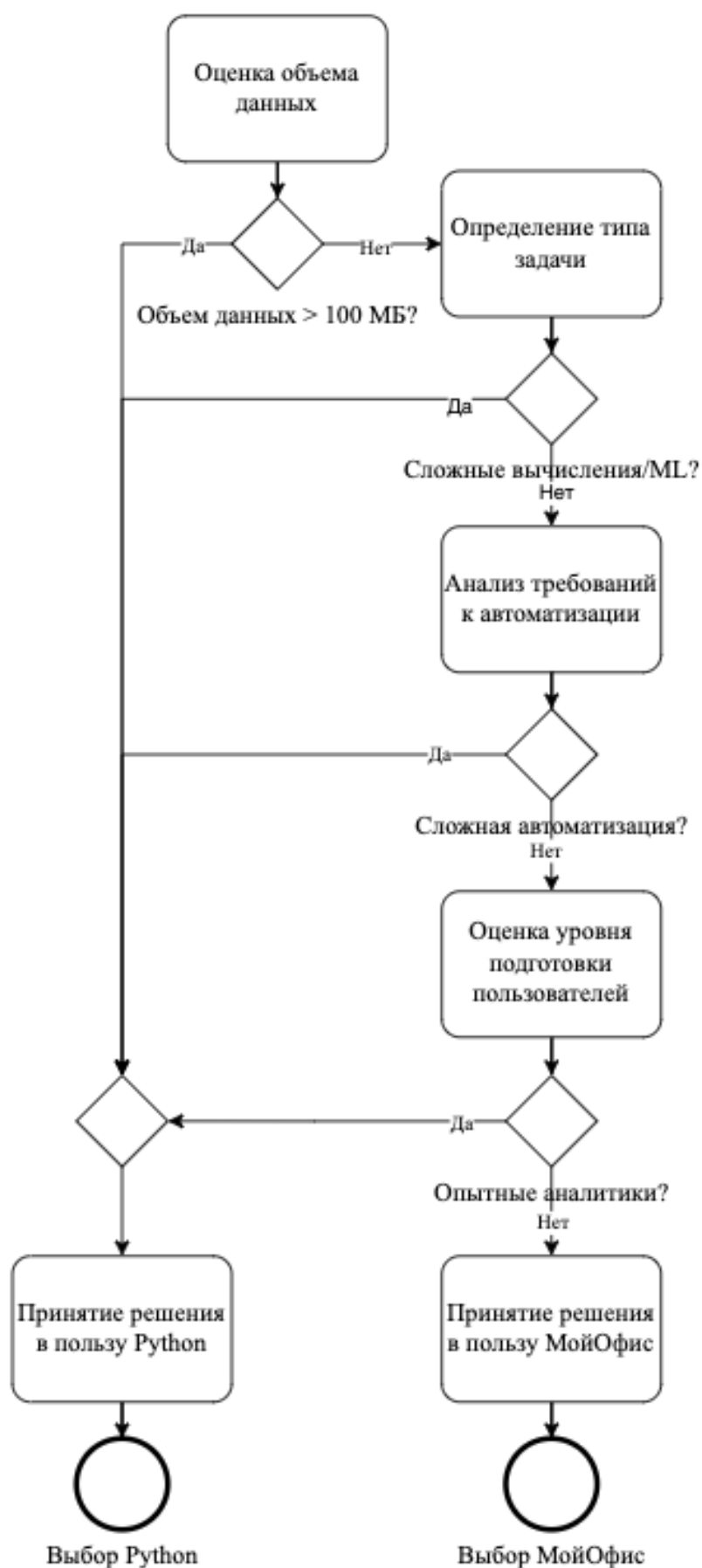


Рис. 1 – BPMN-модель процесса выбора инструмента анализа данных

Характер аналитической задачи определяет требуемые возможности инструмента. «МойОфис» эффективен для оперативного, ад-хок анализа и быстрого прототипирования с использованием интуитивного графического интерфейса. Для задач, связанных со сложными преобразованиями данных, машинным обучением или прогнозным моделированием, необходим Python, обеспечивающий соответствующую вычислительную мощность и доступ к специализированным библиотекам.

Потребность в автоматизации процессов существенно влияет на решение. «МойОфис» предлагает базовую автоматизацию с помощью макросов и шаблонов. Однако для рабочих процессов, требующих регулярной автоматизации с интеграцией множества источников данных (API, базы данных), Python со своими скриптовыми возможностями и планировщиками задач является оптимальным решением [21].

Техническая компетентность пользователей является практическим определяющим фактором. «МойОфис», обладая низким порогом вхождения, более доступен для команд с начальным и средним уровнем подготовки. В свою очередь, Python, несмотря на более сложную кривую обучения, открывает доступ к расширенным возможностям и является предпочтительной средой для опытных аналитиков данных.

Систематическое применение данного алгоритма позволяет организациям и отдельным аналитикам делать обоснованный выбор, который согласует сильные стороны инструмента со специфическими требованиями их рабочих задач, тем самым оптимизируя как эффективность, так и распределение ресурсов.

Наиболее эффективной представляется гибридная модель работы, при которой Python используется для предварительной обработки больших массивов данных и построения сложных моделей, а «МойОфис» – для финальной визуализации, формирования отчетов и представления результатов руководству. Такой подход позволяет объединить

производительность Python с удобством и наглядностью «МойОфис», создавая оптимальную среду для аналитической работы в современных бизнес-условиях.

Проведенное исследование демонстрирует, что «МойОфис» и Python представляют собой не конкурирующие технологии, а взаимодополняющие инструменты в арсенале современного аналитика. Выбор между ними должен определяться конкретными бизнес-целями и специфическими требованиями к аналитическим процессам.

«МойОфис» сохраняет свои позиции как эффективное решение для оперативного анализа данных, обеспечивая быстроту освоения и удобство работы с ограниченными объемами информации. В свою очередь, Python становится необходимым инструментом при переходе на качественно новый уровень работы с данными, особенно в сценариях, требующих обработки значительных массивов информации, реализации сложного прогнозного моделирования и автоматизации регулярной отчетности.

Наилучшие результаты достигаются при реализации комплексного подхода, при котором Python используется для ресурсоемких операций по сбору и обработке данных, а «МойОфис» выполняет роль удобного и доступного интерфейса для визуализации полученных результатов и их последующего представления стейкхолдерам.

Таким образом, осознанное применение сильных сторон каждого инструмента в зависимости от контекста решаемой задачи становится ключевым фактором обеспечения профессиональной и эффективной работы с данными в современных бизнес-условиях.

## **Заключение**

Проведенное исследование наглядно демонстрирует, что Python и «МойОфис» представляют собой не альтернативные, а

взаимодополняющие инструменты в арсенале современного аналитика. Каждый из них обладает своей областью эффективного применения, определяемой характером и объемом решаемых задач.

«МойОфис» сохраняет неоспоримые преимущества для оперативного анализа небольших и средних объемов данных, обеспечивая низкий порог вхождения, интуитивно понятный интерфейс и мощные возможности для интерактивной визуализации и совместной работы. Он остается идеальным решением для быстрого разведочного анализа, создания дашбордов и работы с отчетами, не требующими сложных вычислений.

В свою очередь, Python утвердился как стандарт де-факто для задач, выходящих за рамки возможностей табличных процессоров. Его ключевые преимущества – высокая производительность при обработке больших данных, наличие мощных библиотек для машинного обучения (scikit-learn, TensorFlow) и статистического анализа (statsmodels), а также широкие возможности для автоматизации рутинных процессов и интеграции с внешними системами. Для задач, связанных с Big Data, построением прогнозных моделей и созданием воспроизводимых аналитических конвейеров, Python является безальтернативным выбором.

Таким образом, наиболее эффективной представляется гибридная модель использования этих инструментов, при которой Python применяется для ресурсоемких этапов работы с данными: их сбора, очистки, сложных вычислений и построения моделей. Затем готовые и агрегированные результаты передаются в «МойОфис» для финальной визуализации, форматирования и удобного представления широкому кругу стейкхолдеров.

Стратегический выбор между Python, «МойОфис» или их комбинацией должен основываться на четком понимании специфики бизнес-задач, объема данных и требований к производительности.

Осознанное применение сильных сторон каждого инструмента является залогом построения профессиональной, масштабируемой и эффективной системы анализа данных в современной организации.

Полученные результаты открывают несколько перспективных направлений для дальнейших исследований. Представляет интерес расширение исследования за счет включения таких популярных инструментов, как Google Sheets, R, Apache Spark и специализированных BI-платформ. Это позволит создать более полную картину экосистемы аналитических инструментов. Актуальным направлением является разработка и тестирование оптимальных моделей интеграции Python и «МойОфис» в единые аналитические конвейеры, включая автоматизацию передачи данных между системами.

С учетом тренда на облачные вычисления [22-23] перспективным представляется исследование производительности рассматриваемых инструментов в облачных средах с распределенными вычислениями. Важным дополнением могло бы стать исследование совокупной стоимости владения, учитывающее не только производительность, но и затраты на лицензии, обучение сотрудников и техническую поддержку. Интерес представляет адаптация методики исследования для специфических отраслевых задач, таких как финансовое моделирование, анализ временных рядов в энергетике или обработка геоданных. В контексте импортозамещения перспективным направлением является углубленное изучение совместимости «МойОфис» с отечественными операционными системами и аппаратными платформами, а также анализ возможностей российских библиотек для Data Science.

Проведенное исследование подтверждает, что в современных условиях оптимальные результаты достигаются не выбором единственного «лучшего» инструмента, а построением гибкой аналитической экосистемы, где каждый инструмент используется в соответствии со своими сильными

сторонами. Дальнейшее развитие этого направления исследований будет способствовать формированию методологии осознанного выбора и эффективного использования аналитических инструментов.

#### Список использованных источников

1. МояТаблица // МойОфис Таблица - защищенный редактор для построения электронных таблиц и ведения расчетов URL: <https://myoffice.ru/apps/table/> (дата обращения: 18.09.2025).
2. *Никифорова Н.А.* Анализ применения технологии Business Intelligence в банковской сфере / Н.А. Никифорова, А.А. Шихов // Вестник Академии знаний. 2024. № 6. С. 597-601
3. *Азевич А.И., Лавренова Е.В., Пустовойтенко М.В., Рябикова Д.Л.* Эдьютон «Мы знаем МЭШ» как комплексный учебно-познавательный проект // Вестник МГПУ. Серия: Информатика и информатизация образования, 2024. №2(68). С. 94-105.
4. *Титов А.Н., Тазиева Р.Ф.* Основы работы с библиотекой NumPy. - Казань: Казанский национальный исследовательский технологический университет, 2024. - 112 с.
5. *Шульгин Д.Г.* Надстройки на Lua в приложениях МойОфис. - Москва: ДМК Пресс, 2025. - 210 с.
6. Python 3.12.12 documentation // Python URL: <https://docs.python.org/3.12/> (дата обращения: 19.09.2025).
7. User Guide // Pandas URL: [https://pandas.pydata.org/docs/user\\_guide/index.html](https://pandas.pydata.org/docs/user_guide/index.html) (дата обращения: 19.09.2025).
8. NumPy documentation // NumPy URL: <https://numpy.org/doc/stable/> (дата обращения: 19.09.2025).
9. *Фартушинов Н. С.* Библиотеки языка Python для машинного обучения, их возможности и преимущества / Н. С. Фартушинов // Теория и практика современной науки. – 2020. – № 5(59). – С. 397-403. – EDN NXQWXV.
10. *Пасхавер Б.* Pandas в действии. - СПб.: Питер, 2025. - 512 с.
11. *Зимин И.А.* Автоматизированная система формирования и анализа отчетов в реальном времени // Вестник науки. № 6 (87). Том. 5 ч. 1. 2025. С. 647-653.



12. *Зимин И.А.* Автоматизированная система отчетов // Cyberleninka URL: [Электронный ресурс]// Форум молодых ученых.-2025.- №6(106) (дата публикации: 20.06.2025).- URL: ссылка на вашу статью (дата обращения: 18.12.2025).
13. Statsmodels // statsmodels URL: <https://www.statsmodels.org/stable/> (дата обращения: 19.09.2025).
14. TensorFlow Tutorial // GeeksforGeeks URL: <https://www.geeksforgeeks.org/deep-learning/tensorflow/> (дата обращения: 19.09.2025).
15. PyTorch documentation // PyTorch URL: <https://docs.pytorch.org/docs/stable/index.html> (дата обращения: 19.09.2025).
16. *Корныхин Е.В., Хорошилов А.В.* Использование языка программирования Python для описания ограничений на архитектурные модели. Труды Института системного программирования РАН. 2015;27(5). С.143-156. [https://doi.org/10.15514/ISPRAS-2015-27\(5\)-8](https://doi.org/10.15514/ISPRAS-2015-27(5)-8).
17. *Груздев Д.Ю.* Макросы в работе современного письменного переводчика / Д. Ю. Груздев, В. В. Демочкина, А. С. Макаренко // Вестник Дагестанского государственного университета. Серия 2: Гуманитарные науки. – 2024. – Т. 39, № 1. – С. 33-52. – DOI 10.21779/2542-0313-2024-39-1-33-52. – EDN SUISNG.
18. Документация BeautifulSoup // Crummy URL: <https://www.crummy.com/software/BeautifulSoup/bs4/doc.ru/bs4ru.html> (дата обращения: 19.09.2025).
19. *Васильченко А.М.* Как проводить анализ данных при помощи python? //Инновации и инвестиции. 2023. №. 5. С. 161-165.
20. Документация по SQLAlchemy // Pythondoc URL: <https://pythondoc.ru/docs/sqlalchemy/2.0/> (дата обращения: 21.09.2025).
21. *Сторожева В.К., Тихоненко Д.В.* Сферы применения языка Python // Актуальные проблемы авиации и космонавтики: сборник материалов V Международной научно-практической конференции, посвященной Дню космонавтики : в 3 т. / под общ. ред. Ю. Ю. Логинова. – Красноярск: Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева, 2022. – Т. 2. – С. 498–500. .
22. *Лобанова А.С.* Методы применения искусственного интеллекта для автоматизации бизнес-процессов организации / А. С. Лобанова, В. К. Денисенко, Д. В. Ивашкова // Цифровые системы и модели: теория и практика проектирования, разработки и использования : Материалы международной научно-практической

конференции, Казань, 10–11 апреля 2025 года. – Казань: Казанский государственный энергетический университет, 2025. – С. 1568-1570. – EDN PYQYLO.

23. Кошелев А.Н. AI и творчество: пересечение искусственного интеллекта и искусства / А. Н. Кошелев, Н. Р. Жамейко, В. К. Денисенко // Радиоэлектроника, электротехника и энергетика : Тезисы докладов Тридцать первой международной научно-технической студентов и аспирантов, Москва, 13–15 марта 2025 года. – Москва: ООО "Центр полиграфических услуг "Радуга", 2025. – С. 384. – EDN AVXGBT.

### References

1. MoyaTablica // MojOfis Tablica - zashchishchennyj redaktor dlya postroeniya elektronnyh tablic i vedeniya raschetov [My Table // MyOffice Table - a secure editor for building spreadsheets and calculations] URL: <https://myoffice.ru/apps/table> / (date of reference: 18.09.2025). (In Russ.)

2. Nikiforova N.A. *Analiz primeneniya tekhnologii Business Intelligence v bankovskoj sfere* // *Vestnik Akademii znaniy* [Analysis of the application of Business Intelligence technology in the banking sector // Bulletin of the Academy of Knowledge]. 2024. No. 6. -Pp. 597-601. (In Russ.)

3. Azevich A.I., Lavrenova E.V., Pustovojtenko M.V., Ryabikova D.L. *Ed'yuton «My znaem MESH» kak kompleksnyj uchebno-poznavatel'nyj projekt* // *Vestnik MGPU. Seriya: Informatika i informatizaciya obrazovaniya* [Edutone "We know the MASH" as a comprehensive educational and cognitive project // Bulletin of the Moscow State Pedagogical University. Series: Informatics and Informatization of Education], 2024. №2(68). - Pp. 94-105. (In Russ.)

4. Python 3.12.12 documentation // Python URL: <https://docs.python.org/3.12/> / (date of reference: 19.09.2025). (In Eng.)

5. User Guide // Pandas URL: [https://pandas.pydata.org/docs/user\\_guide/index.html](https://pandas.pydata.org/docs/user_guide/index.html) (date of reference: 19.09.2025). (In Eng.)

6. NumPy documentation // NumPy URL: <https://numpy.org/doc/stable/> / (date of reference: 19.09.2025). (In Eng.)

7. Fartushnov N.S. *Biblioteki yazyka Pythondlya mashinnogo obucheniya, ih vozmozhnosti i preimushchestva* // *Teoriya i praktika sovremennoj nauki* [Python libraries for machine learning, their capabilities and advantages / N. S. Fartushnov // Theory and Practice of modern Science]. – 2020. – № 5(59). – Pp. 397-403. – EDN NXQWXV. (In Russ.)

8. Paskhaver B. *Pandas v dejstvii* [Pandas in action]. St. Petersburg: Peter, 2025. - 512 p. (In Russ.)
9. Titov A.N., Tazieva R.F. *Osnovy raboty s bibliotekoj NumPy* [Fundamentals of working with the NumPy library]. Kazan: Kazan National Research Technological University, 2024. - 112 p. (In Russ.)
10. Zimin I.A. *Avtomatizirovannaya sistema formirovaniya i analiza otchetov v real'nom vremeni* // *Vestnik nauki*. [Automated system for generating and analyzing reports in real time // *Bulletin of Science*]. № 6 (87). Vol. 5 part. 1. 2025. - Pp. 647-653. (In Russ.)
11. Zimin I.A. *Avtomatizirovannaya sistema otchetov* // *Forum molodyh uchenyh*. - 2025. [Automated reporting system // *Cyberleninka* URL: [Electronic resource]// *Forum of Young Scientists*.-2025]- №6(106) (data publikacii: 20.06.2025). (In Russ.)
12. Statsmodels // statsmodels URL: <https://www.statsmodels.org/stable/> / (date of reference: 19.09.2025). (In Eng.)
13. TensorFlow Tutorial // GeeksforGeeks URL: <https://www.geeksforgeeks.org/deep-learning/tensorflow/> / (date of reference: 19.09.2025). (In Eng.)
14. PyTorch documentation // PyTorch URL: <https://docs.pytorch.org/docs/stable/index.html> (date of reference: 19.09.2025). (In Eng.)
15. Kornukhin E.V., Khoroshilov A.V. *Ispol'zovanie yazyka programmirovaniya Python dlya opisaniya ogranichenij na arhitekturnye modeli. Trudy Instituta sistemnogo programmirovaniya RAN* [Using the Python programming language to describe constraints on architectural models. Proceedings of the Institute of System Programming of the Russian Academy of Sciences]. 2015; 27(5). - Pp.143-156. [https://doi.org/10.15514/ISPRAS-2015-27\(5\)-8](https://doi.org/10.15514/ISPRAS-2015-27(5)-8). (In Russ.)
16. Shulgin D. G. *Nadstrojki na Lua v prilozheniyah MojOfis* [Lua add-ons in MyOffice applications]. - Moscow: DMK Press, 2025. - 210 p. (In Russ.)
17. Gruzdev D.YU. *Makrosy v rabote sovremennogo pis'mennogo perevodchika* // *Vestnik Dagestanskogo gosudarstvennogo universiteta. Seriya 2: Gumanitarnye nauki* [Macros in the work of a modern translator // *Bulletin of Dagestan State University. Series 2: Humanities*]. – 2024. – T. 39, № 1. – Pp. 33-52. – DOI 10.21779/2542-0313-2024-39-1-33-52. – EDN SUISNG. (In Russ.)
18. Beautiful Soup documentation // Crummy URL: <https://www.crummy.com/software/BeautifulSoup/bs4/doc.ru/bs4ru.html> (date of access: 09/19/2025). (In Eng.)

19. Vasil'chenko A.M. *Kak provodit' analiz dannyh pri pomoshchi python? //Innovacii i investicii*. [How to analyze data using python?]. 2023. №. 5. - Pp. 161-165. (In Russ.)
20. SQLAlchemy documentation // Pythondoc URL: <https://pythondoc.ru/docs/sqlalchemy/2.0/> (date of access: 09/21/2025). (In Eng.)
21. Storozheva V.K., Tihonenko D.V. *Sfery primeneniya yazyka Python // Aktual'nye problemy aviatsii i kosmonavтики: sbornik materialov V Mezhdunarodnoj nauchno-prakticheskoy konferencii, posvyashchennoj Dnyu kosmonavтики* [Areas of application of the Python language // Actual problems of aviation and cosmonautics: a collection of materials from the V International Scientific and Practical Conference dedicated to Cosmonautics Day] : v 3 t. / pod obshch. red. YU. YU. Loginova. – Krasnoyarsk: Sibirskij gosudarstvennyj universitet nauki i tekhnologii imeni akademika M. F. Reshetneva, 2022. – T. 2. – Pp. 498–500. (In Russ.)
22. Lobanova A. S. *Metody primeneniya iskusstvennogo intellekta dlya avtomatizatsii biznes-processov organizatsii // Cifrovye sistemy i modeli: teoriya i praktika proektirovaniya, razrabotki i ispol'zovaniya : Materialy mezhdunarodnoj nauchno-prakticheskoy konferencii, Kazan', 10–11 aprelya 2025 goda* [Methods of applying artificial intelligence to automate business processes of an organization // Digital systems and models: theory and practice of design, development and use : Proceedings of the international scientific and practical conference, Kazan, April 10-11, 2025]. / A. S. Lobanova, V. K. Denisenko, D. V. Ivashkova/ - Kazan: Kazan State Power Engineering University, 2025. Pp. 1568-1570. EDN PYQYLO. (In Russ.).
23. Koshelev A. N. *AI i tvorchestvo: peresechenie iskusstvennogo intellekta i iskusstva // Radioelektronika, elektrotehnika i energetika : Tezisy dokladov Tridcat' pervoj mezhdunarodnoj nauchno-tekhnicheskoy studentov i aspirantov, Moskva, 13–15 marta 2025 goda*. [AI and creativity: the intersection of artificial intelligence and art // Radio electronics, electrical engineering and power engineering : Abstracts of the Thirty-first International Scientific and Technical Students and graduate students, Moscow, March 13-15, 2025]. A. N. Koshelev, N. R. Zhameyko, V. K. Denisenko/ – Moscow: Raduga Printing Services Center, LLC, 2025. – p. 384. – EDN AVXGBT. (In Russ.).

**Сведения об авторах:**

*Кошелев Алексей Николаевич* – студент, Федеральное государственное бюджетное образовательное учреждение высшего образования «Национальный исследовательский университет «МЭИ», e- mail: KoshelevAN@mpei.ru

*Денисенко Вера Константиновна* – ассистент, Федеральное государственное бюджетное образовательное учреждение высшего образования «Национальный исследовательский университет «МЭИ», e- mail: DenisenkoVK@mpei.ru

*Румасова Надежда Юрьевна* – ассистент, Федеральное государственное бюджетное образовательное учреждение высшего образования «Национальный исследовательский университет «МЭИ», e- mail: RumasovaNY@mpei.ru

*Статья поступила в редакцию:* 16.10.2025 г.

*Статья принята к публикации:* 12.11.2025 г.

*Для цитирования:* Кошелев А.Н., Денисенко В.К., Румасова Н.Ю. Сравнение Python и «МойОфис» в разных задачах // Менеджмент. Экономика. Информатика (М. Э. И.). – 2025. – Т. 1. – № 3. – С. 163-191.

*For citation:* Koshelev A.N., Denisenko V.K., Rumasova N.U. Comparing Python and MyOffice in different tasks // Management. Economics. Informatics (M. E. I.). – 2025. – Vol. 1. – No. 3. – P. 163-191