

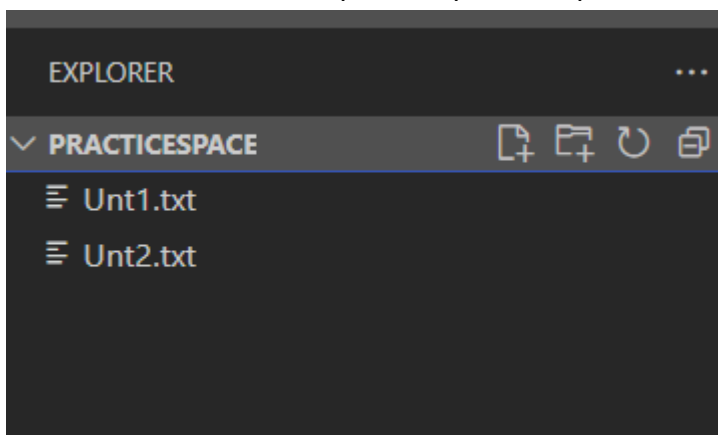
Отчёт по выполнению практического задания №1

Создание локального репозитория

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

PS C:\Users\Тест> cd D:\practicespace
PS D:\practicespace> git init
Reinitialized existing Git repository in D:/practicespace/.git/
PS D:\practicespace> 
```

добавляем 2 текстовых файла в репозиторий



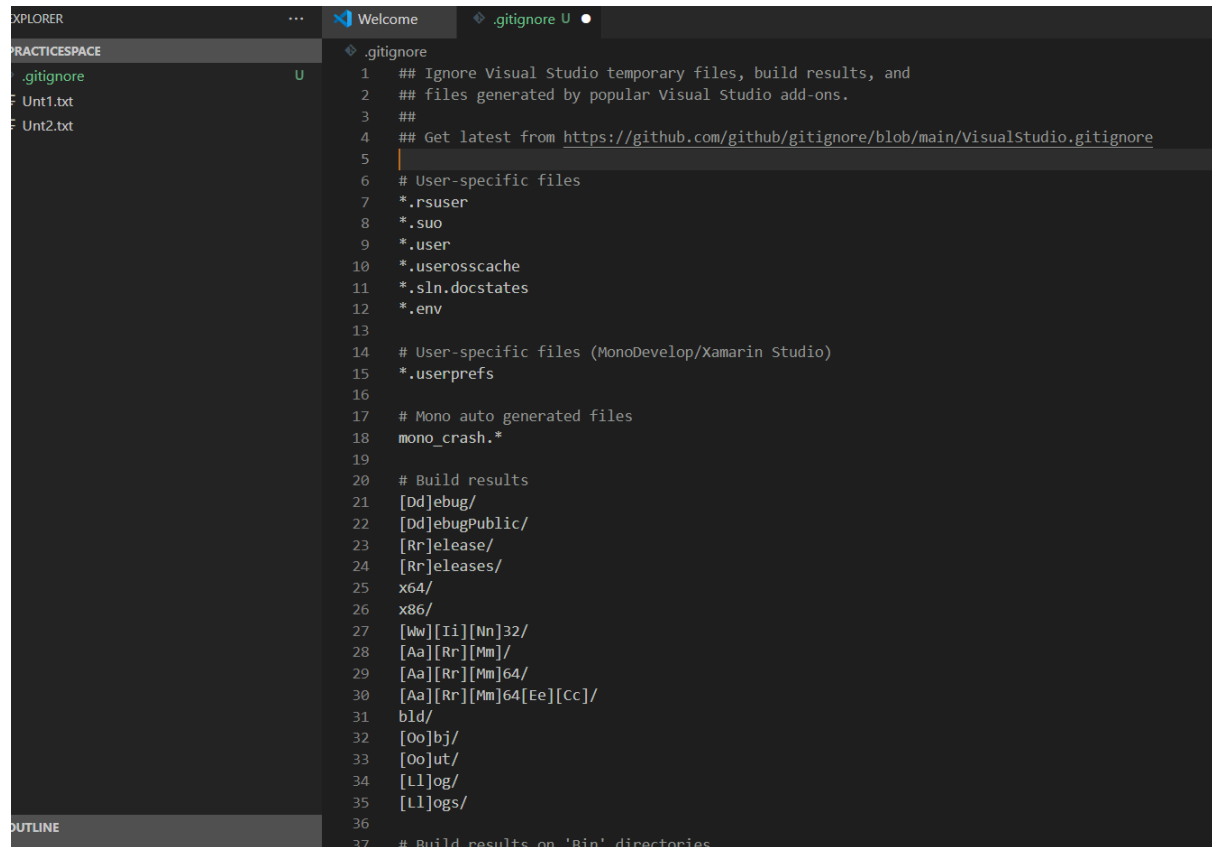
создаем первый коммит

```
+ FullyQualifiedErrorId : CommandNotFoundException

PS D:\practicespace> commit -m "added first commit"
commit : Имя "commit" не распознано как имя командлета, функции, файла сценария или выполняемой программы. Проверьте
строка:1 знак:1
+ commit -m "added first commit"
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (commit:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

PS D:\practicespace> git commit -m "added first commit"
[master (root-commit) 397e389] added first commit
 2 files changed, 2 insertions(+)
 create mode 100644 Unt1.txt
 create mode 100644 Unt2.txt
PS D:\practicespace> 
```

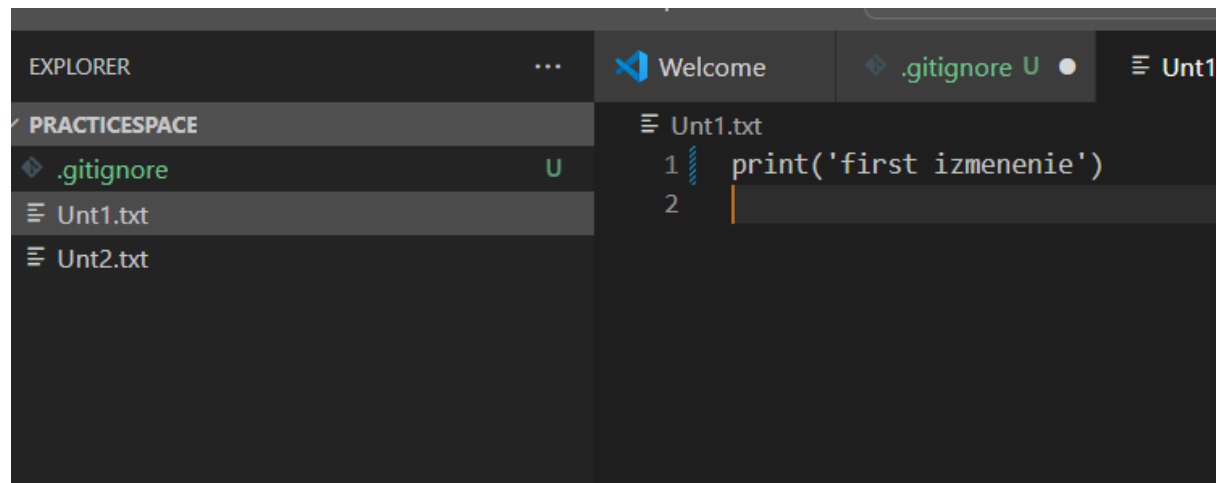
добавляем файл гитигнор



The screenshot shows the Visual Studio interface with the Explorer pane on the left displaying the 'PRACTICESPACE' folder containing '.gitignore', 'Unt1.txt', and 'Unt2.txt'. The main editor pane shows the content of the '.gitignore' file, which is a standard Visual Studio template. The file includes comments about ignoring temporary files, build results, and user-specific files, followed by a list of file patterns to be ignored.

```
.gitignore
1 ## Ignore Visual Studio temporary files, build results, and
2 ## files generated by popular Visual Studio add-ons.
3 ##
4 ## Get latest from https://github.com/github/gitignore/blob/main/VisualStudio.gitignore
5
6 # User-specific files
7 *.rsuser
8 *.suo
9 *.user
10 *.userosscache
11 *.sln.docstates
12 *.env
13
14 # User-specific files (MonoDevelop/Xamarin Studio)
15 *.userprefs
16
17 # Mono auto generated files
18 mono_crash.*
19
20 # Build results
21 [Dd]ebug/
22 [Dd]ebugPublic/
23 [Rr]elease/
24 [Rr]eleases/
25 x64/
26 x86/
27 [Ww][Ii][Nn]32/
28 [Aa][Rr][Mm]/
29 [Aa][Rr][Mm]64/
30 [Aa][Rr][Mm]64[Ee][Cc]/
31 bld/
32 [Oo]bj/
33 [Oo]ut/
34 [Ll]og/
35 [Ll]ogs/
36
37 # Build results on 'Bin' directories
```

изменяем один из файлов



The screenshot shows the Visual Studio interface with the Explorer pane on the left displaying the 'PRACTICESPACE' folder. The main editor pane shows the content of the 'Unt1.txt' file, which has been modified to contain a single line of C# code: 'print('first izmenenie')'.

```
Unt1.txt
1 print('first izmenenie')
2
```

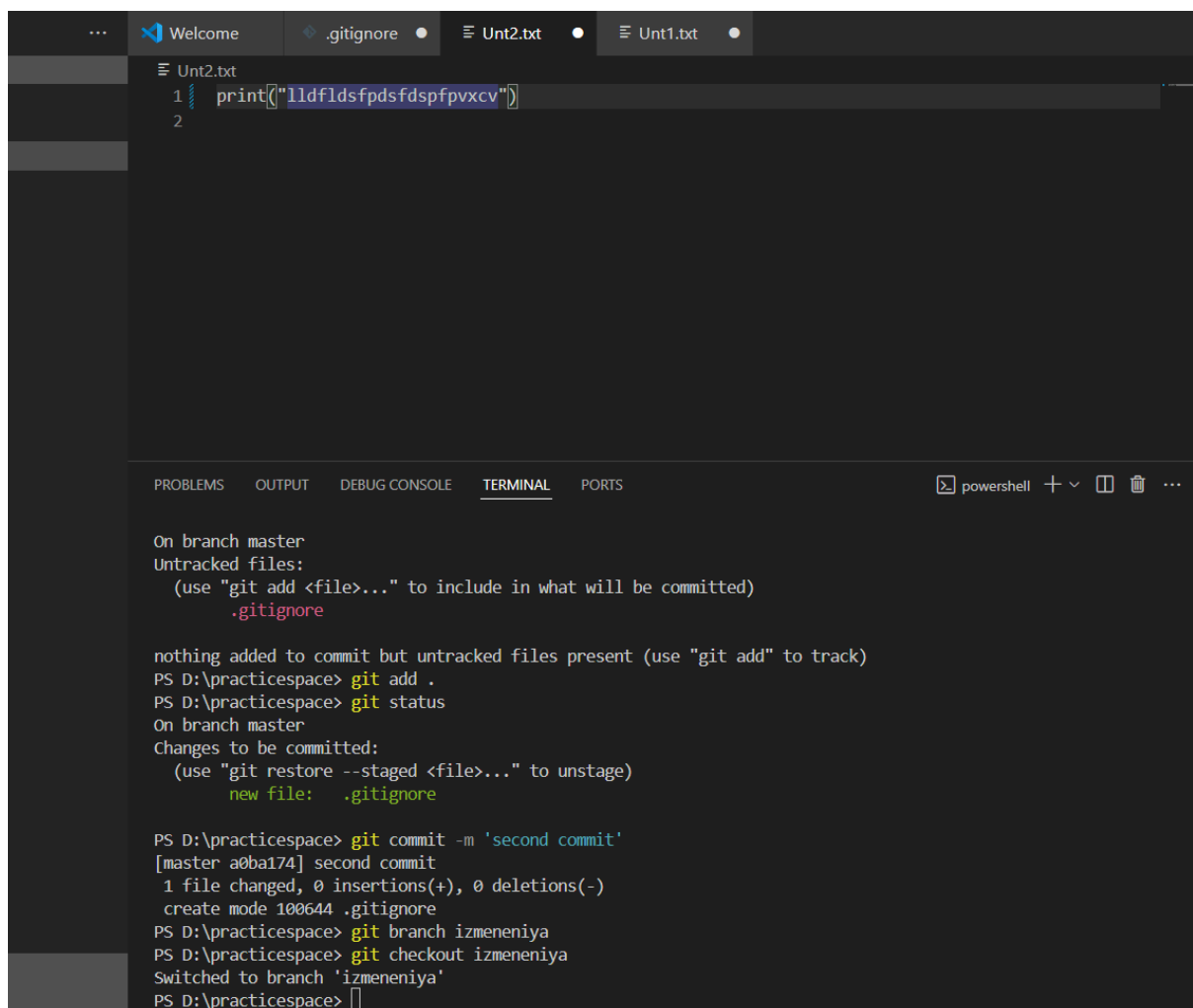
добавляем второй коммит

```
PS D:\practicespace> git commit -m "second commit"
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore

nothing added to commit but untracked files present (use "git add" to track)
PS D:\practicespace> git add .
PS D:\practicespace> git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   .gitignore

PS D:\practicespace> git commit -m 'second commit'
[master a0ba174] second commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 .gitignore
PS D:\practicespace>
```

создаем новую ветку, переключаемся и делаем изменения в файле



The screenshot shows the Visual Studio Code interface. The editor has a tab for 'Unt2.txt' which contains the following code:

```
1 print(["lldfldsfpsfdspfpvxcv"])
2
```

The bottom panel shows the 'TERMINAL' tab with a PowerShell session. The output of the terminal is as follows:

```
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore

nothing added to commit but untracked files present (use "git add" to track)
PS D:\practicespace> git add .
PS D:\practicespace> git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   .gitignore

PS D:\practicespace> git commit -m 'second commit'
[master a0ba174] second commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 .gitignore
PS D:\practicespace> git branch izmeneniya
PS D:\practicespace> git checkout izmeneniya
Switched to branch 'izmeneniya'
PS D:\practicespace>
```

делаем слияние

```
PS D:\practicespace> git checkout master
Switched to branch 'master'
PS D:\practicespace> git merge master
PS D:\practicespace> git log --oneline --graph --all
* 397e389 added first commin
PS D:\practicespace> git merge izmeneniya
Already up to date.
PS D:\practicespace> git log --oneline --graph --all
error: switch `l' expects an integer value with an optional k/m/g suffix
PS D:\practicespace> git log --oneline --graph --all
* a0ba174 (HEAD -> master, izmeneniya) second commit
* 397e389 added first commin
On branch master
```

изменяем файл , добавляем 3 коммит

```
+ FullyQualifiedErrorId : CommandNotFoundException

PS D:\practicespace> git add .
PS D:\practicespace> git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   .gitignore
        modified:   Unt1.txt
        modified:   Unt2.txt

PS D:\practicespace> git commit -m '3 commit'
[master 1cedafd] 3 commit
 3 files changed, 420 insertions(+), 2 deletions(-)
```

возвращаемся в ветку и создаем конфликт.

```
merge: testing - not something we can merge
Already up to date.
* 39ccd55 (master) test change
* 1cedafd 3 commit
* 397e389 added first commin
PS D:\practicespace> git merge master
Unt2.txt
Please commit your changes or stash them before you merge.
Aborting
PS D:\practicespace> git commit -am "change"
[izmeneniya 5ba6b8d] change
 1 file changed, 1 insertion(+)
PS D:\practicespace> git merge master
Auto-merging Unt2.txt
CONFLICT (content): Merge conflict in Unt2.txt
Automatic merge failed; fix conflicts and then commit the result.
PS D:\practicespace> git commit -am "razreshaem conflict"
[izmeneniya 9bc9982] razreshaem conflict
PS D:\practicespace> git commit -am "reshaem conflict"
On branch izmeneniya
nothing to commit, working tree clean
```

решаем конфликт

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  [X] p

Automatic merge failed; fix conflicts and then commit the result.
PS D:\practicespace> git commit -am "razreshaem conflict"
[izmeneniya 9bc9982] razreshaem conflict
PS D:\practicespace> git commit -am "reshaem conflict"
On branch izmeneniya
nothing to commit, working tree clean
PS D:\practicespace> git commit -am "eshe raz conflict razreshaem "
[izmeneniya e1a1e4a] eshe raz conflict razreshaem
1 file changed, 2 insertions(+), 6 deletions(-)
PS D:\practicespace>
PS D:\practicespace> git log --oneline --graph --all
* e1a1e4a (HEAD -> izmeneniya) eshe raz conflict razreshaem
* 9bc9982 razreshaem conflict
|\
| * 39ccd55 (master) test change
| * 1cedafd 3 commit
* | 5ba6b8d change
|/
* a0ba174 second commit
* 397e389 added first commin
PS D:\practicespace> 
```