



colab

Что такое Colab?

Colaboratory, или просто Colab, позволяет писать и выполнять код Python в браузере. При этом:
не требуется никакой настройки;
бесплатный доступ к графическим процессорам;
предоставлять доступ к документам другим людям очень просто.

Colab – это отличное решение для **учащихся, специалистов по обработке данных и исследователей в области искусственного интеллекта**. Чтобы узнать больше, посмотрите видео с [общей информацией о Colab](#) или [описанием функций сервиса](#). Вы также можете просто начать работу с инструментом ниже.

При работе с Python часто нужно использовать различные библиотеки — специальные наборы готового кода, которые расширяют возможности языка.

Например, для анализа данных применяется библиотека [Pandas](#),
для создания графиков — [Matplotlib](#),
а для машинного обучения — [Scikit-learn](#) или [TensorFlow](#).

Каждая из таких библиотек добавляет функции, но может создавать трудности при установке. Google Colab помогает это решить.

Предположим, вы решили поработать с библиотекой TensorFlow. При её установке в Windows часто возникают проблемы с совместимостью версий [CUDA](#) и [cuDNN](#) — компонентов для работы с графическим процессором.

Для решения этих проблем обычно требуется переустановка Python, обновление драйверов видеокарты или установка определённых версий библиотек.

Вместо этого можно использовать бесплатный сервис Google Colab, где TensorFlow и другие библиотеки уже предустановлены и настроены для работы с графическим процессором.

Вы сразу сможете обучать свои модели и не тратить время на настройку окружения. Более того, сервис работает независимо от мощности вашего компьютера и типа операционной системы. В этой статье мы познакомимся с Google Colab и рассмотрим, для кого он предназначен, какими возможностями обладает и как начать с ним работу.

Содержание

[Что такое Google Colab](#)

[Кому и для чего он полезен](#)

[Как работать с Google Colab](#)

[Альтернативные сервисы](#)

[Что дальше](#)

Google Colab (Google Colaboratory) — это бесплатный облачный сервис для программирования на Python. Он похож на Google Docs: как Docs позволяет создавать документы без установки Microsoft Word, так и Colab даёт возможность писать код без установки Python и дополнительных библиотек.

Google Colab построен на основе Jupyter Notebook и имеет схожий интерфейс. Jupyter Notebook — удобная среда разработки для Python, где можно писать код и сразу видеть результаты его выполнения. Различие в том, что Colab работает в облаке, предоставляет доступ к вычислительным ресурсам Google и обеспечивает интеграцию с различными платформами.

Вот примеры сервисов, в которые интегрирован Google Colab:

[Google Drive](#): обеспечивает прямой доступ к файлам и сохранение блокнотов. К примеру, вы можете загрузить CSV-файл с данными о продажах на Google Drive, проанализировать его в Colab, а затем сохранить готовую модель прогнозирования обратно на диск.

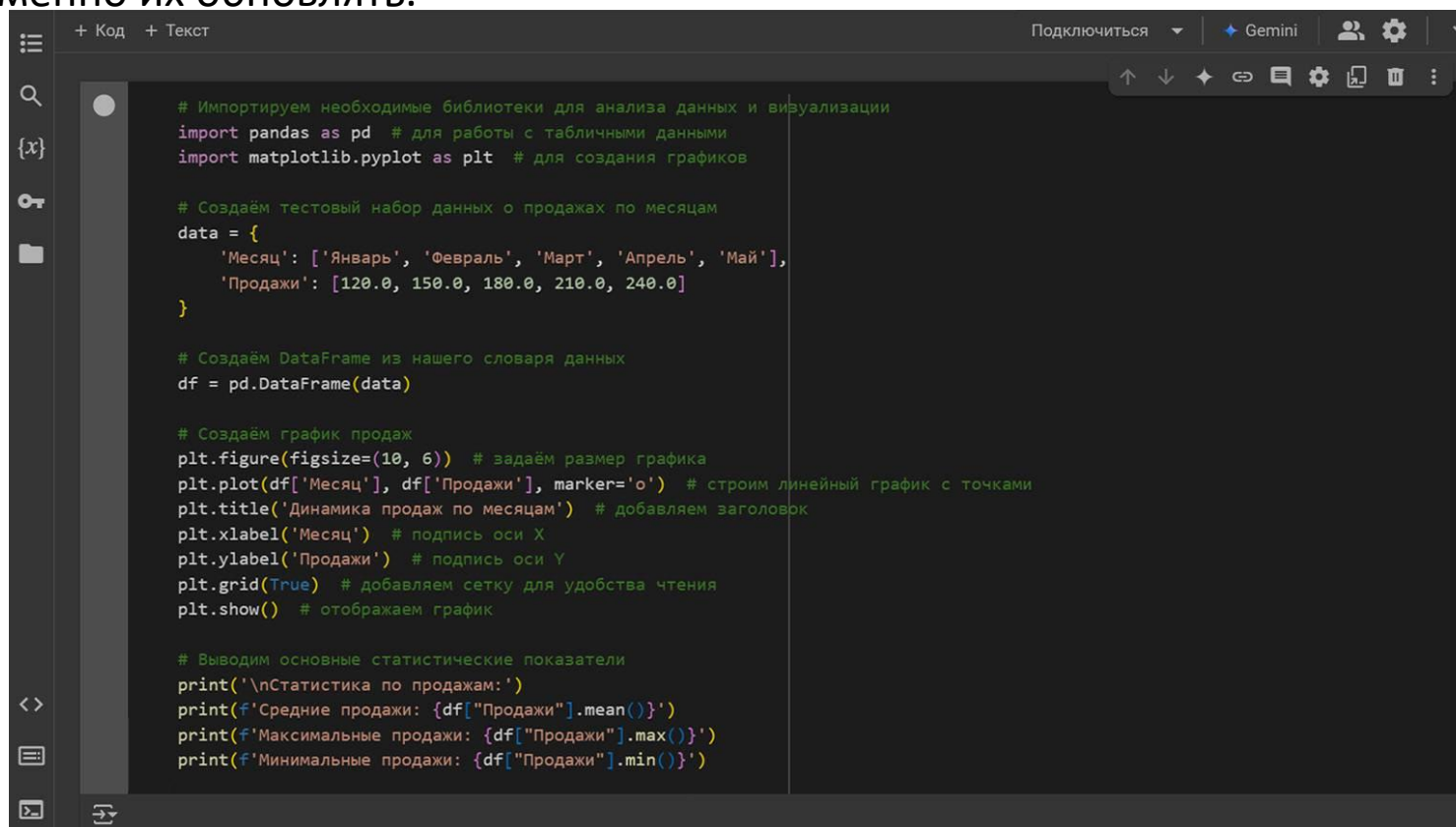
[Google Sheets](#): предоставляет возможность напрямую импортировать и экспортировать данные из электронных таблиц. Например, вы можете загрузить таблицу с данными о ежедневных продажах, обработать их и выгрузить результаты обратно в Google Sheets для создания отчётов.

[GitHub](#): позволяет импортировать код и блокноты напрямую из репозитория. Это даёт возможность использовать готовые модели и библиотеки для анализа данных, не устанавливая их на компьютер.

[Kaggle](#): позволяет работать с готовыми наборами данных для машинного обучения и аналитики. Вы можете загружать датасеты прямо в Colab и использовать существующие решения от сообщества для своих проектов.

[TensorBoard](#): инструмент визуализации для машинного обучения. Интеграция с Colab даёт возможность отслеживать обучение моделей прямо в браузере: смотреть графики, анализировать структуру модели и настраивать параметры — и всё это без установки TensorBoard.

Google Colab работает как веб-приложение и доступен в любом современном браузере с поддержкой HTML5 и JavaScript — Google Chrome, Mozilla Firefox, Microsoft Edge, Safari и других. Для стабильной работы мы рекомендуем использовать последние версии браузеров и своевременно их обновлять.

The image shows a screenshot of a Google Colab notebook interface. The top bar includes a menu icon, tabs for '+ Код' (Code) and '+ Текст' (Text), a 'Подключиться' (Connect) button, and a 'Gemini' logo. The left sidebar contains icons for file management and search. The main area displays a Python script. The script imports 'pandas' and 'matplotlib.pyplot', creates a test dataset for monthly sales, constructs a DataFrame, plots a line graph with markers, and prints out summary statistics like mean, max, and min sales.

```
# Имportsруем необходимые библиотеки для анализа данных и визуализации
import pandas as pd # для работы с табличными данными
import matplotlib.pyplot as plt # для создания графиков

# Создаём тестовый набор данных о продажах по месяцам
data = {
    'Месяц': ['Январь', 'Февраль', 'Март', 'Апрель', 'Май'],
    'Продажи': [120.0, 150.0, 180.0, 210.0, 240.0]
}

# Создаём DataFrame из нашего словаря данных
df = pd.DataFrame(data)

# Создаём график продаж
plt.figure(figsize=(10, 6)) # задаём размер графика
plt.plot(df['Месяц'], df['Продажи'], marker='o') # строим линейный график с точками
plt.title('Динамика продаж по месяцам') # добавляем заголовок
plt.xlabel('Месяц') # подпись оси X
plt.ylabel('Продажи') # подпись оси Y
plt.grid(True) # добавляем сетку для удобства чтения
plt.show() # отображаем график

# Выводим основные статистические показатели
print('\nСтатистика по продажам:')
print(f'Средние продажи: {df["Продажи"].mean()}')
print(f'Максимальные продажи: {df["Продажи"].max()}')
print(f'Минимальные продажи: {df["Продажи"].min()}')
```

В примере мы подключили необходимые библиотеки, создали тестовый набор данных о продажах, построили линейный график и рассчитали основные статистические показатели. Результаты выполнения кода отображаются в блокноте сразу после его запуска

ПЕРЕХОД

Создать блокнот

<https://colab.research.google.com/>

<https://docs.google.com/presentation/u/0/>

презентации

Как работать с Google Colab

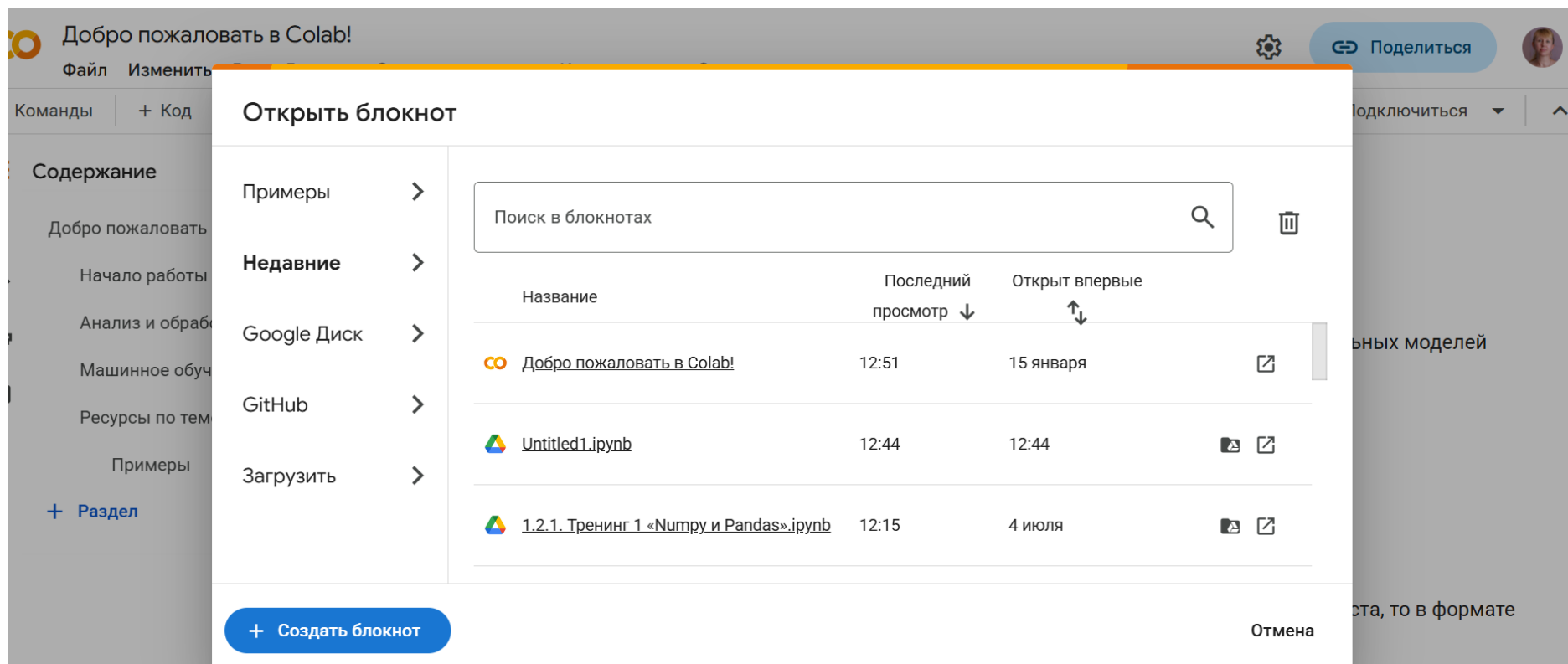
Для начала работы с Google Colab вам понадобится только аккаунт Google. Перейдите на [сайт сервиса](#) и нажмите кнопку Создать блокнот.

Вы также можете воспользоваться Google Drive: нажмите кнопку Создать, выберите раздел Ещё и пункт Google Colaboratory.

В Google Colab блокноты называются «колабами» (colabs) или «ноутбуками» (notebooks).

Файлы сохраняются в формате .ipynb (IPython Notebook).

При создании блокнота вы можете дать ему любое название.



Google Colab построен на системе ячеек — отдельных блоков, которые могут содержать Python-код или текст с разметкой Markdown для пояснений.

Ячейки исполняются по порядку, сохраняя общий контекст выполнения.

Это означает, что все переменные, созданные в одной ячейке, можно использовать в следующих ячейках в течение всей сессии.

A blue rectangular button with the text "Markdown" in white, centered within the button.

Когда вы создадите блокнот, перед вами откроется основной интерфейс сервиса.

В центре находится рабочее пространство, а над ним расположены две кнопки: **[+ Код](#)** и **[+ Текст](#)**.

Нажав на любую из них, вы создадите ячейку соответствующего типа — для кода или для текста.

Для примера возьмём код из статьи «[Как извлечь корень из числа при помощи Python](#)»

и добавим к нему ячейку с текстовым пояснением:

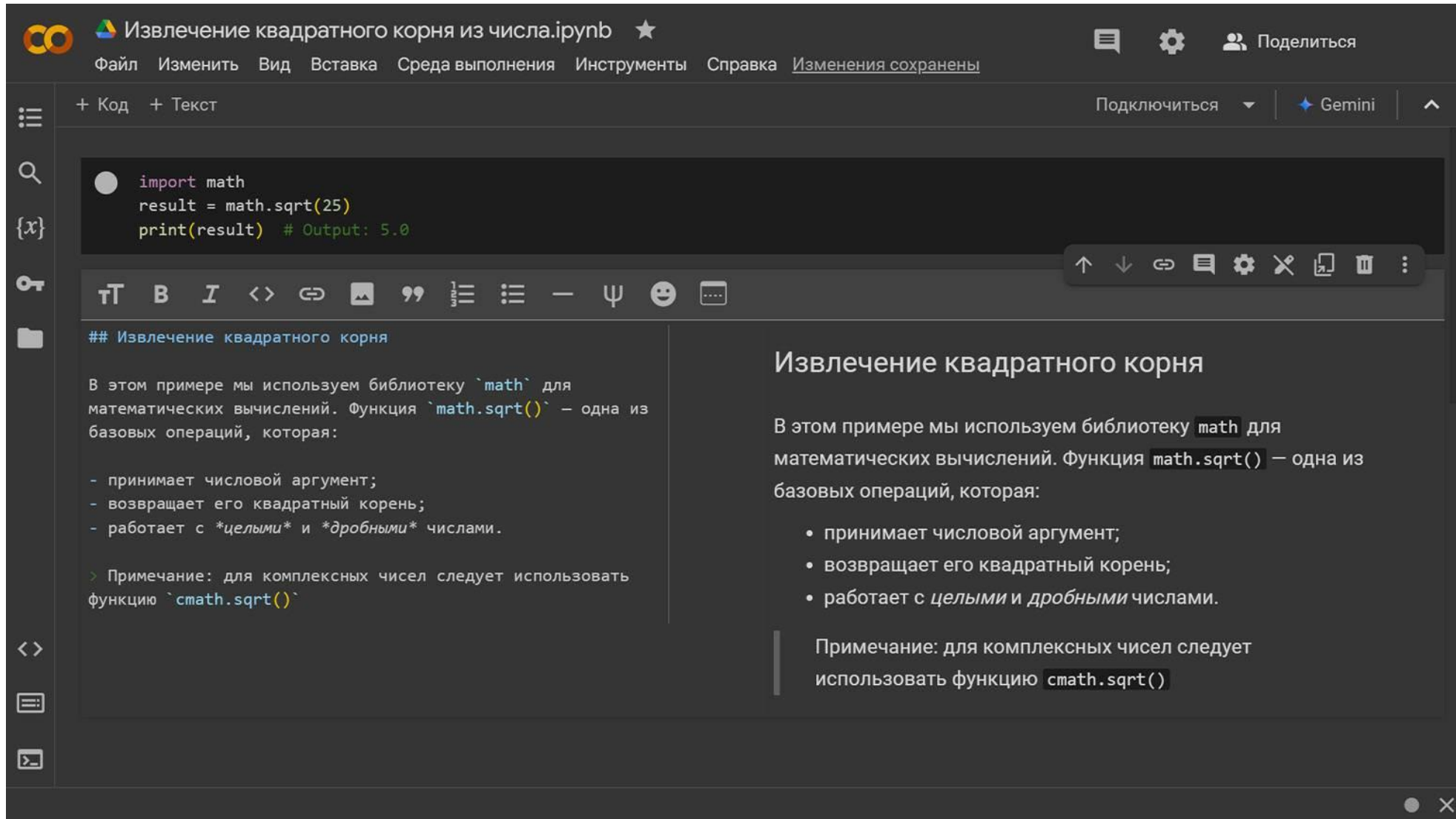
```
import math  
result = math.sqrt(25)  
print(result)  # Output: 5.0
```

Извлечение квадратного корня

В этом примере мы используем библиотеку ``math`` для математических вычислений. Функция

- принимает числовой аргумент;
- возвращает его квадратный корень;
- работает с **целыми** и **дробными** числами.

> Примечание: для комплексных чисел следует использовать функцию ``cmath.sqrt()``



Пример работы в Google Colab в браузере Opera: в верхней ячейке размещён Python-код для вычисления квадратного корня,

в нижней — текстовое описание в формате Markdown.

Для запуска кода нажмите кнопку Play слева от ячейки или используйте сочетание клавиш **CTRL + ENTER**.

Чтобы просмотреть текст без разметки Markdown, дважды щёлкните по ячейке

В Google Colab вы можете перемещать ячейки и выстраивать их в удобном порядке — например, сначала разместить описание, а затем код. Платформа также поддерживает совместную работу: если вы поделитесь ссылкой на блокнот, то можете разрешить другим пользователям просматривать код, комментировать его или вносить изменения.

В Google Colab также встроен ИИ-ассистент Gemini, который интегрирован в интерфейс блокнота и готов помочь с любыми вопросами прямо во время работы.

Например, когда мы спросили ассистента о различных методах извлечения квадратного корня, он не только предложил три альтернативных способа, но и подробно объяснил, какой метод и когда лучше использовать.

The screenshot displays the Google Colab web interface. At the top, the title bar shows the Google Colab logo and the notebook name "Извлечение квадратного корня из числа.ipynb". Below the title bar is a menu bar with options: "Файл", "Изменить", "Вид", "Вставка", "Среда выполнения", "Инструменты", "Справка", and "Изменения сохранены". The main workspace is divided into two panels. The left panel contains a Jupyter notebook cell with Python code:

```
[ ] import math
result = math.sqrt(25)
print(result) # Output: 5.0
```

 Below the code cell is a rich text editor with the title "## Извлечение квадратного корня". The text describes the use of the `math` library and the `math.sqrt()` function. It lists the function's parameters and return values, and includes a note about using `cmath.sqrt()` for complex numbers. The right panel shows the Gemini AI assistant interface. It has a title bar "Gemini" and a search icon. The main text area contains the question "Какие ещё способы существуют?" and the response "Конечно, вот ещё несколько способов извлечения квадратного корня в Python:". The response lists two methods: 1. Using the exponentiation operator `**`, and 2. Using the `pow()` function. Each method is accompanied by a code snippet showing the calculation of the square root of 25. At the bottom of the Gemini panel is a text input field with the placeholder "Введите запрос".

Извлечение квадратного корня из числа.ipynb

Файл Изменить Вид Вставка Среда выполнения Инструменты Справка Изменения сохранены

+ Код + Текст

```
[ ] import math
result = math.sqrt(25)
print(result) # Output: 5.0
```

Извлечение квадратного корня

В этом примере мы используем библиотеку `math` для математических вычислений. Функция `math.sqrt()` — одна из базовых операций, которая:

- принимает числовой аргумент;
- возвращает его квадратный корень;
- работает с *целыми* и *дробными* числами.

> Примечание: для комплексных чисел следует использовать функцию `cmath.sqrt()`

Извлечение квадратного корня

В этом примере мы используем библиотеку `math` для математических вычислений. Функция `math.sqrt()` — одна из базовых операций, которая:

- принимает числовой аргумент;
- возвращает его квадратный корень;
- работает с *целыми* и *дробными* числами.

Примечание: для комплексных чисел следует использовать функцию `cmath.sqrt()`

Gemini

Какие ещё способы существуют?

Конечно, вот ещё несколько способов извлечения квадратного корня в Python:

1. Использование оператора возведения в степень `**`:

```
result = 25 ** 0.5
print(result) # Output: 5.0
```

[Use code with caution](#)

Этот способ основан на математическом свойстве, что извлечение квадратного корня эквивалентно возведению числа в степень 0.5.

2. Использование функции `pow()`:

```
result = pow(25, 0.5)
print(result) # Output: 5.0
```

Введите запрос