**Министерство науки и высшего образования Российской Федерации** федеральное государственное автономное образовательное учреждение высшего образования

# «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

**Отчет**

по лабораторной работе №2

по дисциплине «**Инструментальные средства разработки ПО**»

## Автор: Пискунов И.Р. Факультет: ИТИП Группа: M33071

Преподаватель: Приискалов Р.А.



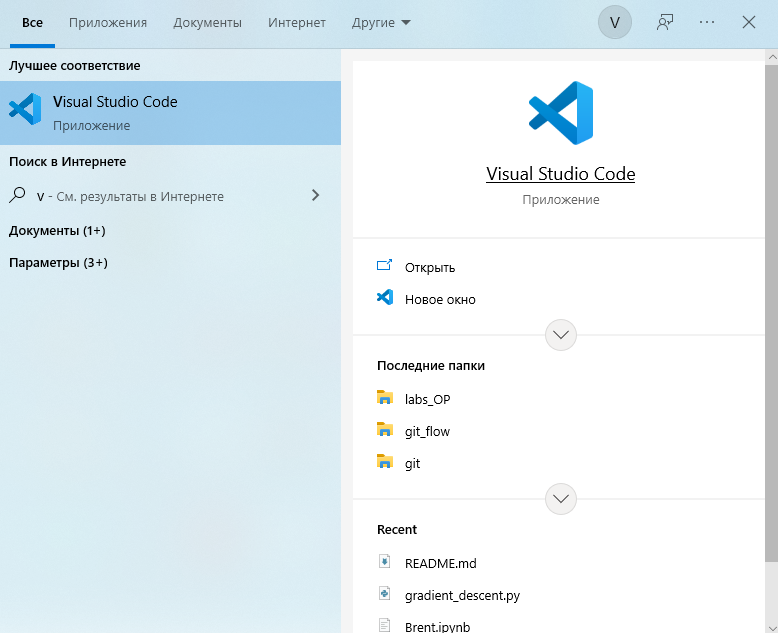
## Санкт-Петербург 2022

# Цели работы:

1. Скачать и установить Visual Studio Code (можно другую IDE, главное произвести настройку всех нижеописанных пунктов)
2. Сконфигурировать для запуска и отладки кода (любой из 3 языков java, python, c++).
3. Написать простой проект (чтение файла, простой консольный калькулятор)
4. Написать тесты к проекту.
5. Изучить Visual Studio Code CLI: описать не менее 10 команд, используемых в данном интерфейсе.
6. Написать 5 собственных сниппетов
7. Изучить и описать наиболее распространённые HotKeys
8. Задокументировать код с помощью расширения https://marketplace.visualstudio.com/items?itemName=cschlosser.doxdocgen (или аналоги)
9. Установить шрифт FiraCode (или другой)
10. Установить https://marketplace.visualstudio.com/items?ite mName=MS-vsliveshare.vsliveshare и показать, чего получилось) (любое другое расширение или встроенные возможности ide для Беринга вашего кода и совместной работы)

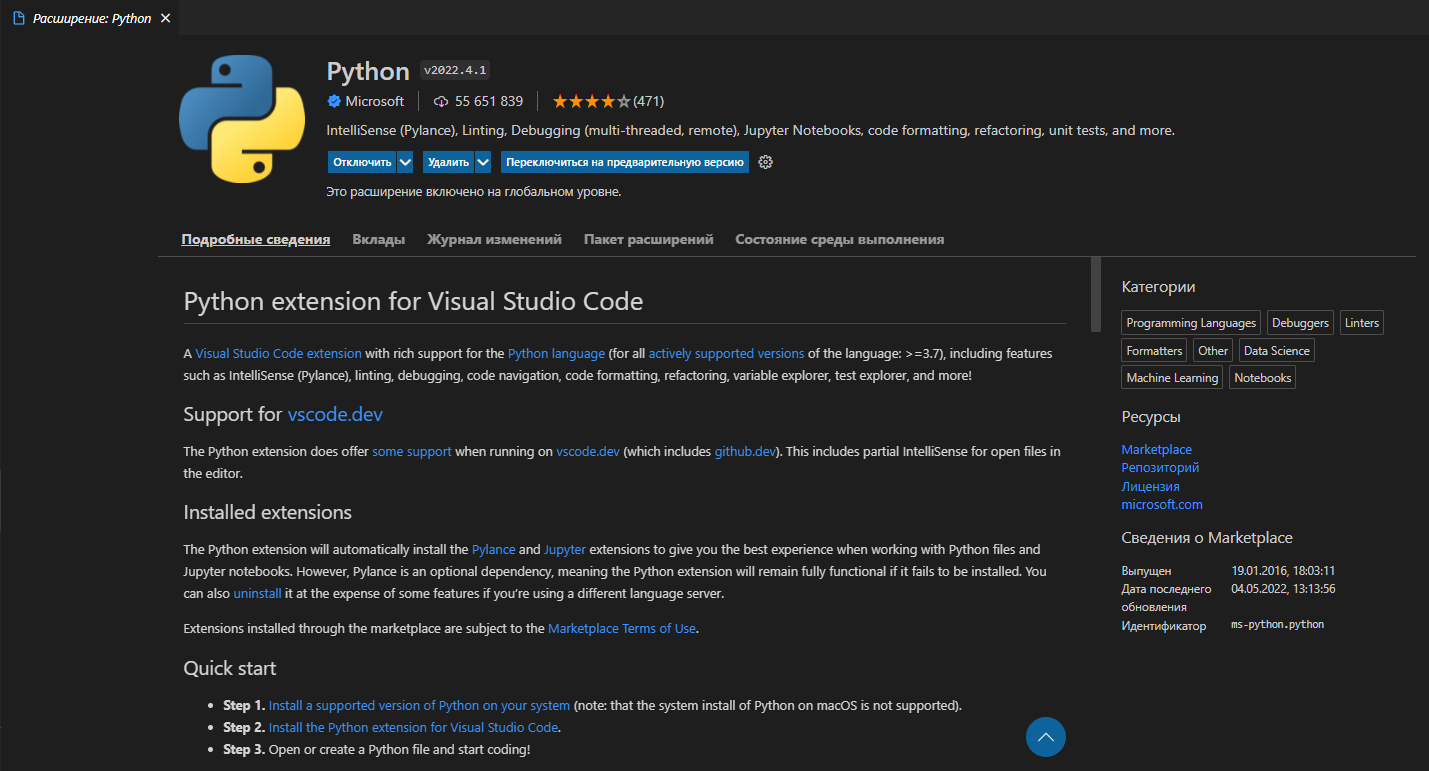
**1. Скачать и установить Visual Studio Code (можно другую IDE, главное произвести настройку всех нижеописанных пунктов)**

Visual Studio Code — это бесплатный редактор исходных кодов от Microsoft для Windows, Linux и MacOS. Возможности включают поддержку отладки, подсветку синтаксиса, интеллектуальное завершение кода, фрагменты, рефакторинг кода и встроенный Git.

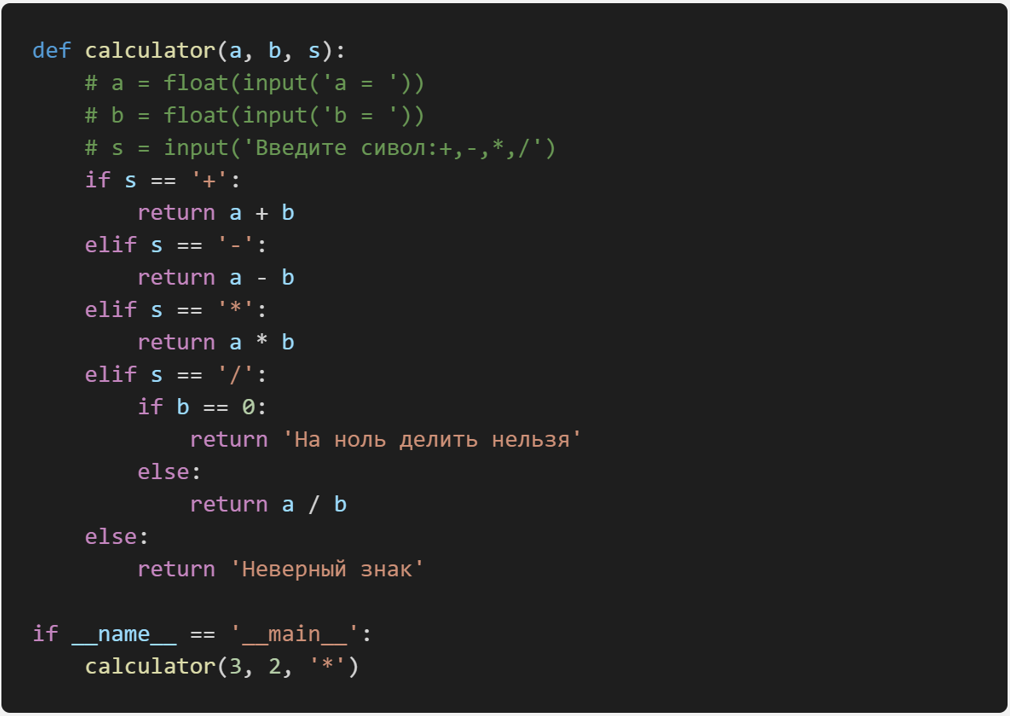
****

**2. Сконфигурировать для запуска и отладки кода (любой из 3 языков java, python, c++).**

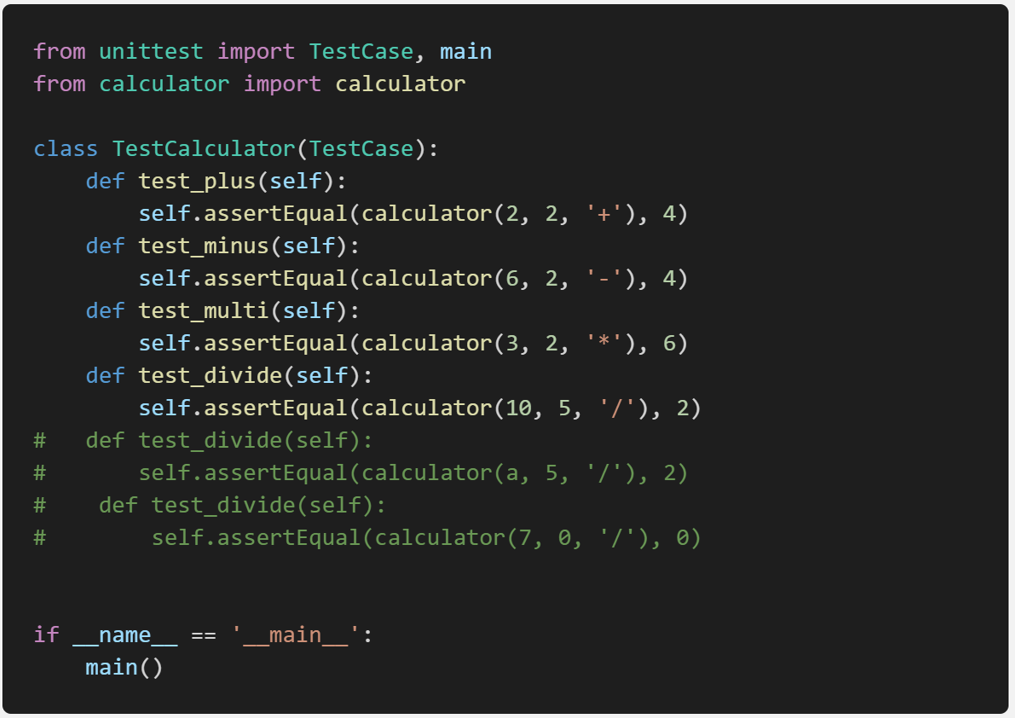
Был выбран python



**3. Калькулятор**



**4. Тесты к проекту**

****

**Ссылка на GitHub:** https://github.com/F1ame21/lab2\_PO/tree/master

**5. Visual Studio Code CLI**

|  |  |
| --- | --- |
| code -h --help | Вызов справки |
| code -d -diff <file> <file> | Сравнить два файла |
| code -a -add <folder> | Добавляет папку (файл) в рабочую область, так же можно создать файл |
| code -v --version | Версию VS code (например, 1.66.2), GitHub commit ID, архитектуру системы (например, x64) |
| code -n –-new-window | Открывает новое окно |
| code --list-extensions | Список установленных расширений |
| code –-locale <locale> | Запустить на другом языке (например en-US или ru-RU) |
| code –-install-extension <id\_ext> | Установить расширение |
| code –-uninstall-extension <id\_ext> | Удалить расширение |
| code -s -status | Выводит информацию о процессе, а также о системе |

**6. Написать 5 собственных сниппетов**

Создал глобальный файл, далее в нем по шаблону написал сниппеты.

* **scope**, это поддерживаемые языки для этого сниппета.У нас это python. Каждый поддерживаемый язык отделяется запятой. Это означает, что, если мы находимся внутри .py файла, то можно будет использовать этот сниппет.
* **prefix**, это то, что вам нужно ввести в окне кода, чтобы появился сниппет.
* **body**, это непосредственно ваш фрагмент кода.
* **description**, это поле, в котором нужно дать краткое описание того, какой компонент за что ответственен.
* **$1, $2** это просто отметки, где заканчивается курсор, когда вы нажимаете кнопку табуляции

****

**7. Изучить и описать наиболее распространённые HotKeys**

|  |  |
| --- | --- |
| Ctrl + Shift + P | Открывает палитру команд |
| Ctrl + P | Найти – открыть фал |
| Ctrl + L | Выбрать текущую линию |
| Ctrl + \,  Ctrl+1, Ctrl+2, Ctrl+3 | Открыть новую вкладку и переключение между ними |
| Ctrl + Shift + ` | Создать терминал |
| Ctrl + Alt + Up  Ctrl + Alt + Down | Выбор строк для редактирования (вверх или вниз) |
| Alt + UP  Alt + Down | Переместить линию вверх или вниз |
| Shift + Alt + Left  Shift + Alt + Right | Расширение или сокращение выбора |
| Shift + Alt + Up  Shift + Alt + Down | Скопирует линию (выбранную область) вверх или вниз |
| F2 | Переименовать переменную |

**8. Задокументировать код**

В качестве генератора для документации был выбран Python Sphinx.

При выполнении задания мною были использованы консольные команды.

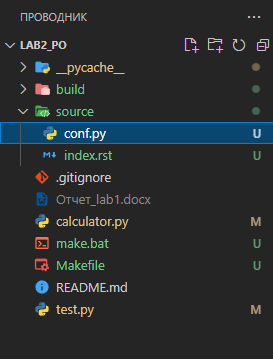
Установка:

pip install sphinx

Для инициализации проекта необходимо выполнить команду:

sphinx quickstart

После ответа на вопросы будут созданы файлы index.rst, conf.py, Makefile, build, static, templates.



**Makefile** — содержит инструкции для генерации результирующего документа командой make.

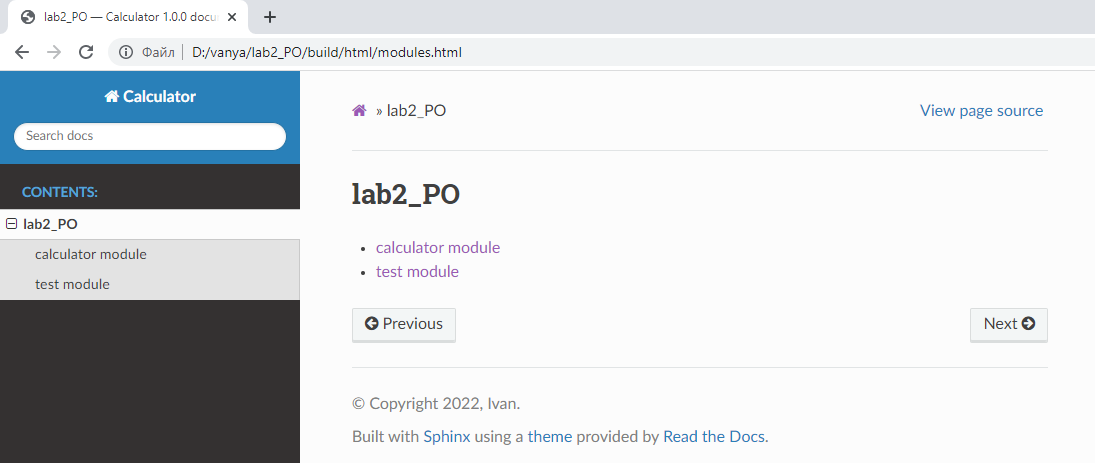
**build** — директория, в которую будут помещены файлы в определенном формате после того, как будет запущен процесс их генерации.

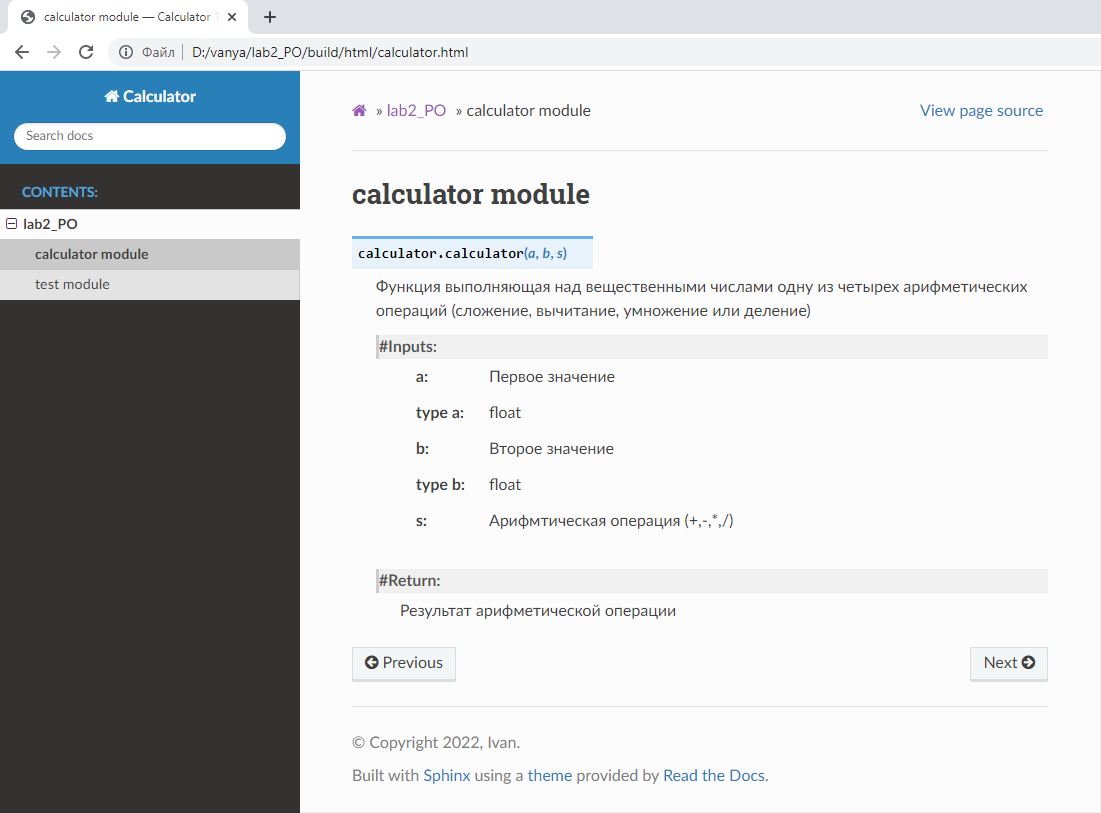
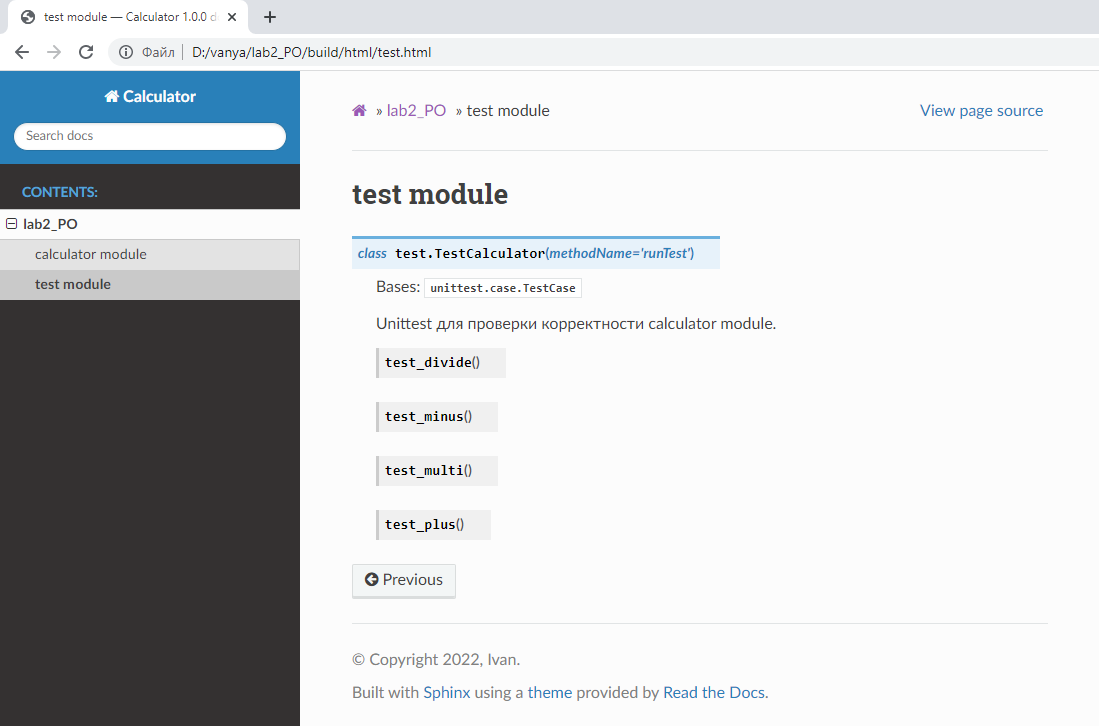
**static** — в эту директорию помещаются все файлы, не являющиеся исходным кодом (например, изображения). Позже создаются связи этих файлов с директорией build.

**conf.py** — содержит конфигурационные параметры Sphinx, включая те, которые были выбраны при запуске sphinx-quickstart в окне терминала.

**index.rst** — это корень проекта. Он соединяет документацию воедино, если она разделена на несколько файлов.

Результат:

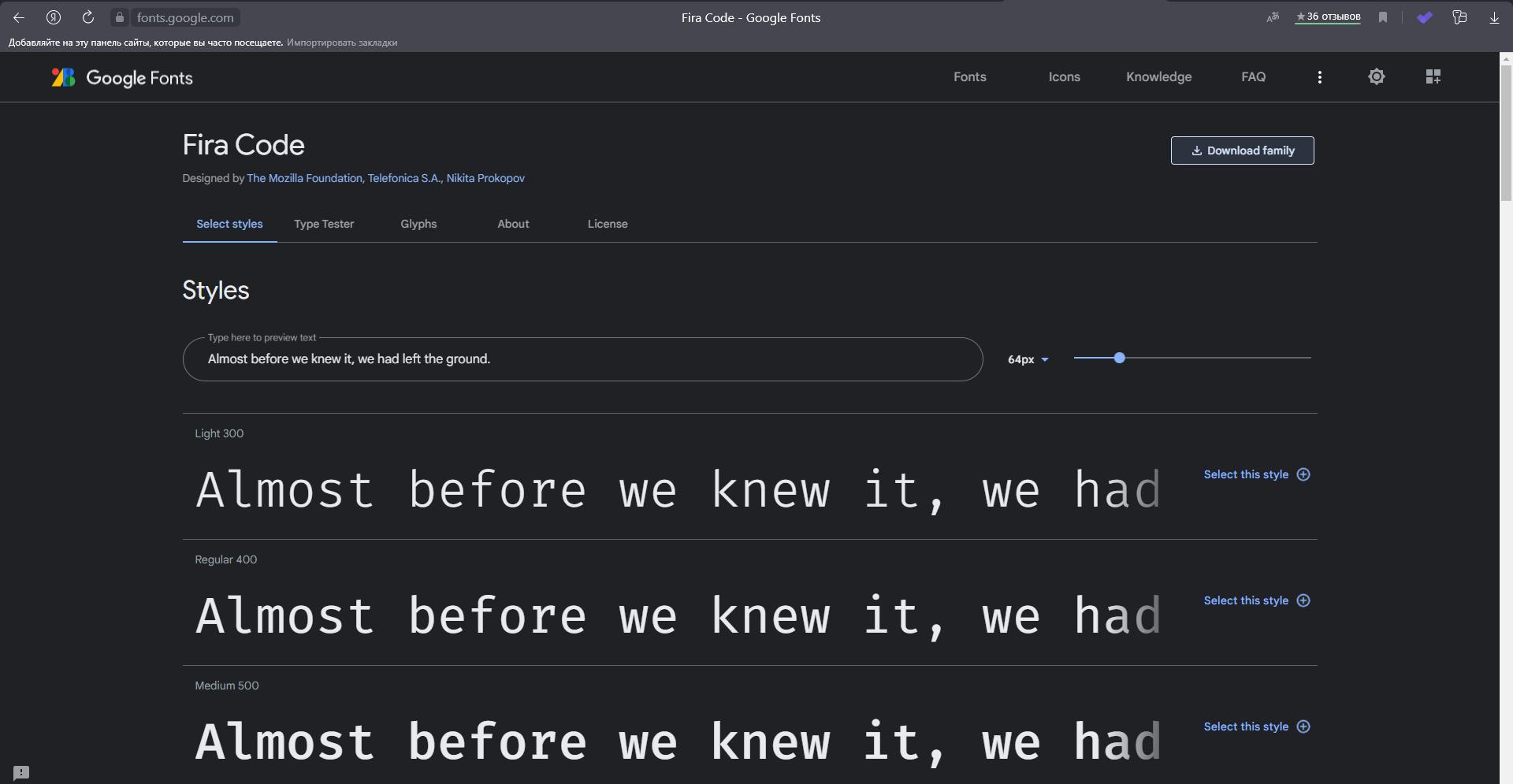




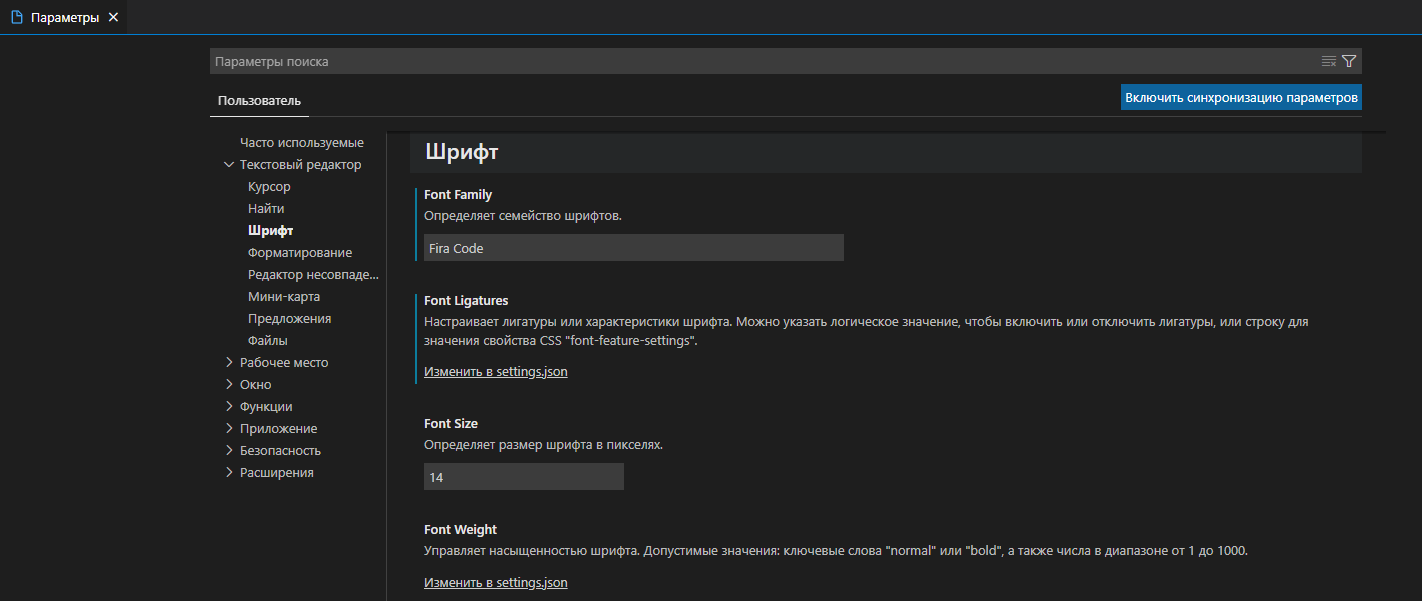
**9. Установить шрифт FiraCode**

Fira code - моноширинный шрифт, в котором используются лигатуры (объединяет несколько символов в один) для общих комбинаций символов в программировании. Это только визуальная надстройка, с которой читать и понимать код быстрее.

Для его скачивания воспользуемся сайтом:fonts.google.com

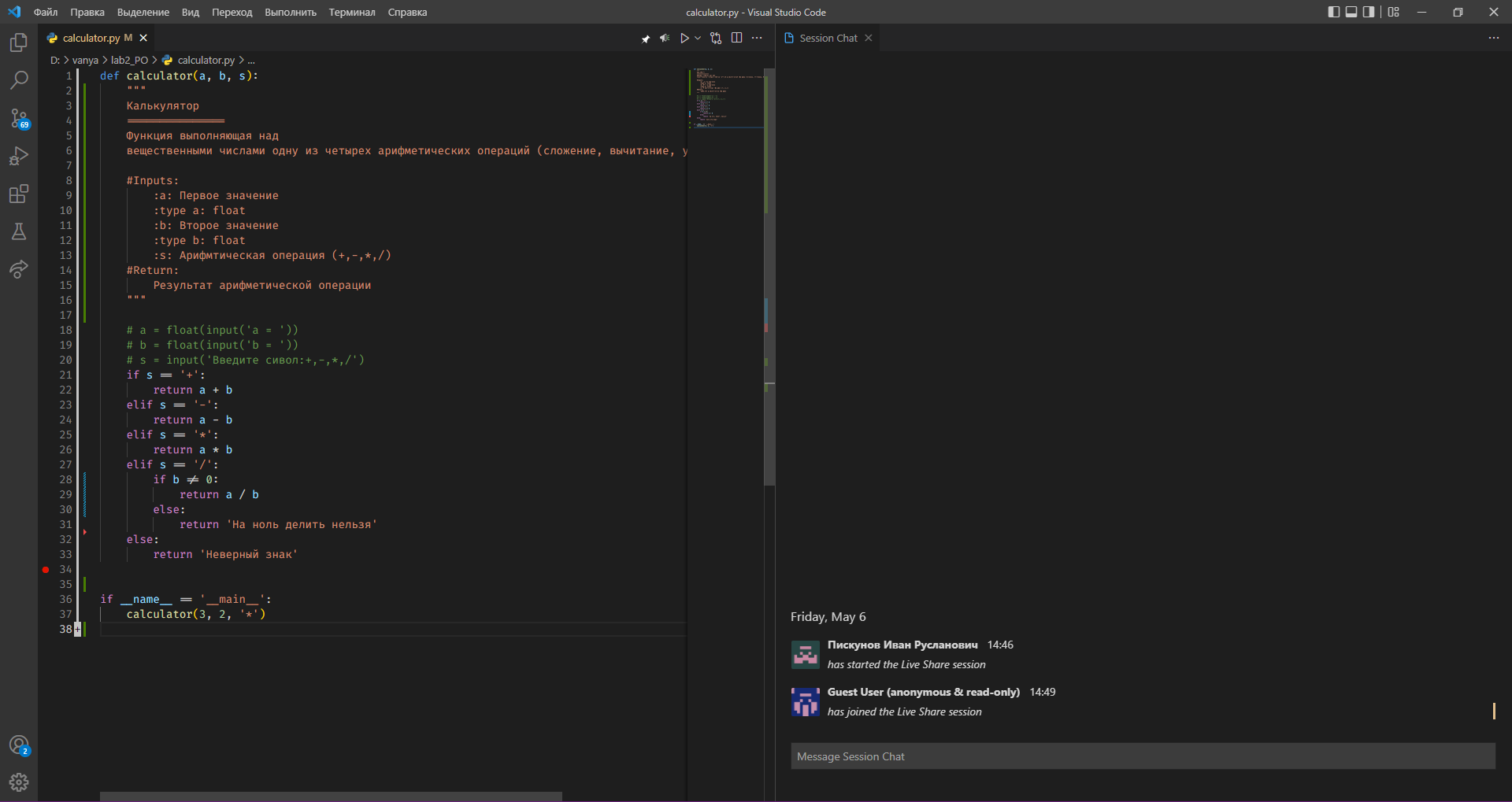


После скачивания архива и распаковки, нам нужно в настройках VS Code добавить шрифт Fira Code



В итоге получим:

**10. Live Share**

****

**Вывод:** В этой лабораторной работе мною был изучен функционал VS Code. Была произведена настройка Visual Studio Code для python, написан простой калькулятор, а также тесты для него. Кроме этого, я поработал в командной строке и написал пару сниппетов. Установил Sphinx и узнал о его возможностях, в качестве шрифта был использован Fira Font. В конце был установлен Live Share