Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт перспективной инженерии

Департамент цифровых, робототехнических систем и электроники

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

**дисциплины**

**«Искусственный интеллект и машинное обучение»**

**Вариант 2**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Выполнила:  Алексеева Софья Алексеевна  2 курс, группа ИВТ-б-о-23-1,  09.03.01 «Информатика и вычислительная техника», направленность (профиль) «Программное обеспечение средств вычислительной техники и автоматизированных систем», очная форма обучения  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | |
|  | | Проверил:  доцент департамента цифровых, робототехнических систем и электроники института перспективной инженерии Воронкин Р. А.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | |
|  | |  | |

Отчет защищен с оценкой \_\_\_\_\_\_\_\_\_\_\_ Дата защиты\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ставрополь, 2025 г.

**Тема:** Работа с Jupyter Notebook, JupyterLab и Google Colab.

**Цель:** исследовать базовые возможности интерактивных оболочек Jupyter Notebook, JupyterLab и Google Colab для языка программирования Python.

**Порядок выполнения работы:**

1. Запустили оболочку Jupyter Notebook в браузере.

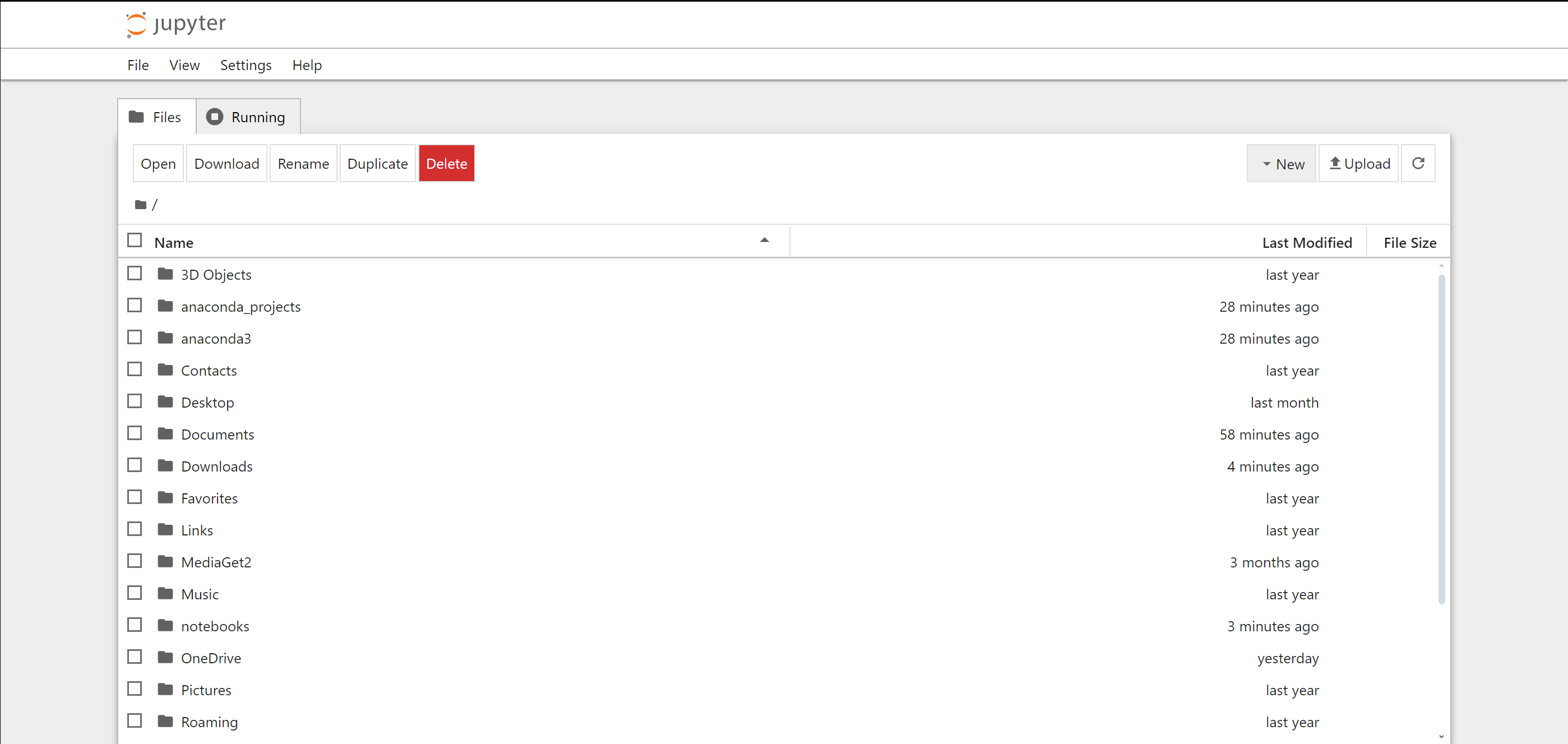


Рисунок 1. Оболочка Jupyter Notebook в браузере

1. Создали первый файл.

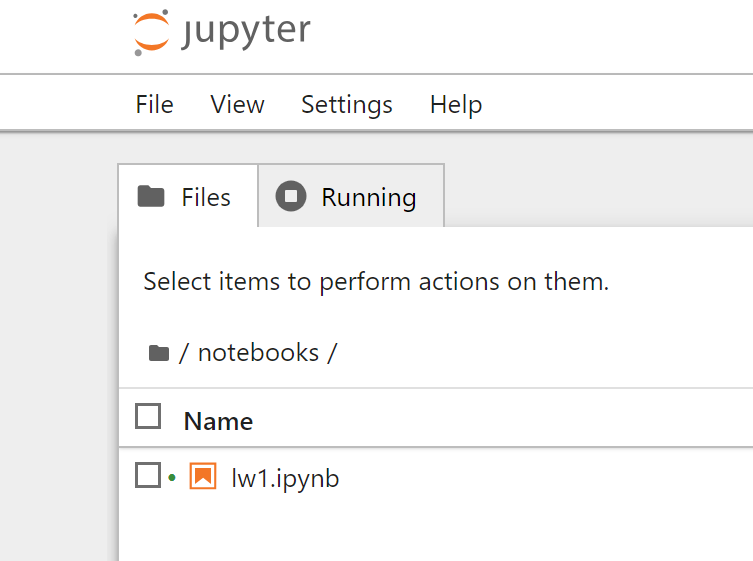


Рисунок 2. Первый файл в Jupyter Notebook

1. Выполнили несколько примеров.

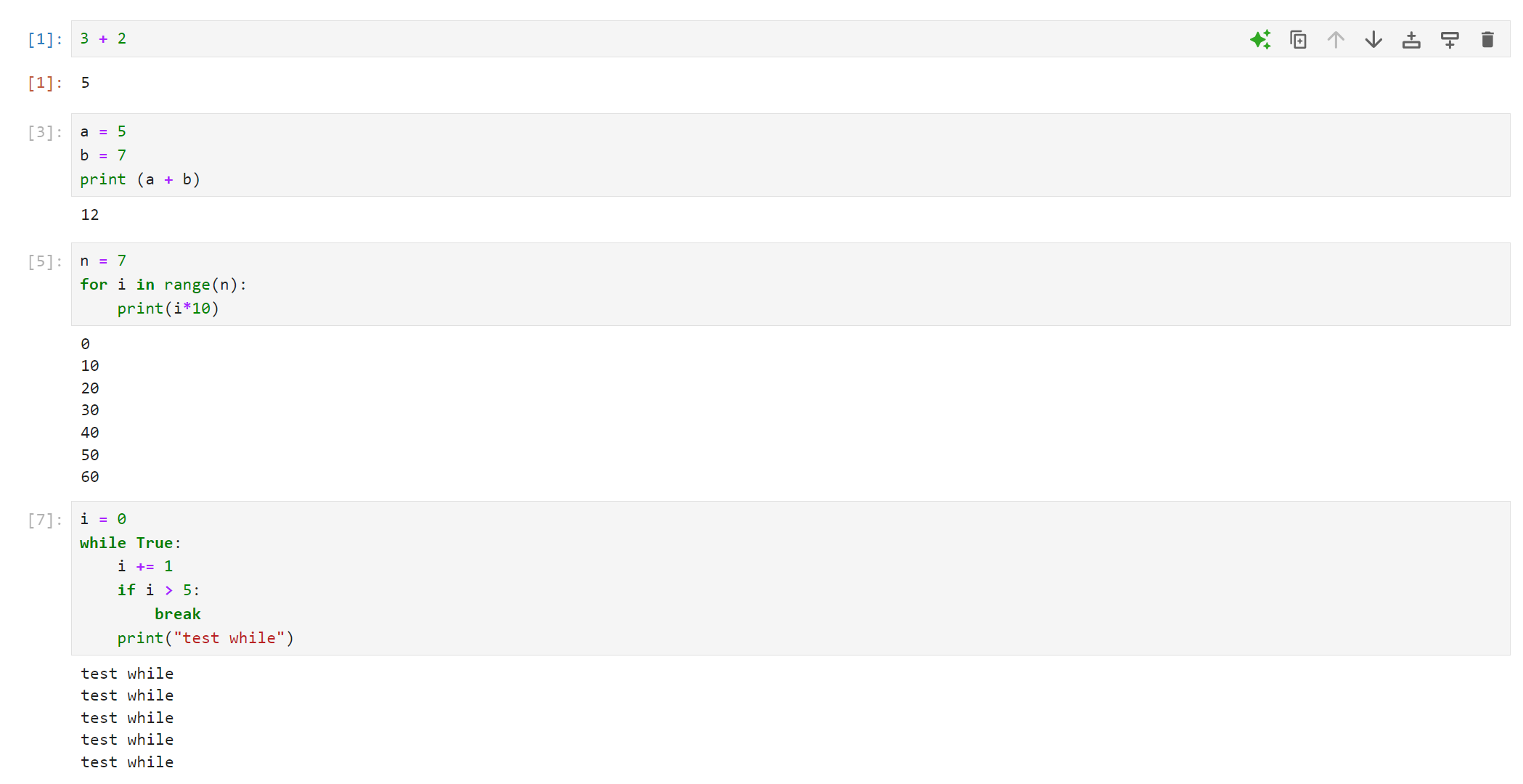


Рисунок 3. Несколько выполненных примеров

1. Вывели изображение на экран.

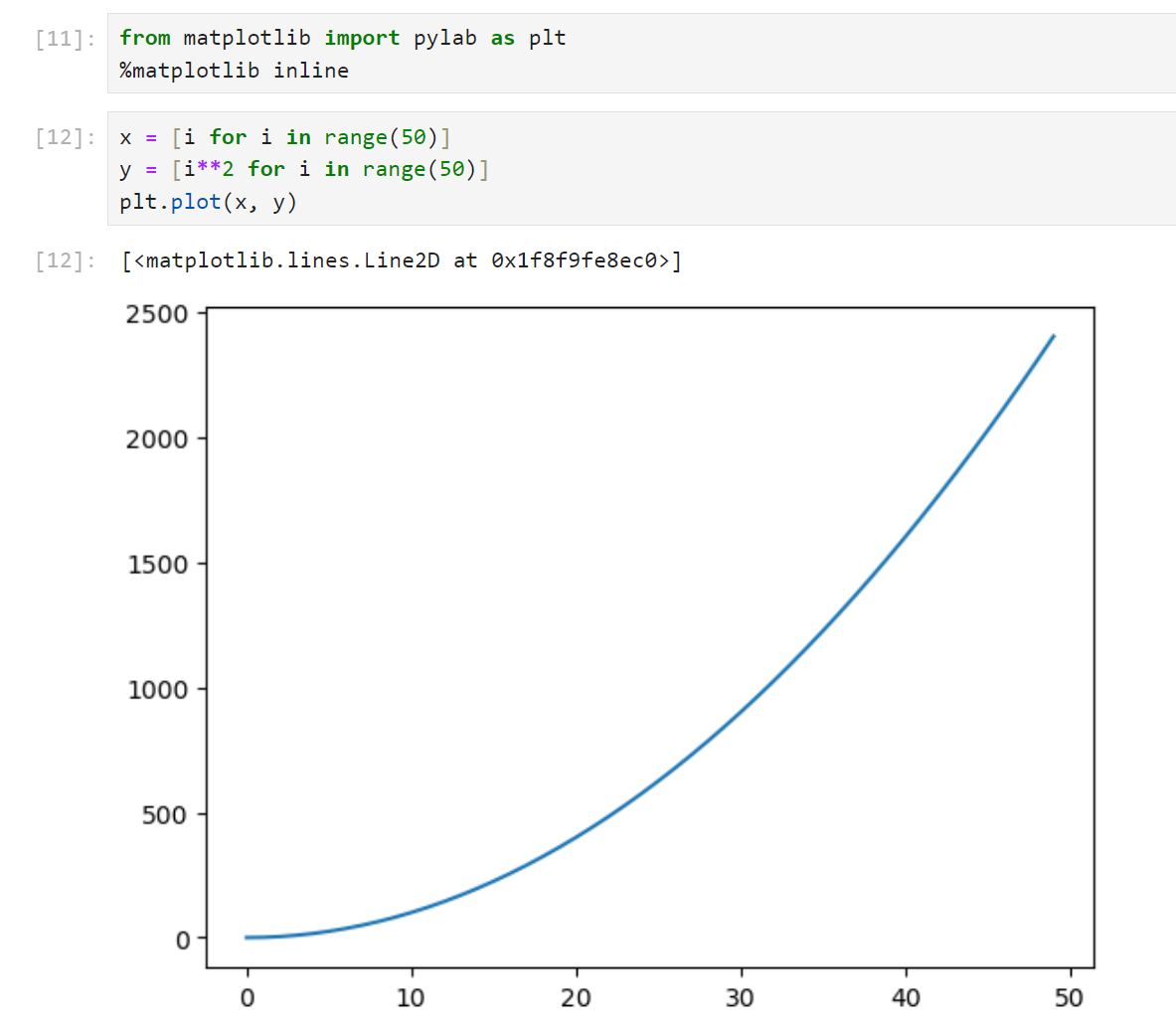


Рисунок 4. Выведение графика в Jupyter Notebook

1. Использовали магические команды. Они были выведены в виде списков.

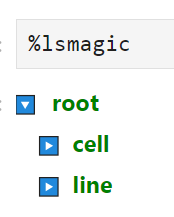


Рисунок 5. Магические команды в Jupyter Notebook

1. Применили команду %env для работы с окружением.

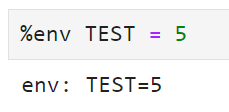


Рисунок 6. Команда %env

1. Примени команду %run для запуска кода из другого файла.

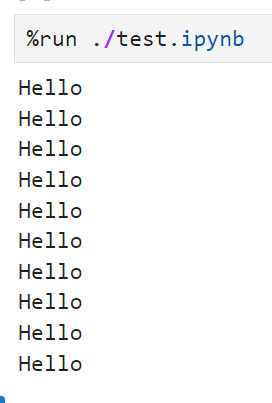


Рисунок 7. Команда %run

1. Использовали команду %%time. Параметр CPU times выводит время работы процессора, исключая периоды ожидания между ними. Параметр Wall time – общее время работы операции.

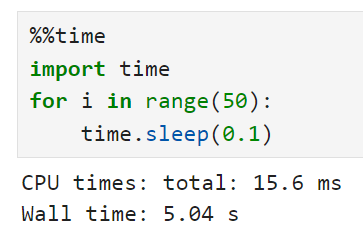


Рисунок 8. Команда %%time

1. Использовали команду %timeit.

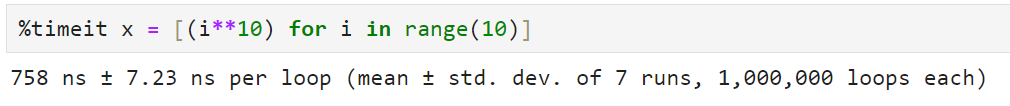


Рисунок 9. Команда %timeit

1. Использовали Markdown для форматирования текста и формул.

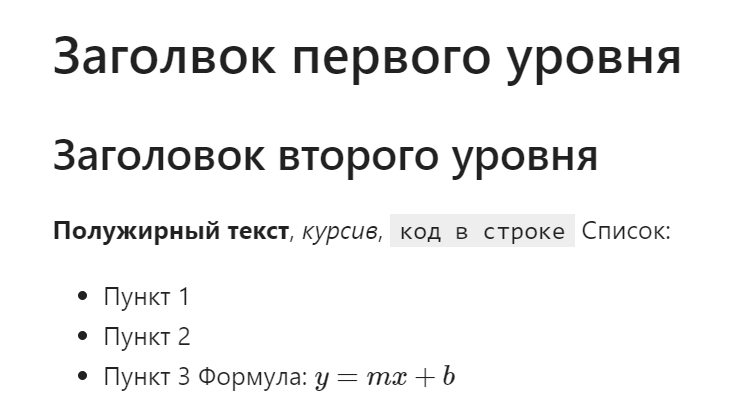


Рисунок 10. Использование Markdown

1. Использовали линейную магическую команду %timeit и блочную магическую команду %%time в JupyterLab.

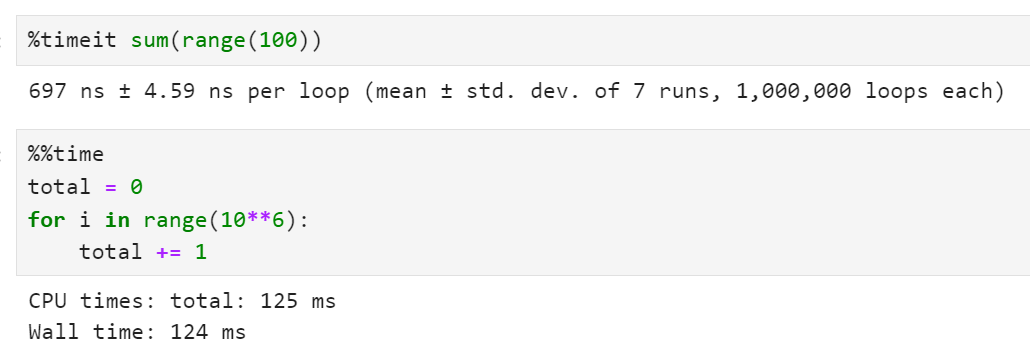


Рисунок 11. Команды %timeit и %%time

1. Построили график с помощью библиотеки matplotlib.

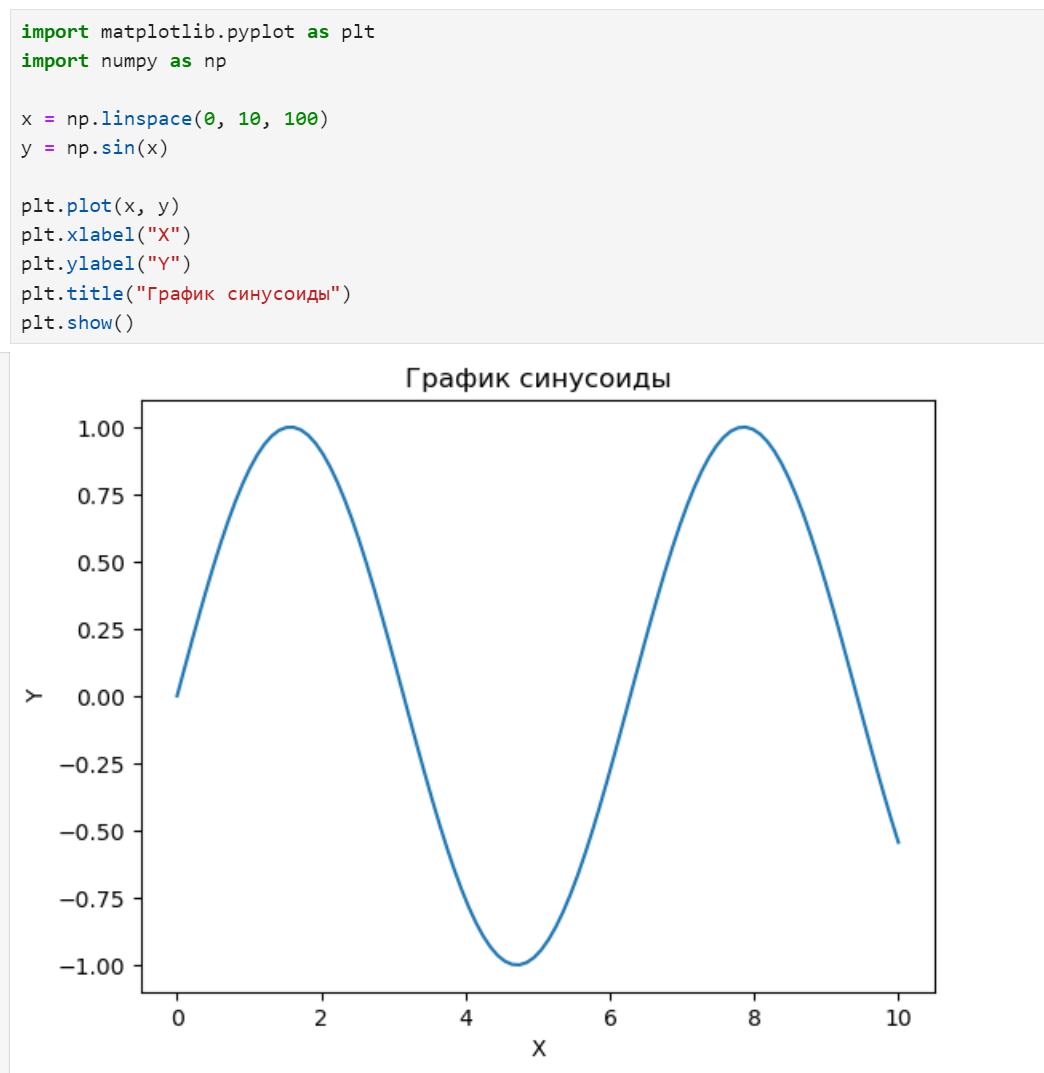


Рисунок 12. Построение графика в JupyterLab

1. Установили с помощью терминала несколько библиотек. Проверили версию Python.

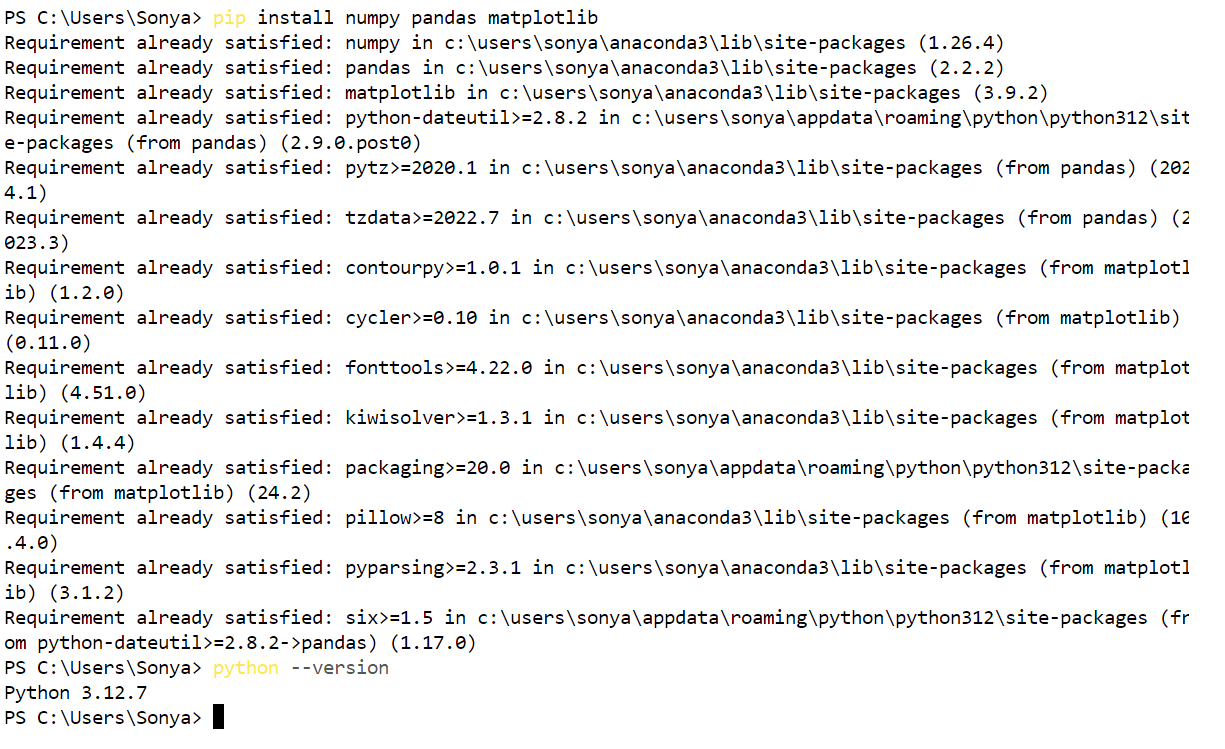


Рисунок 13. Использование терминала

1. Создали новый ноутбук в Google Colab. Использовали тип ячейки Markdown.

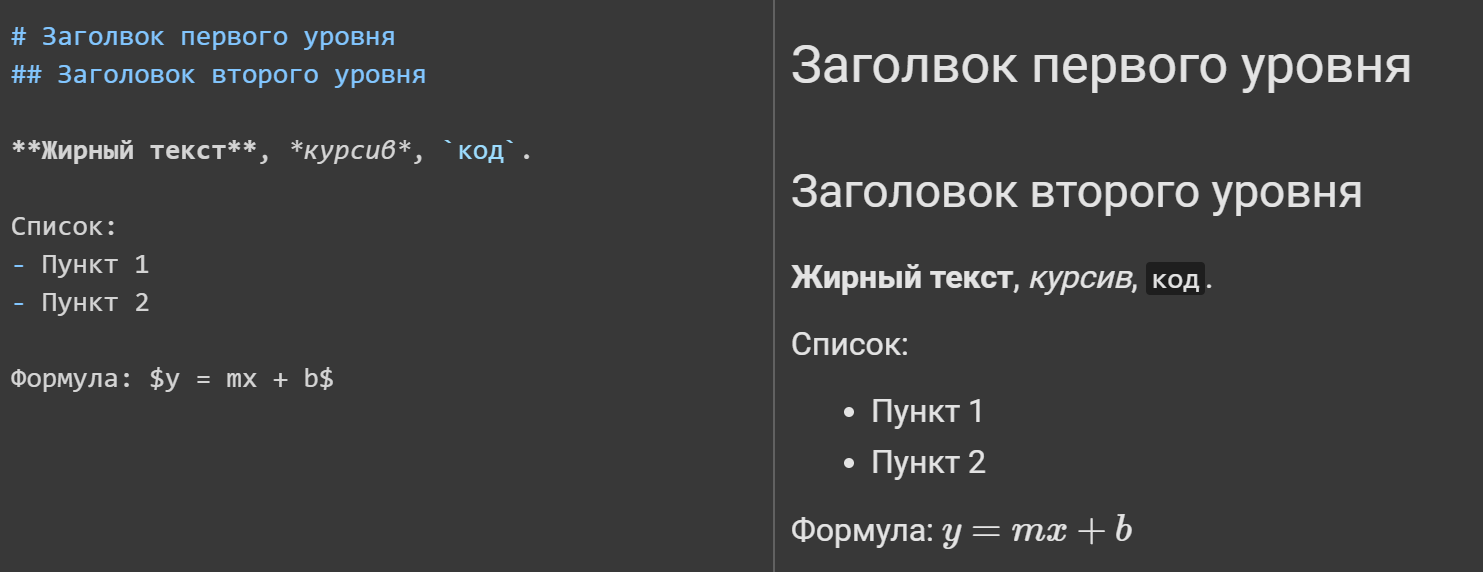


Рисунок 14. Использование типа ячейки Markdown в Google Colab

1. Изменили среду выполнения и проверили наличие GPU.

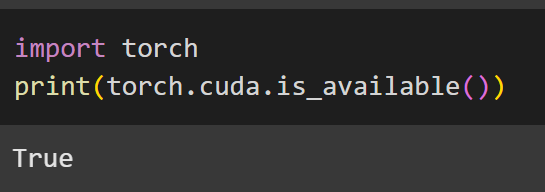


Рисунок 15. Проверка наличия GPU

1. Подключили Google диск. Далее появляется окно с авторизацией и запросом на доступ.

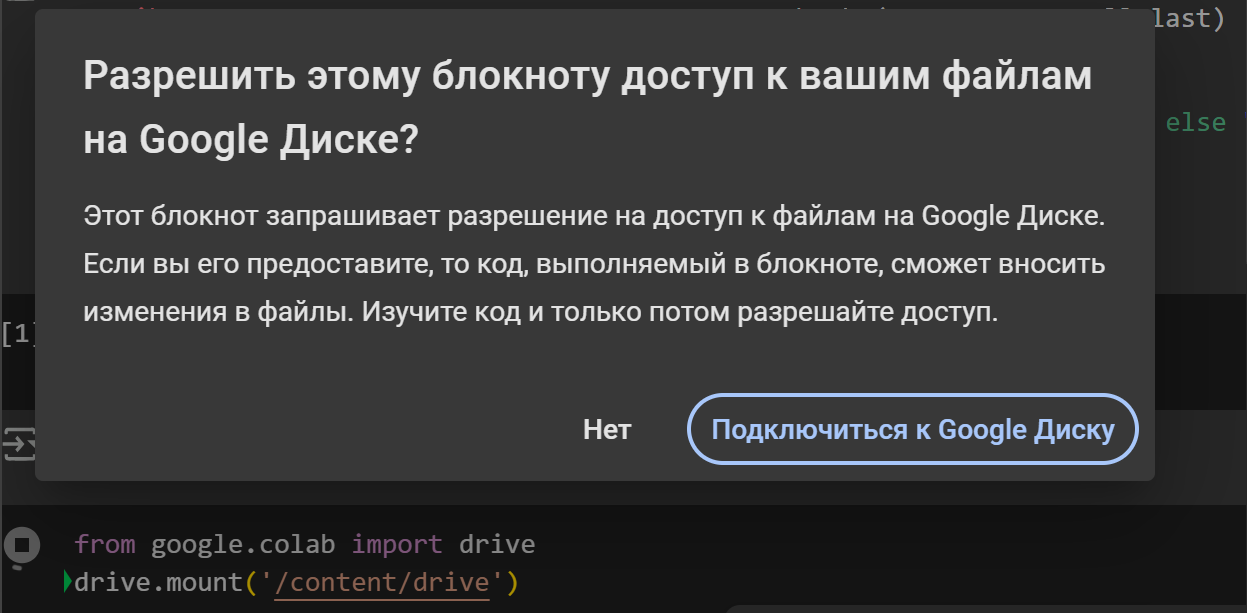


Рисунок 16. Подключение Google диска к Google Colab

1. Выполнили загрузку файлов через код.

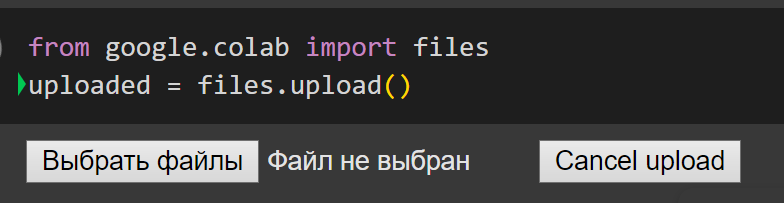


Рисунок 17. Загрузка файлов

1. Выполнили работу с файловой системой: вывели текущую директорию, создали каталог, просмотрели файлы в директории и удалили каталог.

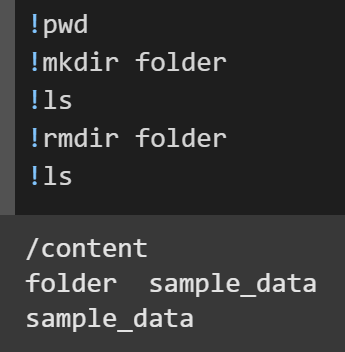


Рисунок 18. Команды для работы с файловой системой

1. Использовали магические команды в Google Colab.

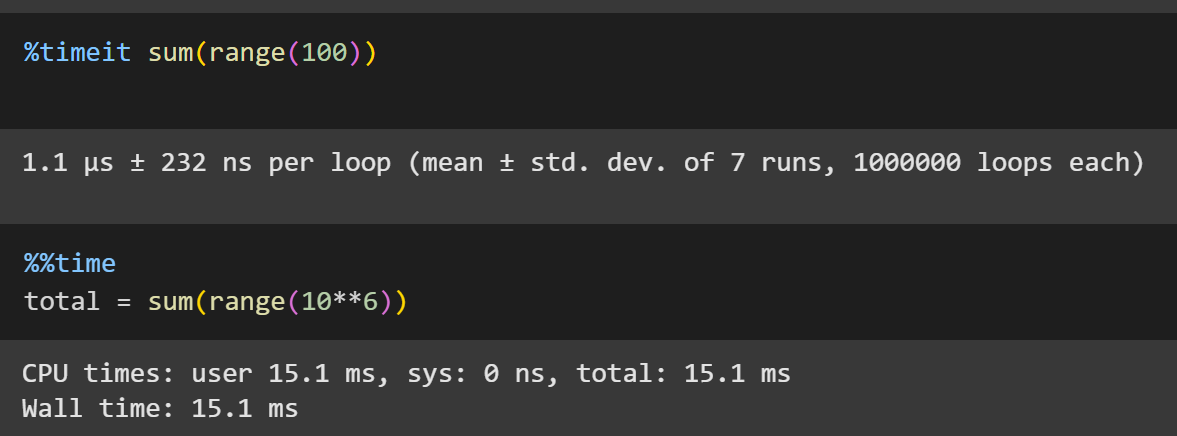


Рисунок 19. Магические команды в Google Colab

1. Построили график в Google Colab.

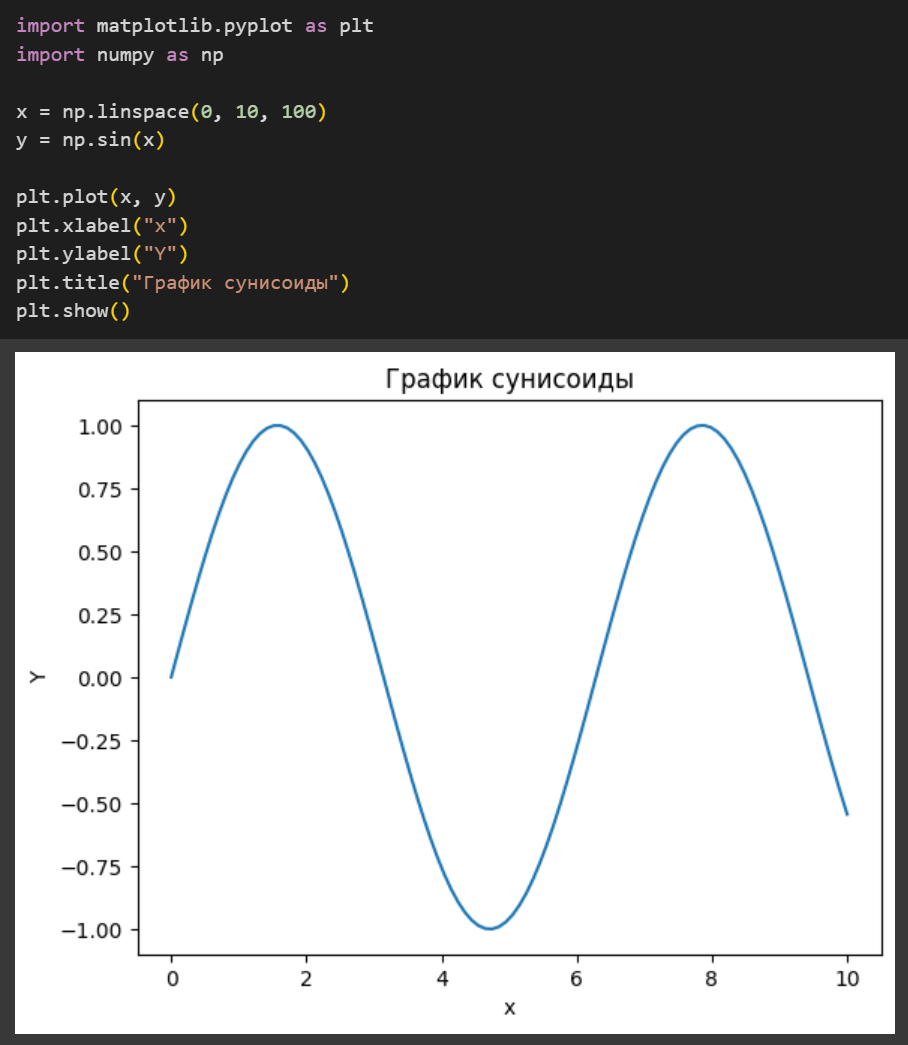


Рисунок 20. Построение графика в Google Colab

1. Установили стандартные библиотеки.

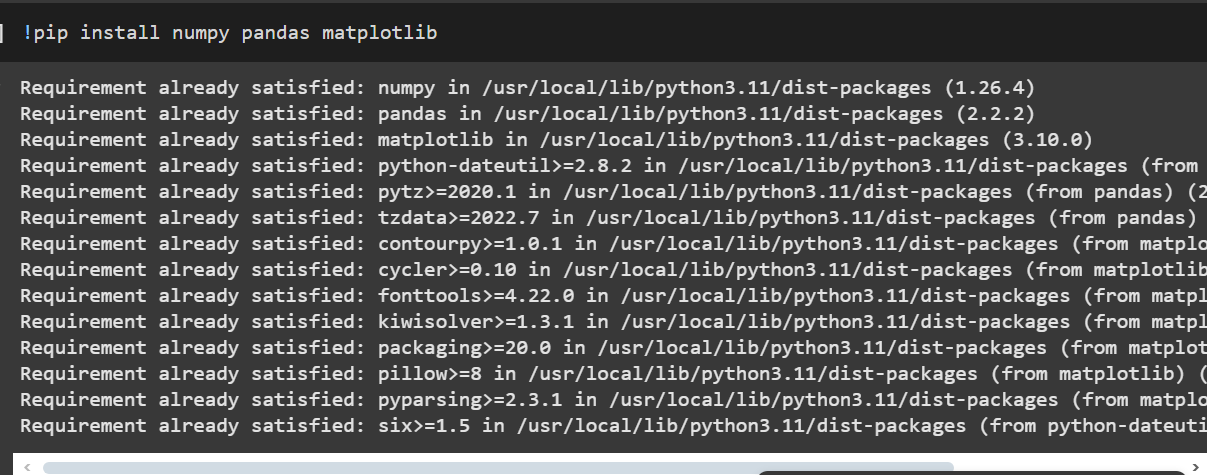


Рисунок 21. Установка стандартных библиотек

1. Создать Markdown и:

* написали заголовок "Практическое задание №1";
* добавили жирный и курсивный текст;
* создали нумерованный и маркированный списки;
* вставили согласно индивидуальному заданию;
* вставили изображение через ![Описание](URL).

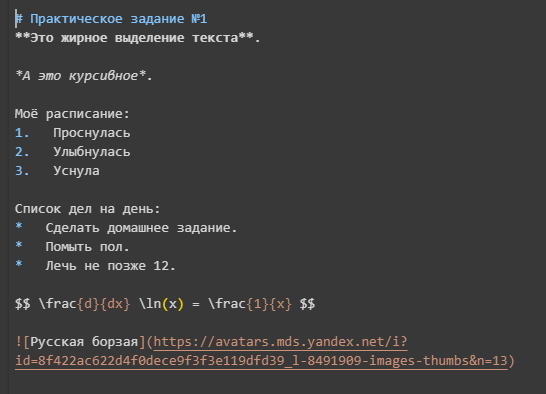


Рисунок 22. Код ячейки с типом Markdown



Рисунок 23. Код программы

1. Создали ячейку Python-кода и:

* запросили у пользователя его имя с помощью input();
* вывели приветствие: «Привет, <имя>! Добро пожаловать в Google Colab!»;
* запустили ячейку (Shift + Enter).

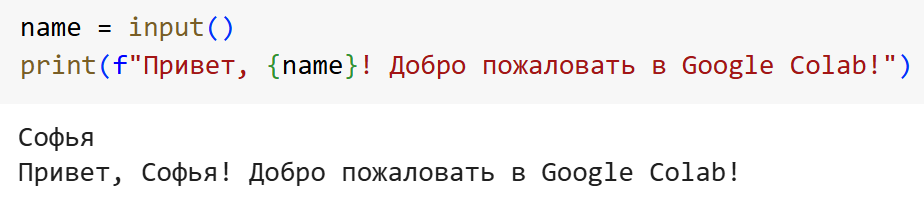


Рисунок 24. Вывод приветствия с помощью input()

1. Выполнили список указаний:

* создали и сохранили текстовый файл с помощью open();
* записали в него несколько строк текста;
* закрыли и затем снова открыли его, считалв содержимое и выведя на экран;
* проверили, существует ли файл, используя os.path.exists();
* удалили файл с помощью модуля os.



Рисунок 25. Выполнение работ с файлами

1. Вывели список всех доступных магических команд.

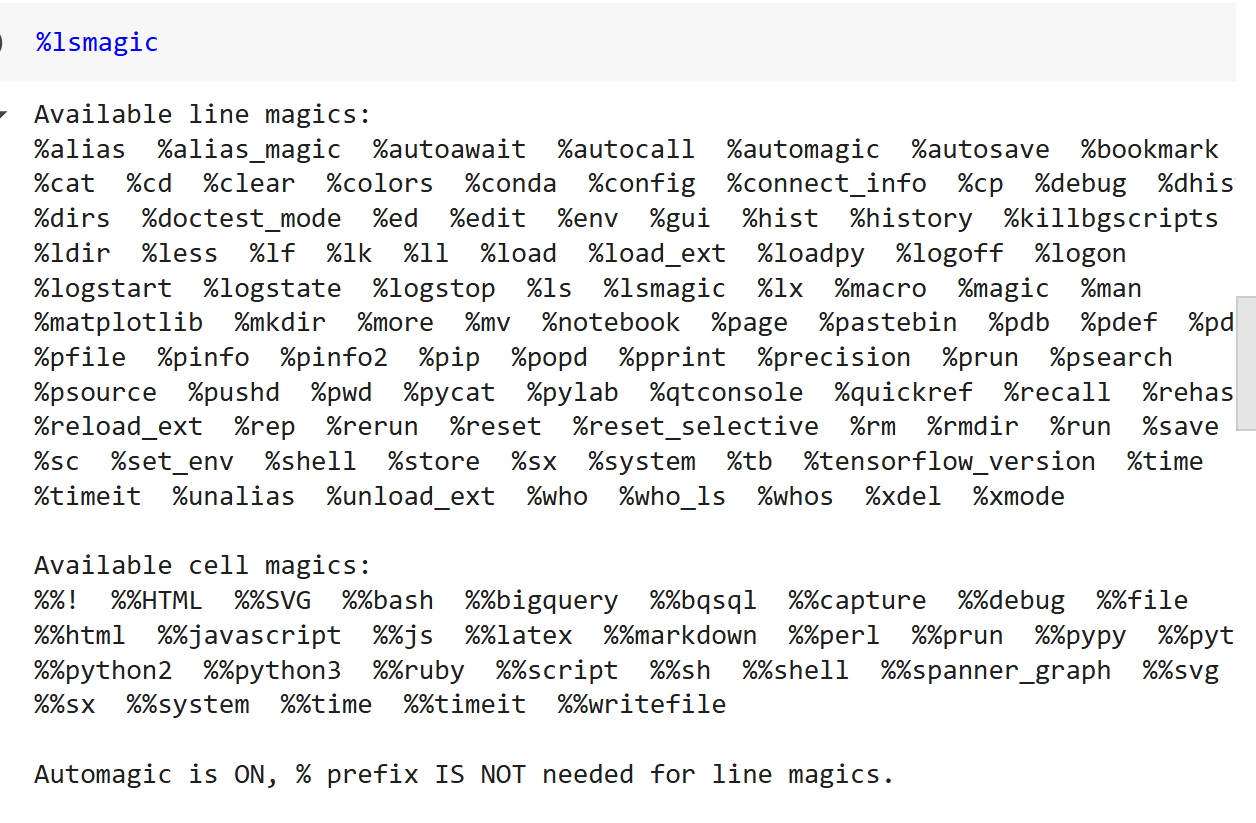


Рисунок 26. Список магических команд в Google Colab

1. Выполнили список указаний:

* использовали %time для измерения врмени выполнения кода;
* создали python-скрипт (%%writefile script.py) и выполнили его через !python script.py;
* вывели список файлов в текущей директории с помощью %ls;
* использовали %history для просмотра истории команд.



Рисунок 27. Выполненные указания в Google Colab

1. Выполнили список указаний по командам в терминале:

* вывели список файлов в текущей директории с помощью !ls;
* проверили, какой Python используется;
* создали папку test и убедитесь, что она появилась;
* удалили папку;
* очистили вывод в ячейке.

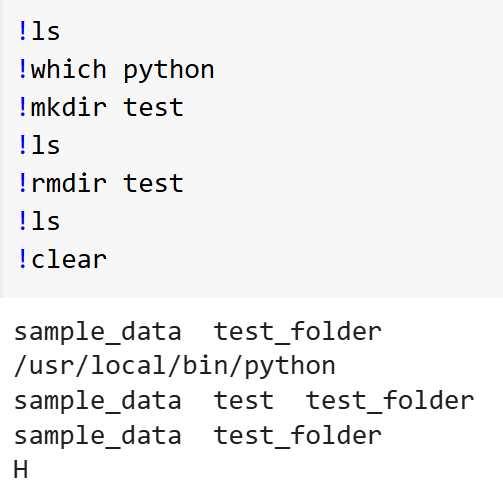


Рисунок 28. Выполнили команды терминала в Google Colab

1. Выполнили список указаний по командам в терминале:

* подключили Google Drive;
* создали и сохранили текстовый файл в Google Drive;
* прочитали файл из Google Drive;
* создали и сохранили csv-файл вручную.



Рисунок 29. Работа с Google Drive в Google Colab

**Ответы на контрольные вопросы:**

1. Какие основные отличия JupyterLab от Jupyter Notebook?

|  |  |  |
| --- | --- | --- |
| Функции | Jupyter Notebook | JupyterLab |
| Интерфейс и организации работы | Отдельные веб-страницы с линейной последовательностью ячеек. Окружение ограничено одной тетрадью. | Интегрированная среда (IDE) с вкладками, панелями, файловыми менеджерами, терминалом и редактором кода. Редактирование нескольких файлов одновременно. |
| Работа с файлами | .ipynb. | .ipynb, .py, .csv, .md, .json, .yaml, .txt. |
| Многозадачность | Требуется несколько вкладов браузера для работы с разными тетрадями. | Система вкладок и разделения экрана позволяет работать с несколькими файлами в одном окне. |
| Поддержка расширений | Поддерживает, сложная настройка. | Встроенный менеджер расширений, удобное добавление новых функций. |
| Гибкость интерфейса | Фиксированный интерфейс. | Гибкий интерфейс: настройка панелей, вида и работа в нескольких окнах. |
| Поддержка терминала | Требуется ручное подключение и настройка. | Встроенный терминал. |
| Производительность | Высокая. | Потребляет много ресурсов при нескольких вкладках. |

Таблица 1. Основные отличия Jupyter Notebook и JupyterLab

1. Как создать новую рабочую среду (ноутбук) в JupyterLab?

Необходимо в меню выбрать File, затем в выпадающим меню выбрать New и Notebook. Новая тетрадь сразу же откроется.

1. Какие типы ячеек поддерживаются в JupyterLab и как их переключать?

Код (Code) – для написания и выполнения программного кода.

Текст (Markdown) – используется для оформления пояснений, форматированного текста и математических формул на основе LaTeX.

Вывод (Raw) – предназначен для хранения необработанного текста, например, для экспорта в другие форматы.

1. Как выполнить код в ячейке и какие горячие клавиши для этого используются?

Запустить код в ячейке можно с помощью кнопки перевернутого треугольника. Выполнение кода с помощью горячих клавиш запускается сочетанием Shift + Enter.

1. Как запустить терминал или текстовый редактор внутри JupyterLab?

Терминал запускается через меню File, затем пункты New и Terminal.

Текстовой редактор запускается при изменении типа ячейки на Markdown.

1. Какие инструменты JupyterLab позволяют работать с файлами и структурами каталогов?

JupyterLab предоставляет удобную систему взаимодействия с файлами благодаря файловому браузеру в левой боковой панели.

Сама среда позволяет открывать и редактировать файлы разных типов, включая. ipynb, .py, .txt, .json, .csv, и другие.

1. Как можно управлять ядрами (kernels) в JupyterLab?

Управление ядрами осуществляется несколькими способами. Первый способ – выбрать в верхнем меню кнопку Kermel.

Второй способ – в левом боковом меню выбрать круглую иконку.

1. Каковы основные возможности системы вкладок и окон в интерфейсе JupyterLab?

JupyterLab имеет гибкий интерфейс, позволяющий пользователю настраивать его под себя.

Программа имеет многооконный режим. Он позволяет работать сразу с несколькими файлами. Так же рабочее пространство можно разделить на несколько панелей.

В левом боковом меню расположена удобная система работы с вкладками, позволяющая взаимодействовать с файловой системой.

В правом боковом меню находятся несколько функций: сохранение и использование макетов кода, хранение переменных, использованных ресурсов, точки остановы и встроенный искусственный интеллект.

1. Какие магические команды можно использовать в JupyterLab для измерения времени выполнения кода? Приведите примеры.

Для измерения времени выполнения команды используется %timeit.

Для измерения времени выполнения всей ячейки используется %%time.

1. Какие магические команды позволяют запускать код на других языках программирования в JupyterLab?

Некоторые языки программирования можно запустить с помощью магических команд. Например, код Python можно запустить с помощью команды %run. Прочие языки запускать с помощью команд %%bash, %sql, %%js и т.д.

1. Какие основные отличия Google Colab от JupyterLab?

|  |  |  |
| --- | --- | --- |
| Функции | Google Colab | JupyterLab |
| Интерфейс и организации работы | Облачный сервис, работающий в браузере.  Интерфейс с вкладками, но ограниченный по настройке. | Интегрированная среда (IDE) с вкладками, панелями, файловыми менеджерами, терминалом и редактором кода. Редактирование нескольких файлов одновременно. |
| Работа с файлами | Виртуальная файловая система, доступ через Google Drive. | .ipynb, .py, .csv, .md, .json, .yaml, .txt. |
| Многозадачность | Открытие нескольких вкладок браузера возможно, но ограничено сессиями. | Система вкладок и разделения экрана позволяет работать с несколькими файлами в одном окне. |
| Поддержка расширений | Ограничена, нельзя устанавливать пользовательские расширения. | Встроенный менеджер расширений, удобное добавление новых функций. |
| Гибкость интерфейса | Фиксированный интерфейс. | Гибкий интерфейс: настройка панелей, вида и работа в нескольких окнах. |
| Поддержка терминала | Доступ к терминалу через !. | Встроенный терминал. |
| Производительность | Требует подключения к серверу. | Потребляет много ресурсов при нескольких вкладках. |

Таблица 2. Основные отличия Google Colab и JupyterLab

1. Как создать новый ноутбук в Google Colab?

В верхнем меню необходимо нажать Файл, а затем «Создать новый блокнот».

1. Какие типы ячеек доступны в Google Colab, и как их переключать?

В Google Colab доступно 2 вида ячеек: код и текст.

Переключаться между ними можно несколькими способами. Первый – горячие клавиши Ctrl + M M для смены на Markdown и Ctrl + M Y для смены на код.

Второй способ – нажать на кнопки +Код и + Текст под верхним меню.

Третий способ – при наведении мышью под текущую ячейку появится выбор типа новой ячейки.

1. Как выполнить код в ячейке Google Colab и какие горячие клавиши для этого используются?

Выполнить код в ячейке можно двумя способами.

Первый способ – навести на область слева от ячейки.

Второй способ – нажать комбинацию горячих клавиш Shift + Enter.

1. Какие способы загрузки и сохранения файлов поддерживает Google Colab?

Загрузка файлов с компьютера осуществляется с помощью следующих команд:

from google.colab import files

uploaded = files.upload().

Так же файлы можно загрузить из хранилища Google Drive:

from google.colab import drive

drive.mount('/content/drive').

1. Как можно подключить Google Drive к Google Colab и работать с файлами?

Подключение Google Drive происходит следующим образом:

from google.colab import drive

drive.mount('/content/drive').

1. Какие команды используются для загрузки файлов в Google Colab из локального компьютера?

from google.colab import files

uploaded = files.upload().

1. Как посмотреть список файлов, хранящихся в среде Google Colab?

Необходимо использовать команду !ls.

1. Какие магические команды можно использовать в Google Colab для измерения времени выполнения кода? Приведите примеры.

Для измерения времени выполнения команды используется %timeit.

Для измерения времени выполнения всей ячейки используется %%time.

1. Как можно изменить аппаратные ресурсы в Google Colab (например, переключиться на GPU)?

Для этого необходимо в верхнем меню выбрать пункт Среда выполнения. В выпадающем меню выбрать «сменить среду выполнения». В открывшимся окне выбрать необходимую среду.

**Вывод:** в ходе лабораторной работы были исследованы базовые возможности интерактивных оболочек Jupyter Notebook, JupyterLab и Google Colab для языка программирования Python, выполнены практические задания.