

# **Инвариантное задание 1**

## **Задание 1.1**

Подготовить обзор программных продуктов, применяемых в организации, где вы проходите практику.

План обзора программного продукта:

- общая характеристика;
- функции;
- необходимое программное и аппаратное обеспечение

### **Общая характеристика**

Visual Studio Code — это легкий, но мощный редактор исходного кода, который работает на вашем рабочем столе и доступен для Windows, macOS и Linux. Он поставляется со встроенной поддержкой JavaScript, TypeScript и Node.js а также имеет богатую экосистему расширений для других языков и сред выполнения (таких как C++, C#, Java, Python, PHP, Go, .NET).

### **Базовая планировка**

VS Code поставляется с простым и интуитивно понятным макетом, который максимально увеличивает пространство, предоставляемое редактору, оставляя достаточно места для просмотра и доступа к полному контексту вашей папки или проекта. Пользовательский интерфейс разделен на пять основных областей:

- Редактор - основная область для редактирования файлов. Вы можете открыть столько редакторов, сколько захотите, рядом по вертикали и горизонтали. (Рисунок 1)

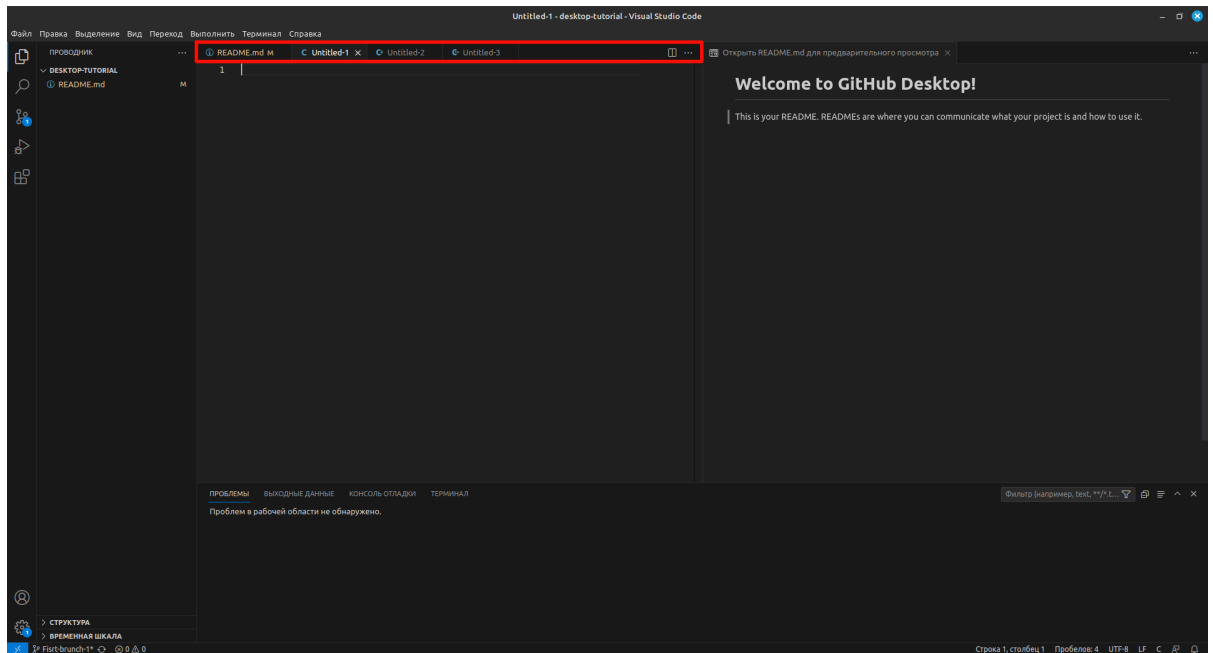


Рисунок 1.

- Основная боковая панель - содержит различные представления, такие как проводник, чтобы помочь вам в работе над вашим проектом. (Рисунок 2)

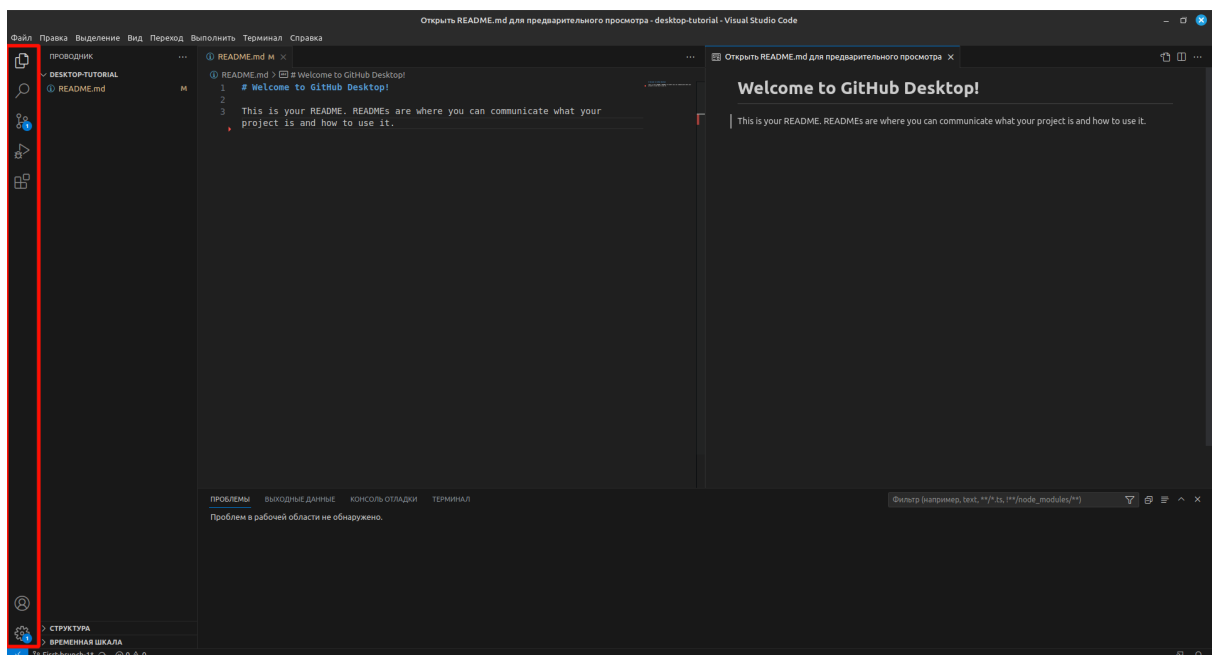


Рисунок 2.

- Строка состояния - Информация об открытом проекте и редактируемых файлах.  
(Рисунок 3)

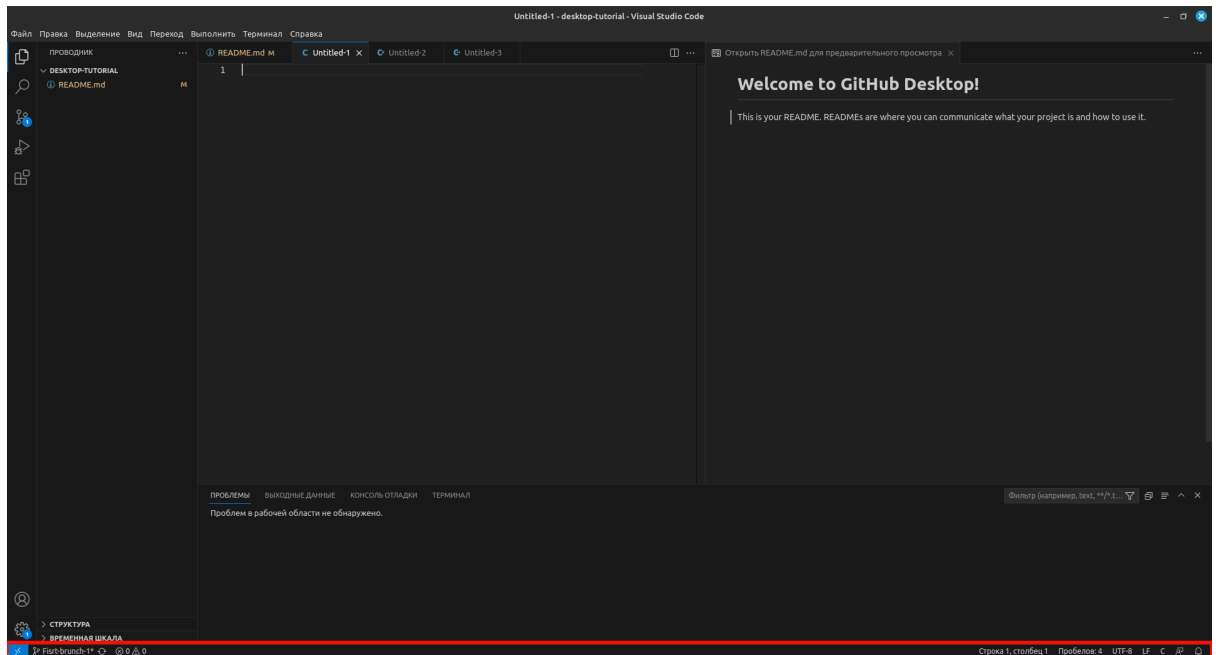


Рисунок 3.

- Панель активности - расположена в крайнем левом углу, позволяет переключаться между представлениями и дает вам дополнительные контекстно-зависимые индикаторы, такие как количество исходящих изменений при включении Git. (Рисунок 4)

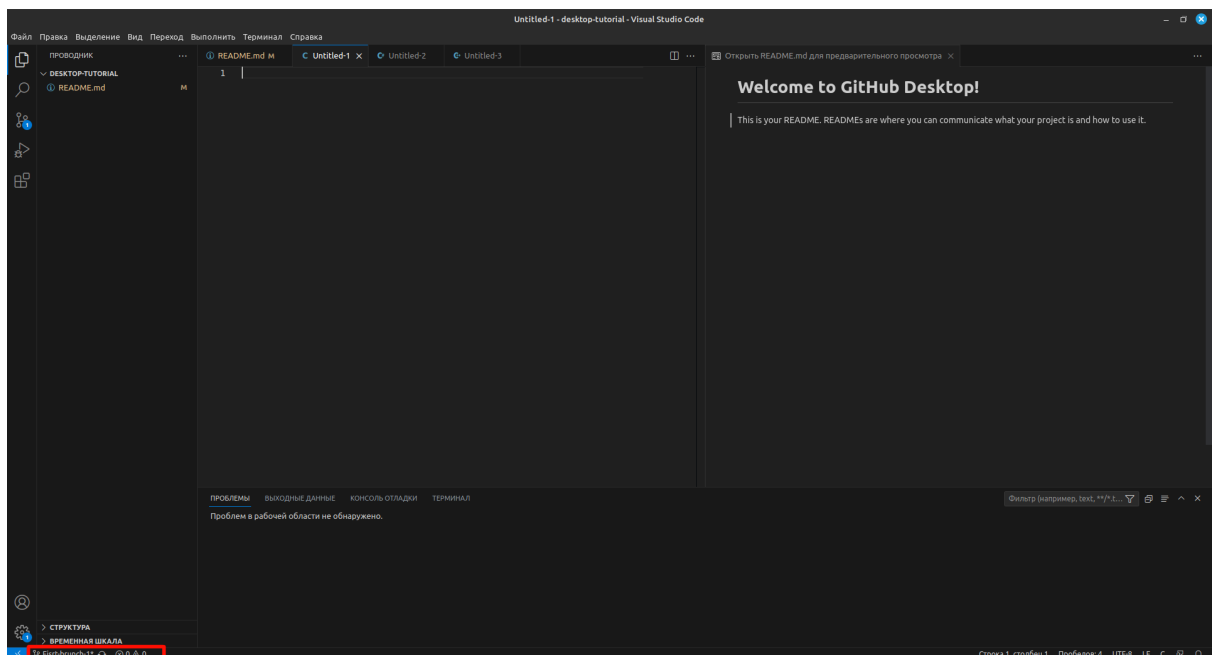


Рисунок 4.

- Панель — дополнительное пространство для представлений под областью редактора. По умолчанию в нем размещаются выходные данные, отладочная информация, ошибки и предупреждения, а также встроенный терминал. Панель также можно перемещать влево или вправо для большего вертикального пространства. (Рисунок 5)

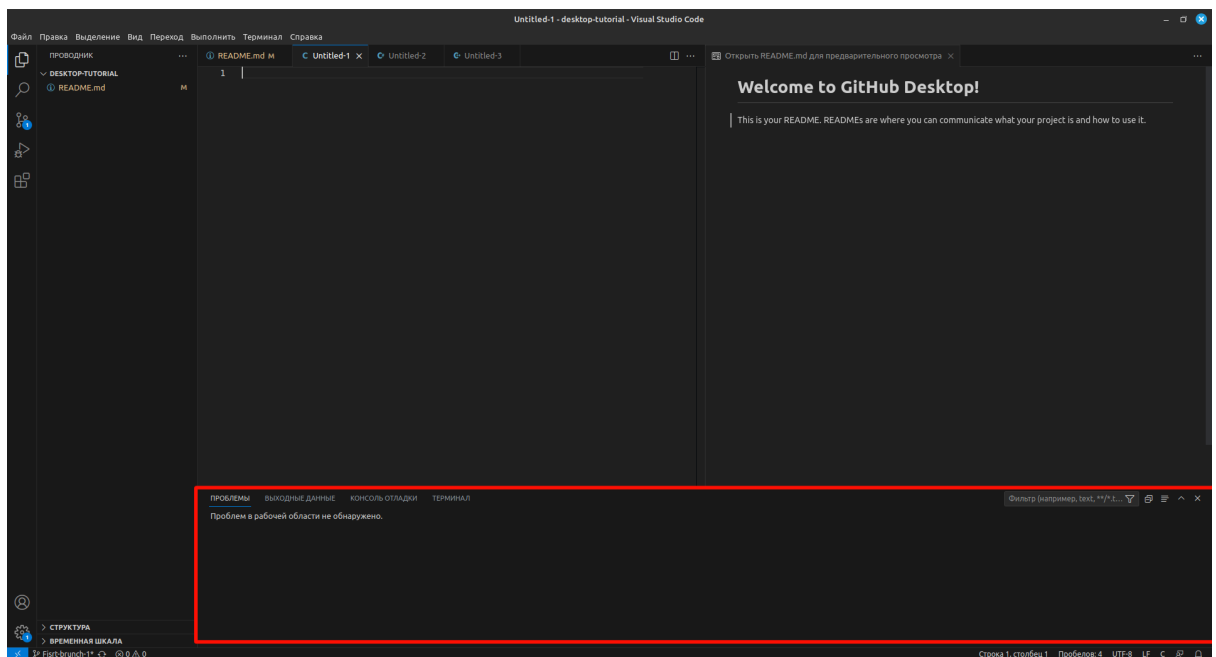


Рисунок 5.

## Функции

### 1. Создание проекта

Для создания необходимо в верхней панели выбрать вкладку «Файл», затем открыть папку, где будет находится код программы. (Рисунок 6)

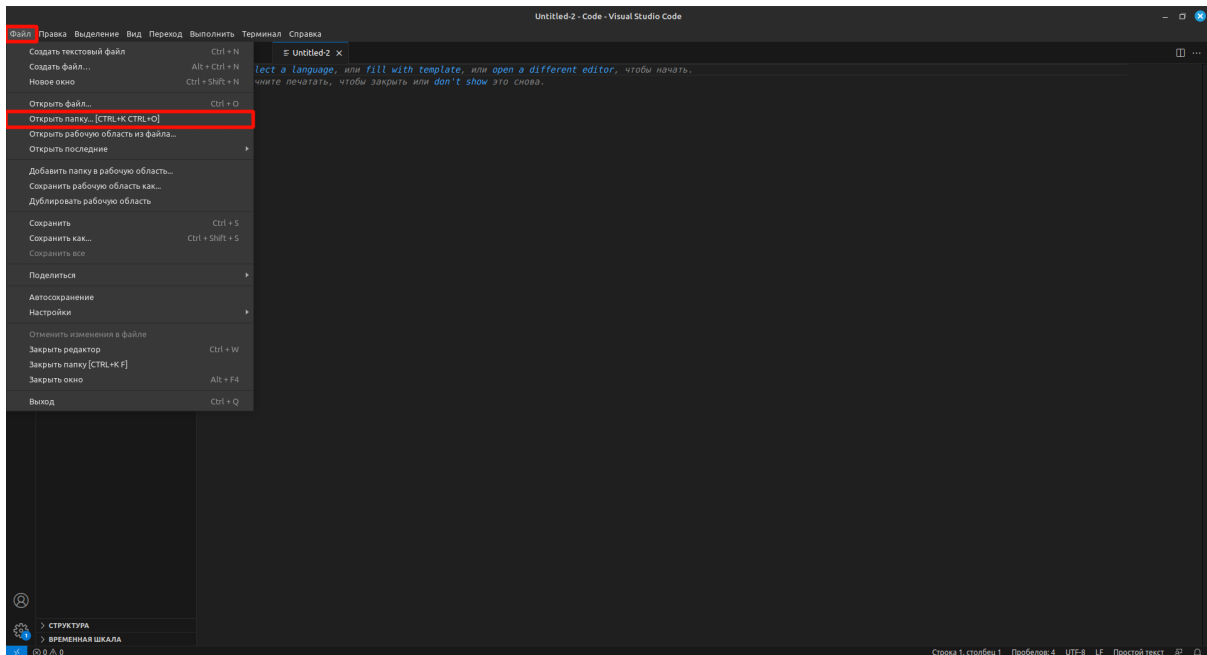


Рисунок 6.

Затем в верхней панели выбрать вкладку «Файл», затем «Создать файл» и выбираем язык программирования. (Рисунок 7)

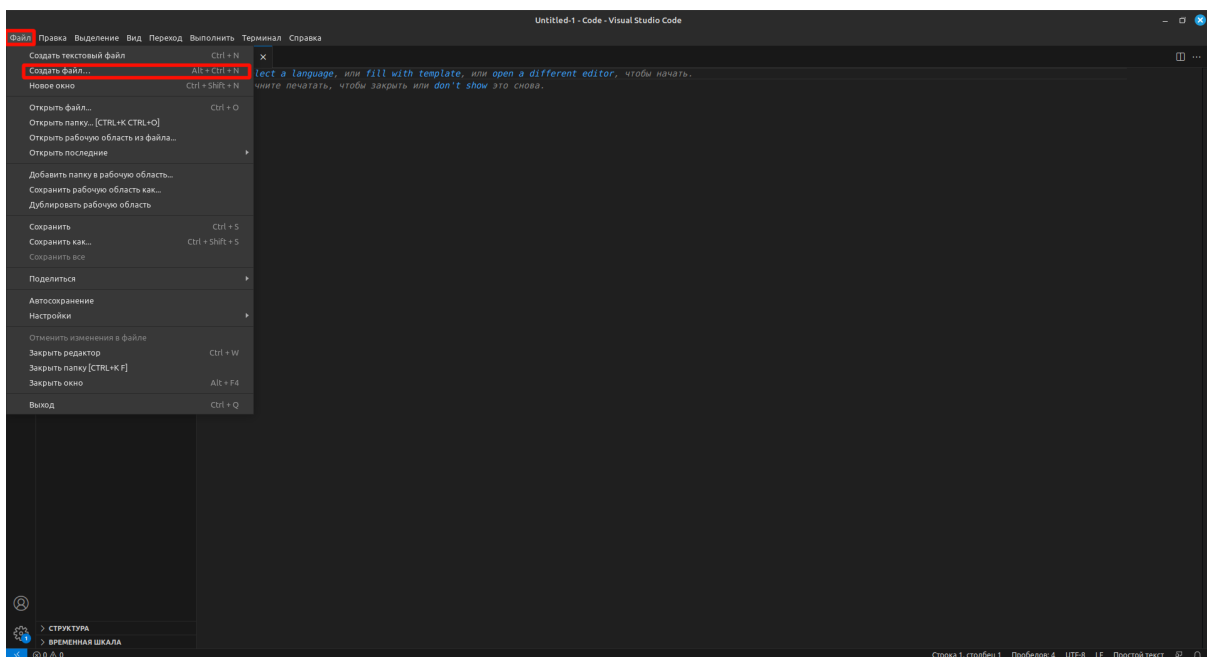


Рисунок 7.

Далее пишем код, сохраняем и запускаем

## 2. Кодирование

## Цветовая раскраска синтаксических конструкций

Некоторые элементы синтаксиса кода и файлов разметки для наглядности выделяются разными цветами. Например, ключевые слова (такие как `using` в C# и `Imports` в Visual Basic) выделены одним цветом, а типы (такие как `Console` и `Uri`) — другим. Другие элементы синтаксиса (например, строковые литералы и комментарии) также выделены цветом. Язык C++ использует цвета для различения типов, перечислений и макросов среди других токенов. (Рисунок 8).

```
14     return;
15 }
16 for (int i=0;i<numStudents;i++){
17     fprintf(fp,"%d,%s,%s,%.2f\n", students[i].id,students[i].name,
18     students[i].faculty, students[i].rating);}
19     fclose(fp);
20     printf("Данные успешно записаны в файл.\n");
21 }
22 void readStudents(struct Student students[], int* numStudents,
23 char* filename){
24     FILE* fp=fopen(filename, "r");
25     if (fp==NULL){
26         printf("Ошибка при открытии файла для чтения.\n");
27         return;
28     }
29     char line[256];
30     int i=0;
```

Рисунок 8.

## Маркеры ошибок и предупреждений

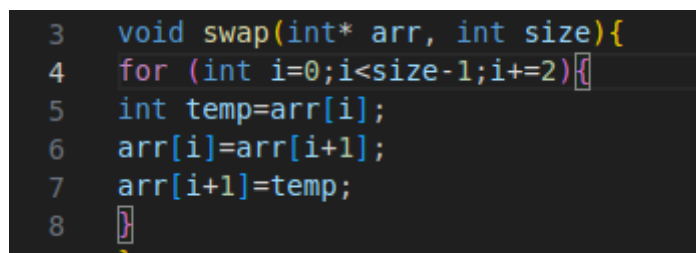
В процессе добавления кода и сборки решения вы можете увидеть в коде волнистые линии различного цвета (знак «тильда») или лампочки. Красные волнистые линии обозначают ошибки синтаксиса, синие обозначают ошибку компилятора, зеленые — предупреждения, а фиолетовые — другие типы ошибок. (Рисунок 9).

```
15     printfhf("Before swapping: ");
16     four (int i=0;i<12;i++){
17         print4f("%d ", arr[i]);
18     }
```

Рисунок 9.

## Согласование скобок

Если курсор мыши поместить на открывающую фигурную скобку в файле кода, выделяются обе скобки — открывающая и закрывающая. Эта функция позволяет оперативно реагировать на неправильно поставленную или отсутствующую фигурную скобку. (Рисунок 10).

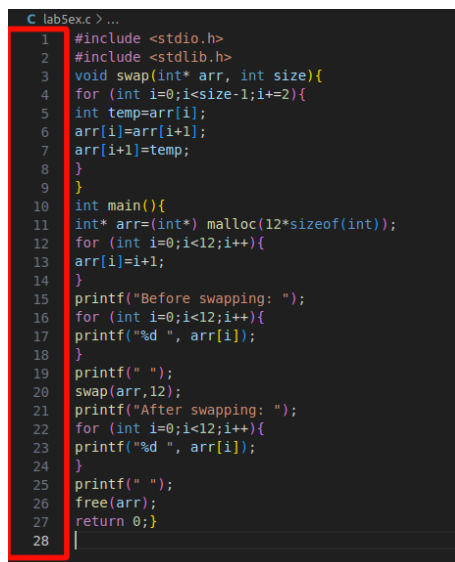


```
3 void swap(int* arr, int size){
4   for (int i=0;i<size-1;i+=2){
5     int temp=arr[i];
6     arr[i]=arr[i+1];
7     arr[i+1]=temp;
8   }
```

Рисунок 10.

## Номера строк

Номера строк могут отображаться в левом поле окна кода.(Рисунок 11).



```
C lab5ex.c > ...
1  #include <stdio.h>
2  #include <stdlib.h>
3  void swap(int* arr, int size){
4    for (int i=0;i<size-1;i+=2){
5      int temp=arr[i];
6      arr[i]=arr[i+1];
7      arr[i+1]=temp;
8    }
9  }
10 int main(){
11   int* arr=(int*) malloc(12*sizeof(int));
12   for (int i=0;i<12;i++){
13     arr[i]=i+1;
14   }
15   printf("Before swapping: ");
16   for (int i=0;i<12;i++){
17     printf("%d ", arr[i]);
18   }
19   printf(" ");
20   swap(arr,12);
21   printf("After swapping: ");
22   for (int i=0;i<12;i++){
23     printf("%d ", arr[i]);
24   }
25   printf(" ");
26   free(arr);
27   return 0;}
28 |
```

Рисунок 11.

## Форматирование кода

Автоматическое форматирование кода в Visual Studio Code выполняется следующими комбинациями:

- Windows: Shift + Alt + F
- Mac: Shift ⌘ + Option ⌥ + F
- Ubuntu: Ctrl + Shift + I

## Отладка и запуск

Отладку можно запустить сочетанием клавиш **Ctrl+Shift+D**. В представлении «Запуск и отладка» отображаются все сведения, связанные с запуском и отладкой, и имеется верхняя панель с командами отладки и параметрами конфигурации. (Рисунок 12).

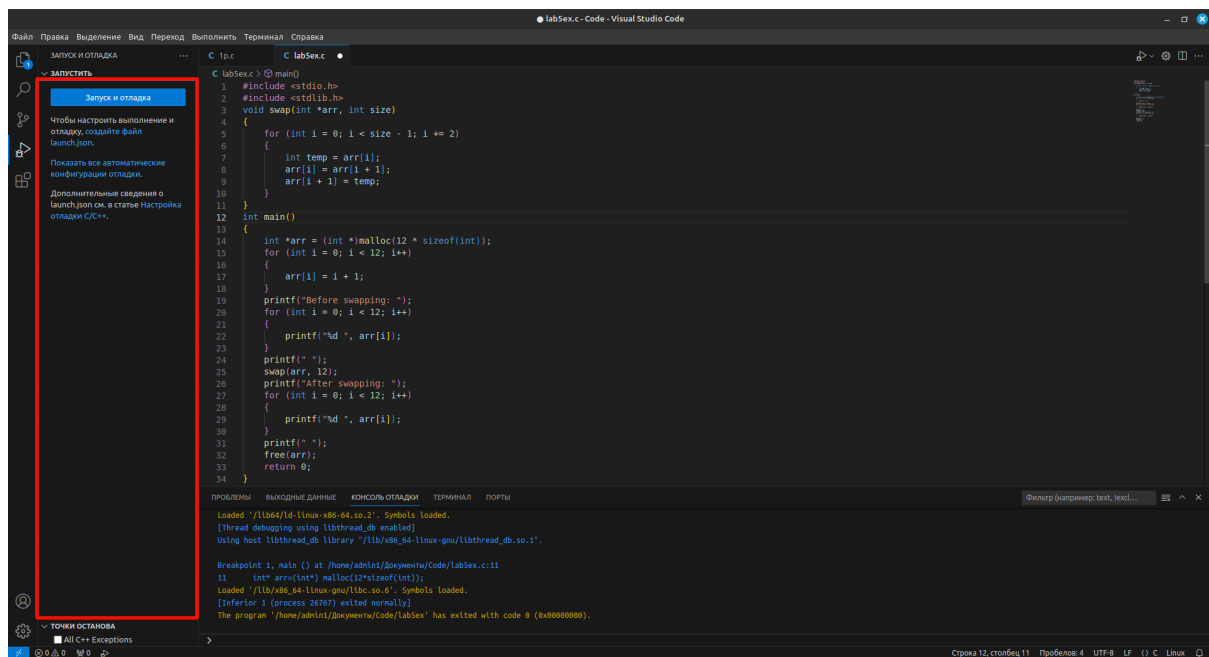


Рисунок 12.



Также запуск или отладку можно выполнить, нажав стрелочку, показанную на рисунке. (Рисунок 13).

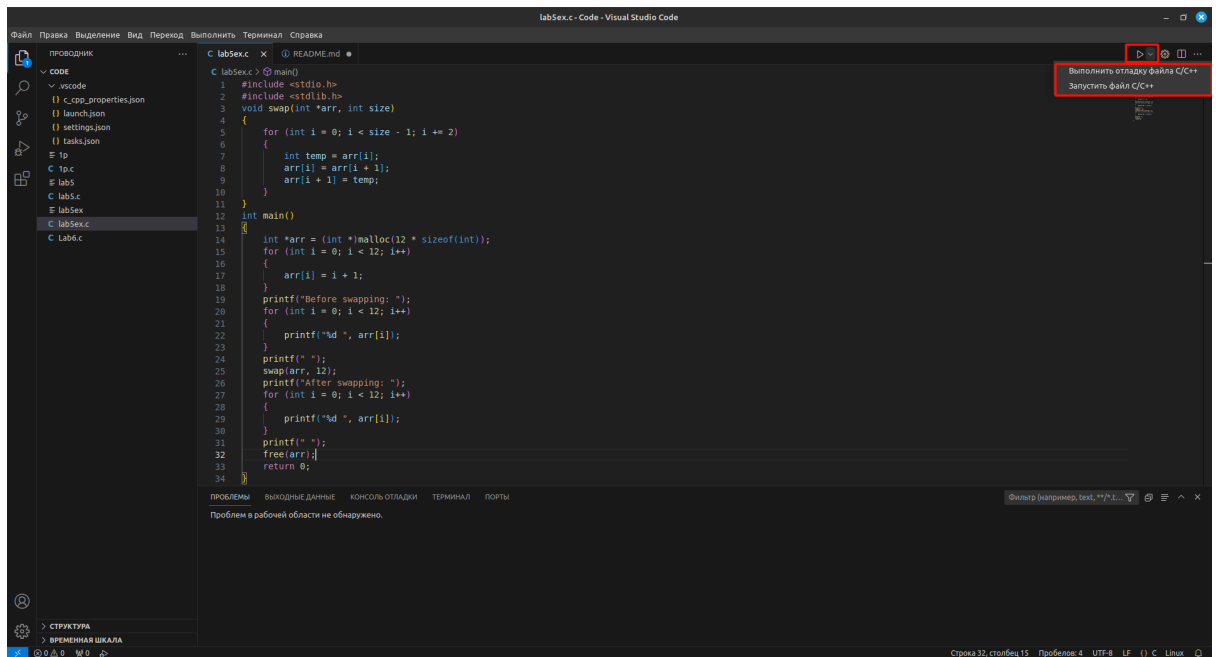


Рисунок 13.

## Публикация в репозитории

Открываем GitHub Desktop, нажимаем CTRL+Shift+P и создаем ветку.(Рисунок 14).

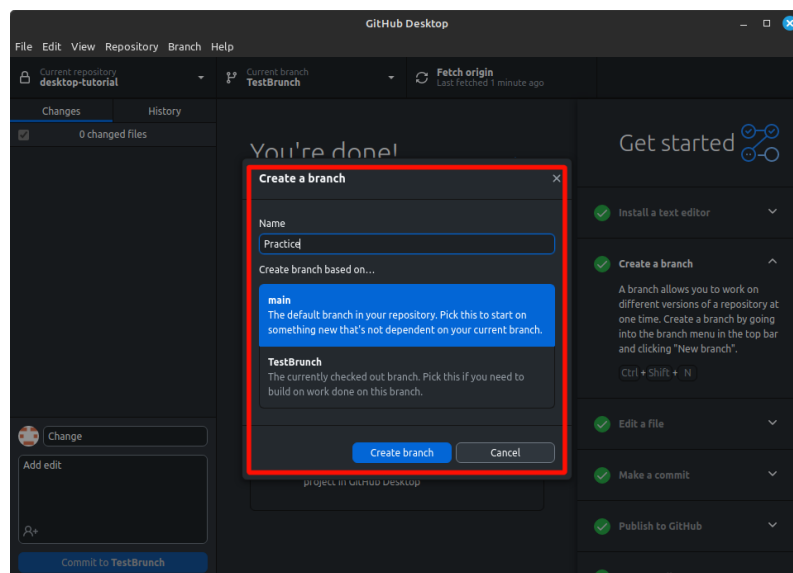


Рисунок 14.

Затем нажимаем CTRL+SHIFT+A откроется VS Code делаем изменения в коде, сохраняем и заходим в GitHub Desktop. Добавляем в описание, что изменилось (Рисунок 15).

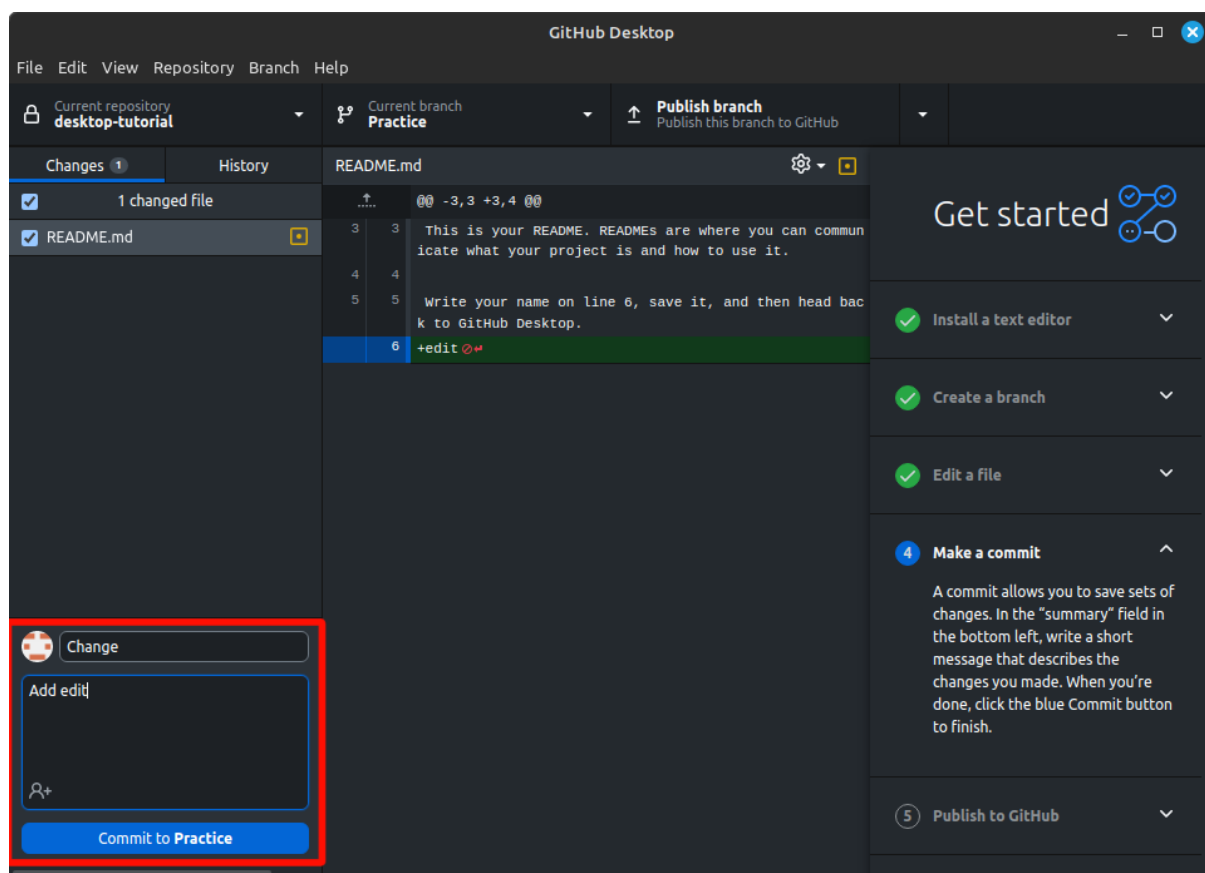


Рисунок 15.

Затем нажимаем CTRL+P и обновленная версия файла публикуется на GitHub.

### **Необходимое программное и аппаратное обеспечение**

Visual Studio Code — это небольшой загружаемый файл (< 200 МБ), занимающий место на диске < 500 МБ. VS Code является легким и должен легко работать на современном оборудовании.

Системные требования:

- Процессор с тактовой частотой 1,6 ГГц или выше
- 1 ГБ оперативной памяти или выше

VS Code поддерживается на следующих платформах:

- Windows 10 и 11 (32-разрядная и 64-разрядная)

- Версии macOS с поддержкой обновлений безопасности Apple. Как правило, это последняя версия и две предыдущие версии.
- Linux (Debian): Ubuntu Desktop 16.04, Debian 9
- Linux (Red Hat): Red Hat Enterprise Linux 7, CentOS 7, Fedora 34