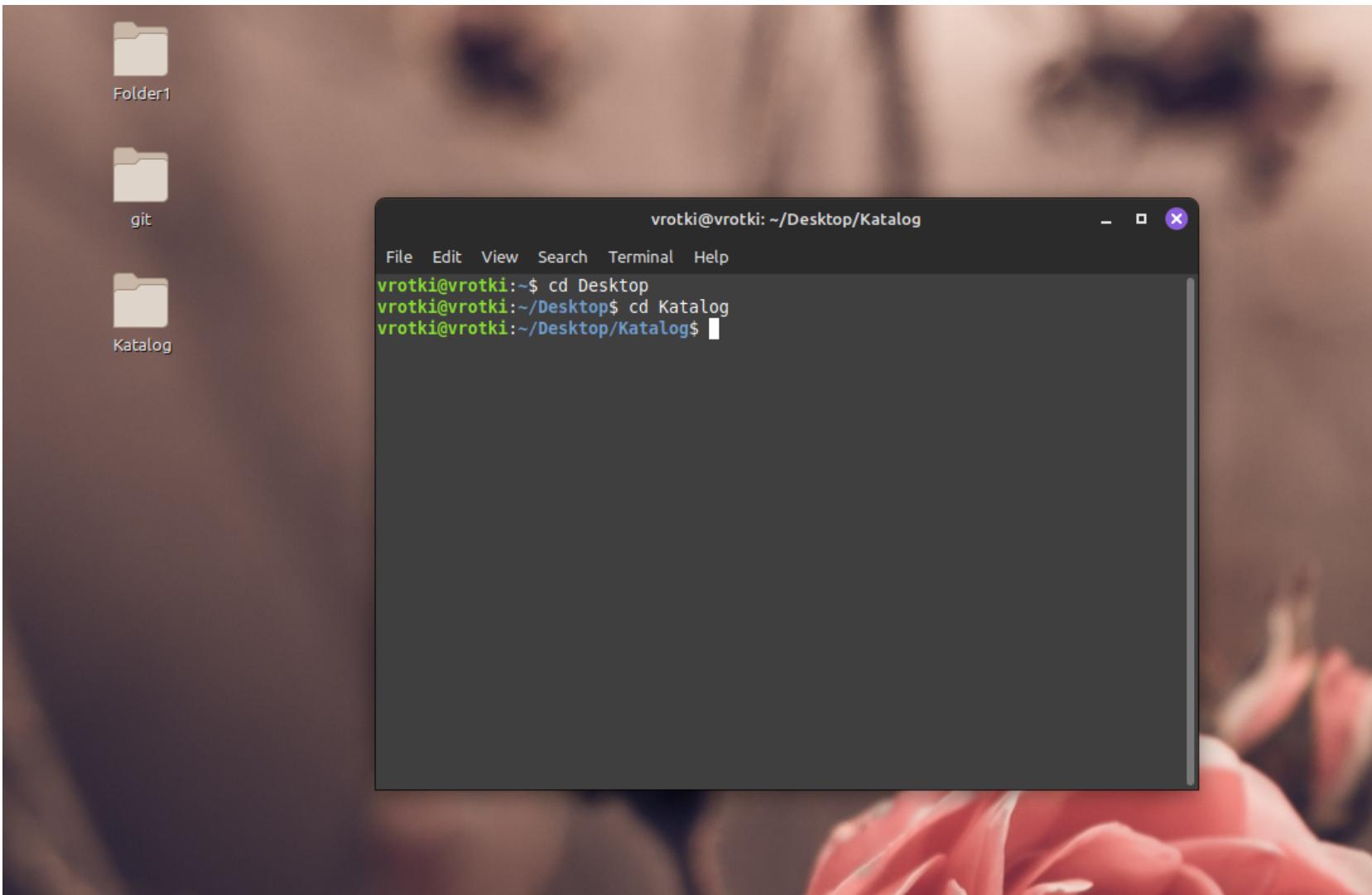


Część 1. Wiersz poleceń

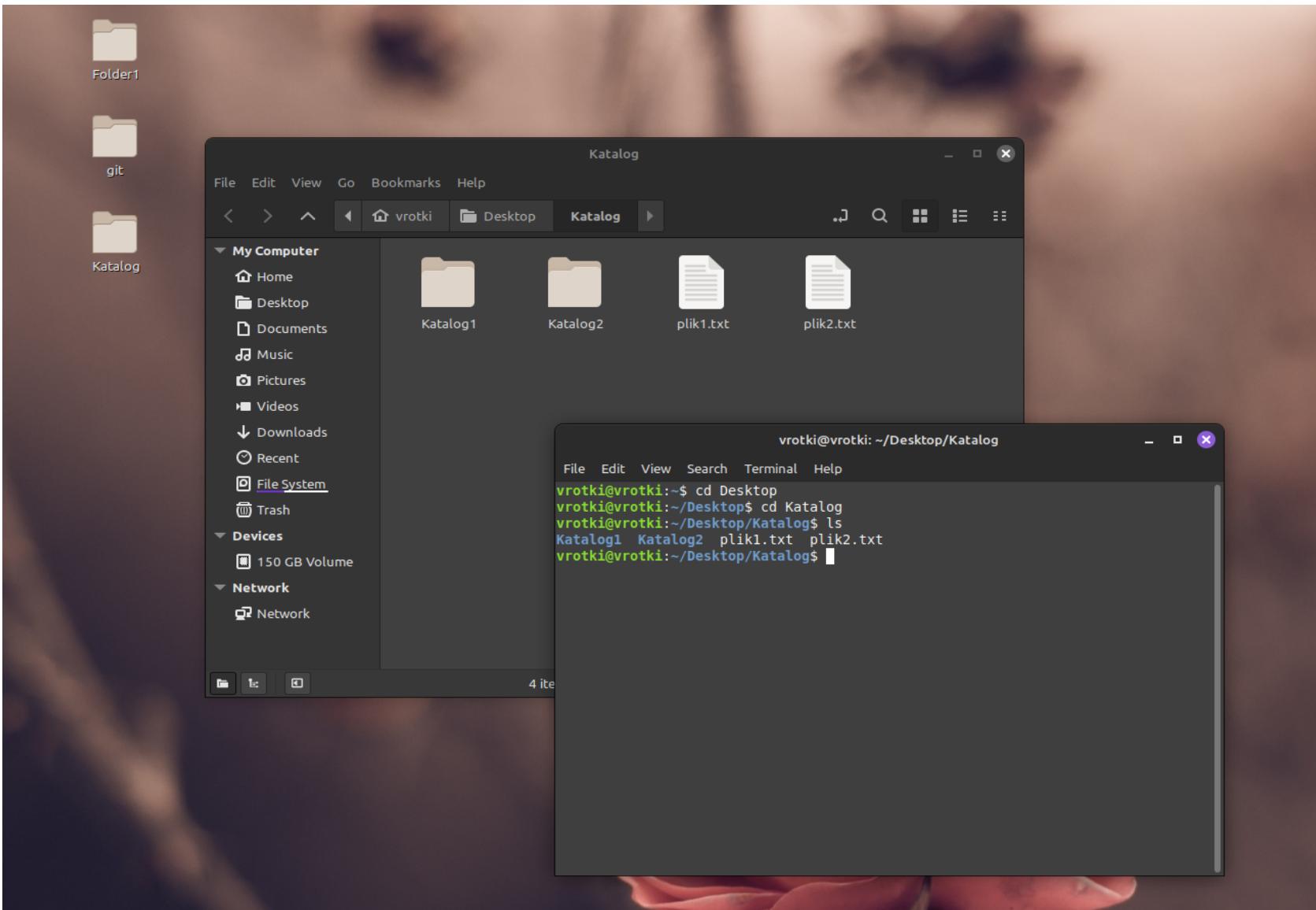
1. Przejdz do dowolnego katalogu, w którym znajduje się co najmniej jeden podkatalog.
2. Wyświetl listę plików/folderów w tym katalogu, a następnie wybierz jeden folder w bieżącym katalogu roboczym.
3. Wyświetl listę plików w wybranym folderze podrzędnym.
4. Wyświetl ponownie jego zawartość, tym razem jednocześnie wyświetlając: 4.a. informacje o właściwym każdego pliku, 4.b. wszystkie ukryte pliki.
5. Wyświetl listę plików w bieżącym katalogu roboczym, **POSORTOWANA** według rozmiaru plików wewnątrz:
 - 5.a. od najmniejszego do największego rozmiaru,
 - 5.b. od największego do najmniejszego rozmiaru.

Część 1. Wiersz poleceń

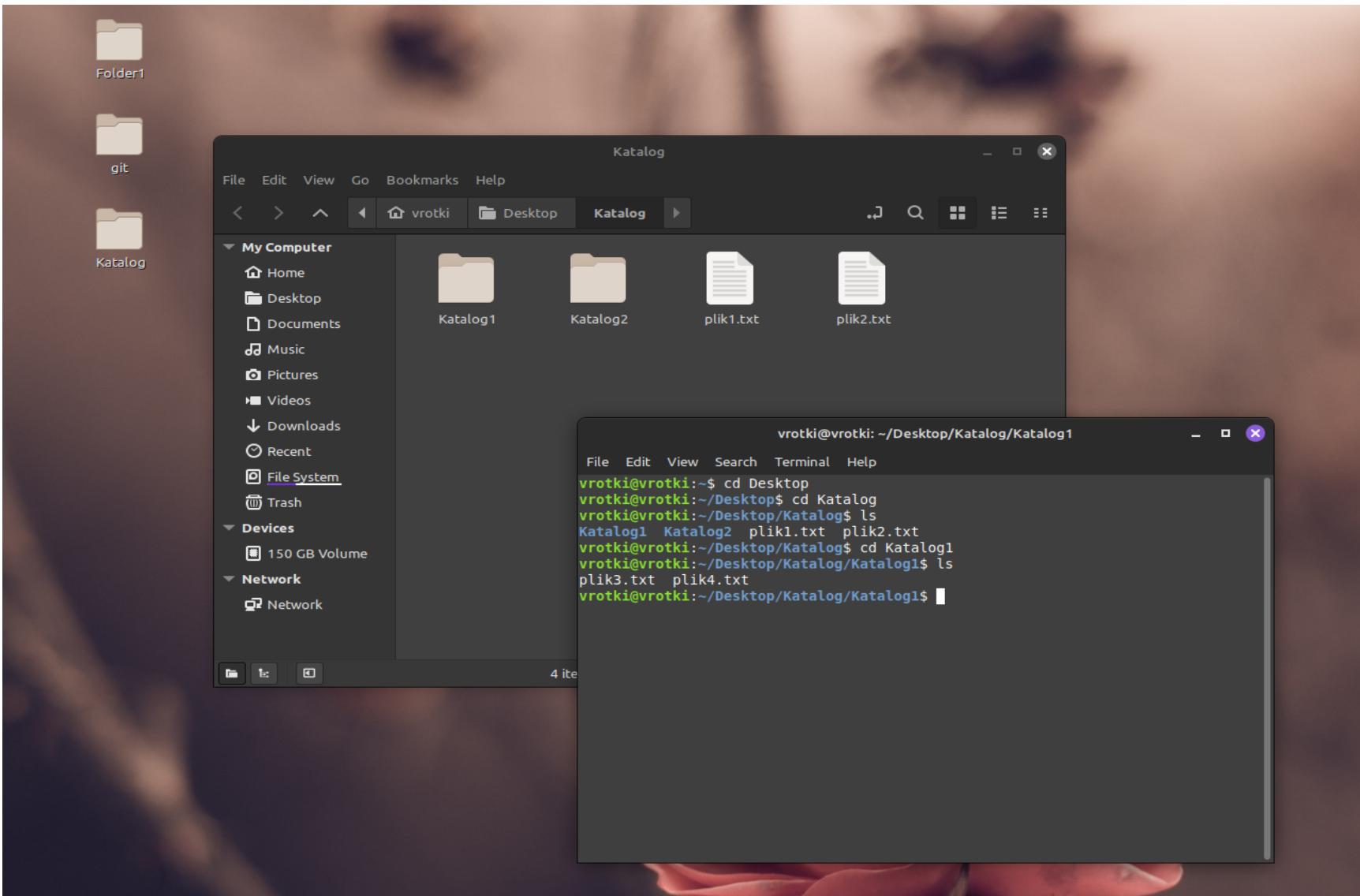
1.



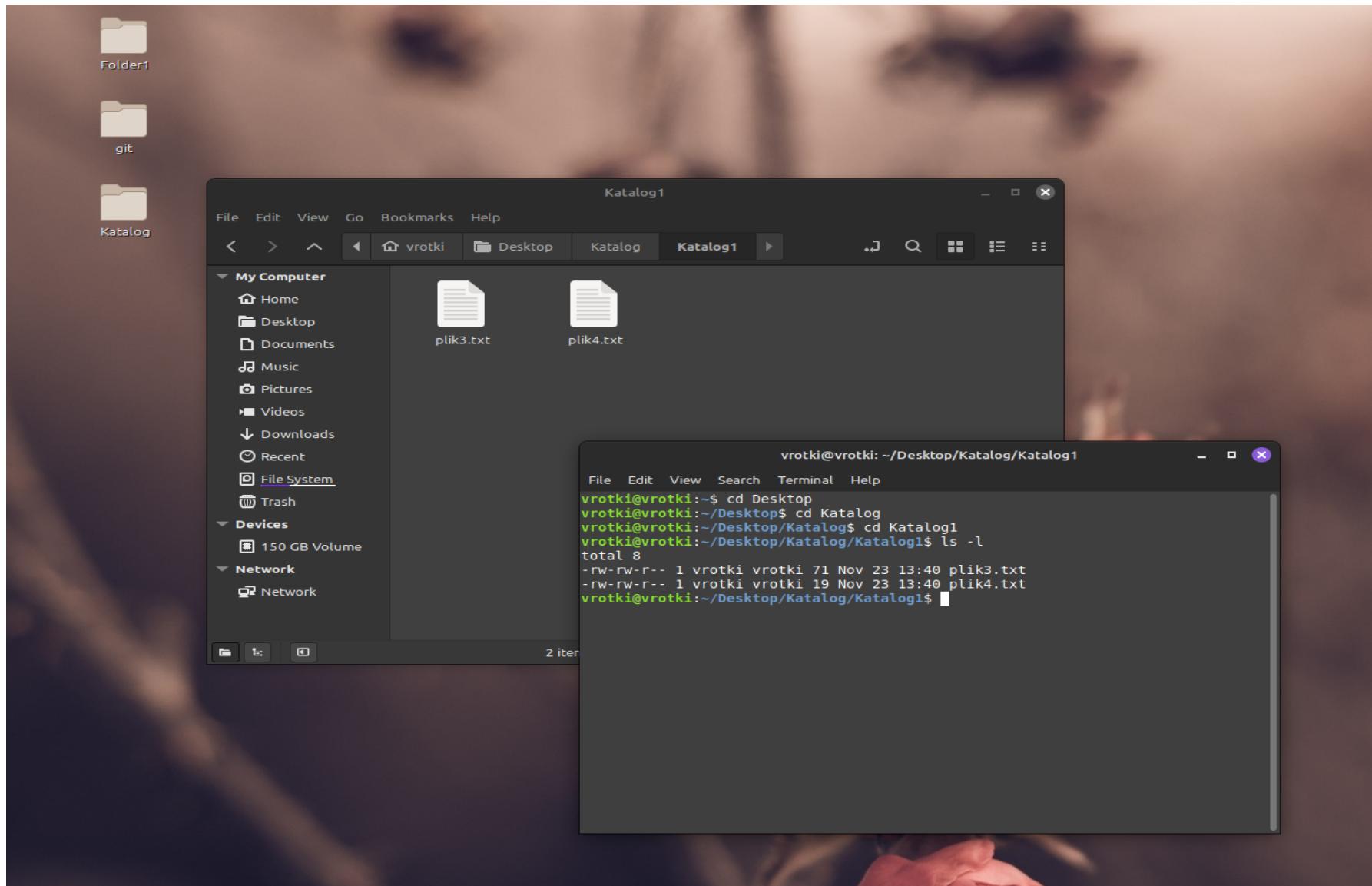
2.



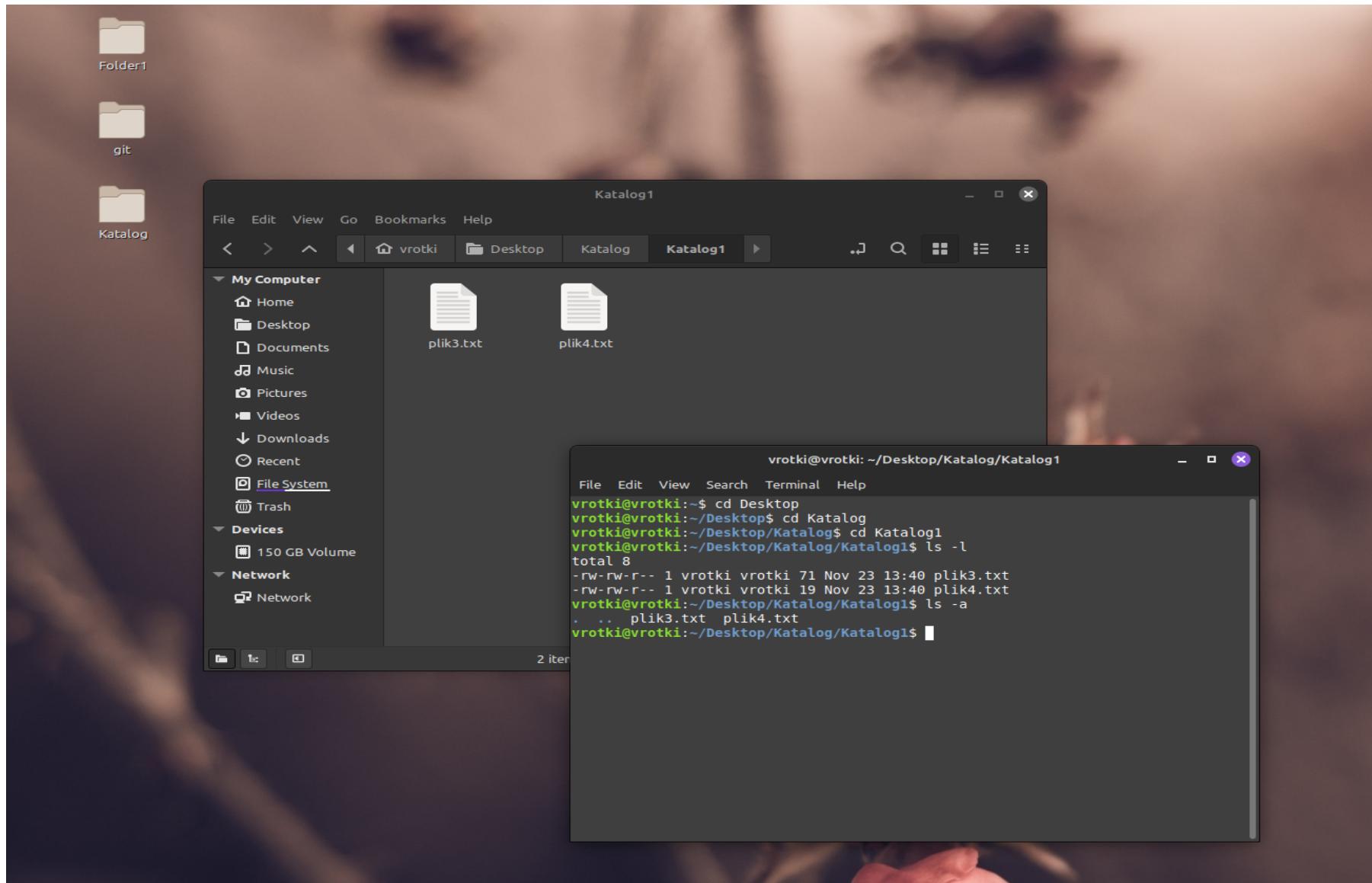
3.



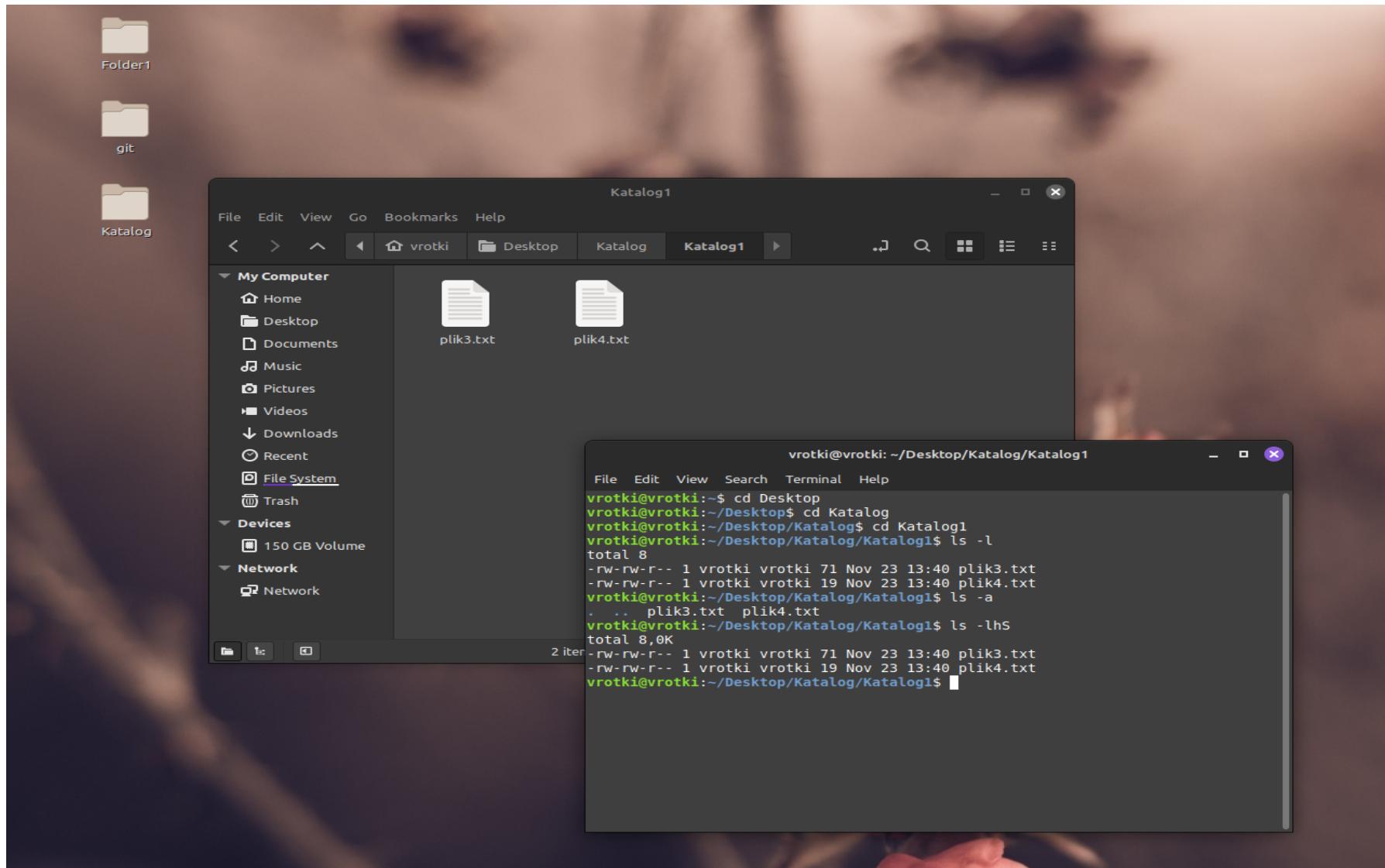
4a.



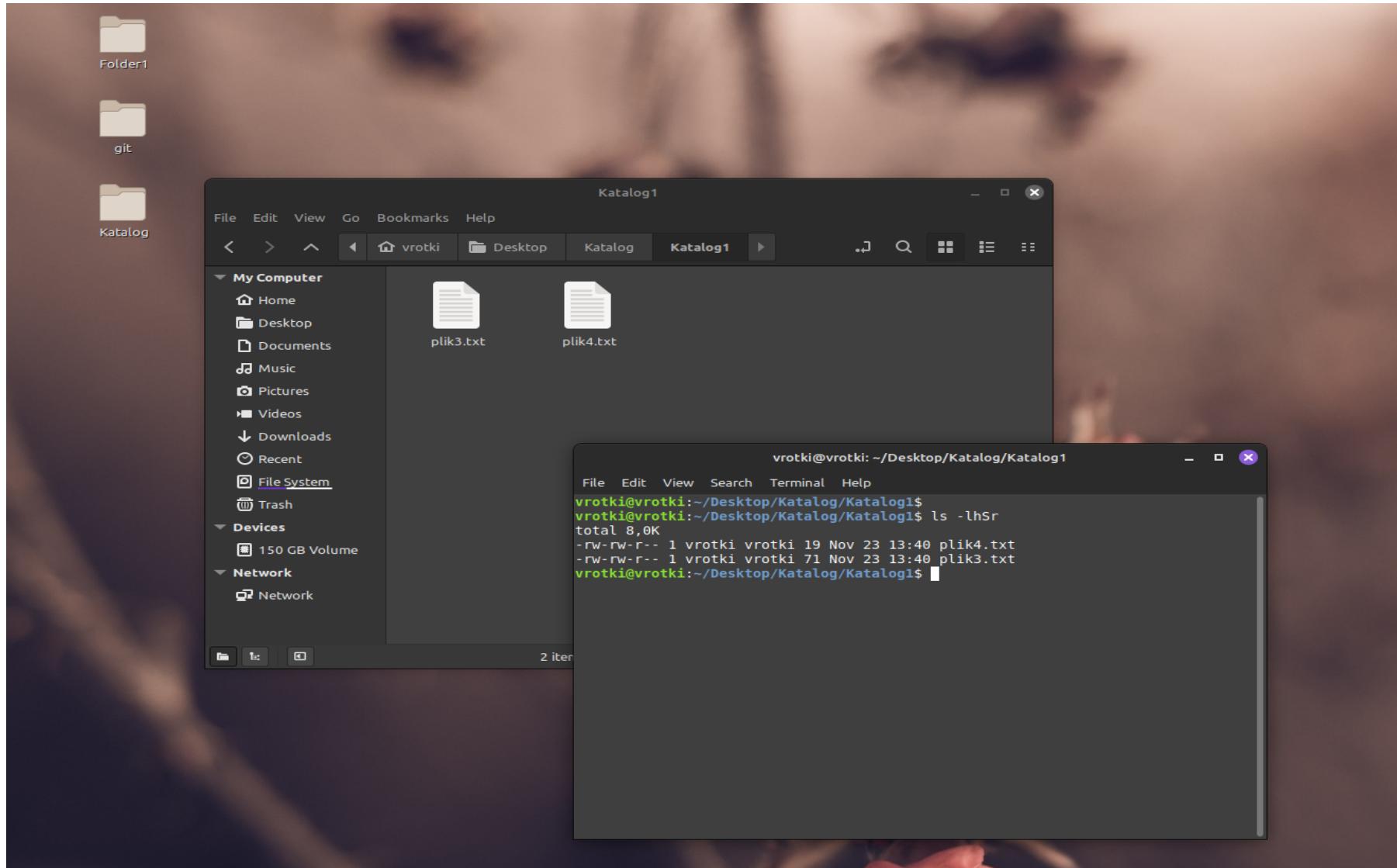
4b.



5a.



5b.



Część 2. Git

1. Zainstaluj GIT na swoim komputerze.
2. Utwórz oddzielną folder na swoim komputerze, z którego będziesz otwierać konsolę GIT.
3. Otwórz konsolę GIT w utworzonym folderze.
4. Skonfiguruj GIT: zmień globalne ustawienia (dodaj swoje imię i adres e-mail).
5. Utwórz nowy projekt z repozytorium na GitHub.
6. Sklonuj projekt na swój komputer.
7. Przejdź do projektu za pomocą komendy cd.
8. Utwórz plik .gitignore.
9. Utwórz plik text_ignore.txt;
10. Dodaj go do pliku .gitignore;
11. Upewnij się, że plik .gitignore nie jest widoczny, wykonując polecenie git status.
12. Utwórz pusty plik readme.txt.
13. Wykonaj commit, aby zatwierdzić swoje zmiany.
14. Utwórz gałąź develop i przejdź na nią za pomocą komendy checkout.
15. Utwórz pusty plik test.txt i wykonaj commit.
16. Wprowadź zmiany w pliku test.txt.
17. Sprawdź zmiany za pomocą komendy git status.
18. Wykonaj commit.
19. Połącz gałąź develop z master (merge).
20. Spróbuj sprawdzić historię swojego repozytorium za pomocą komendy git log.

Część 2. Git

Za pomocą komend:

- git config --global user.name „moja nazwa użytkownika”
- git config --global user.email „mój email”

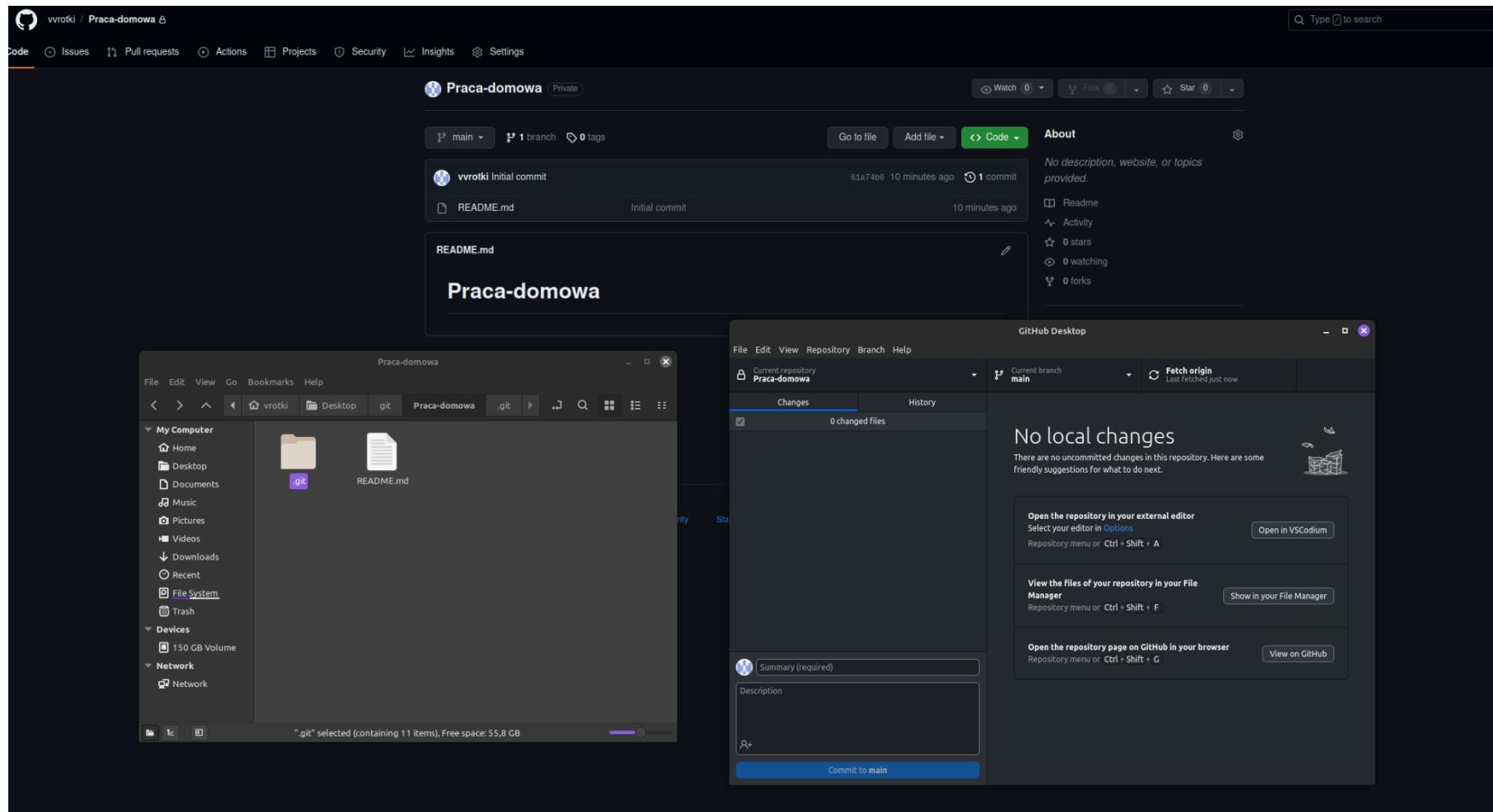
skonfigurowano Git.

Po utworzeniu repozytorium w GitHub, za pomocą komendy:

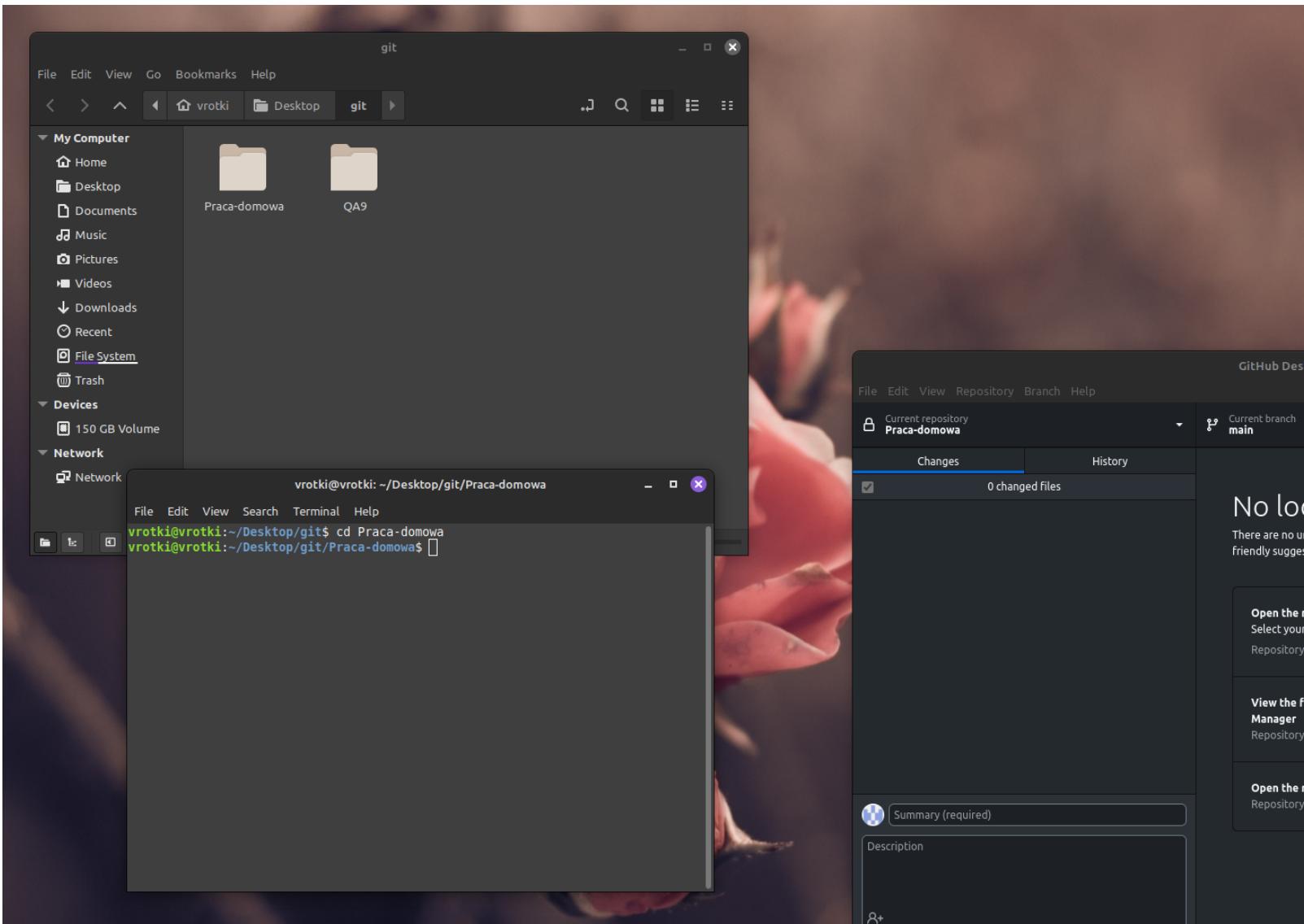
- `git clone "link do mojego repozytorium"`

nie udało się uwierzytelnić repozytorium, więc zainstalowano GitHub Consol.

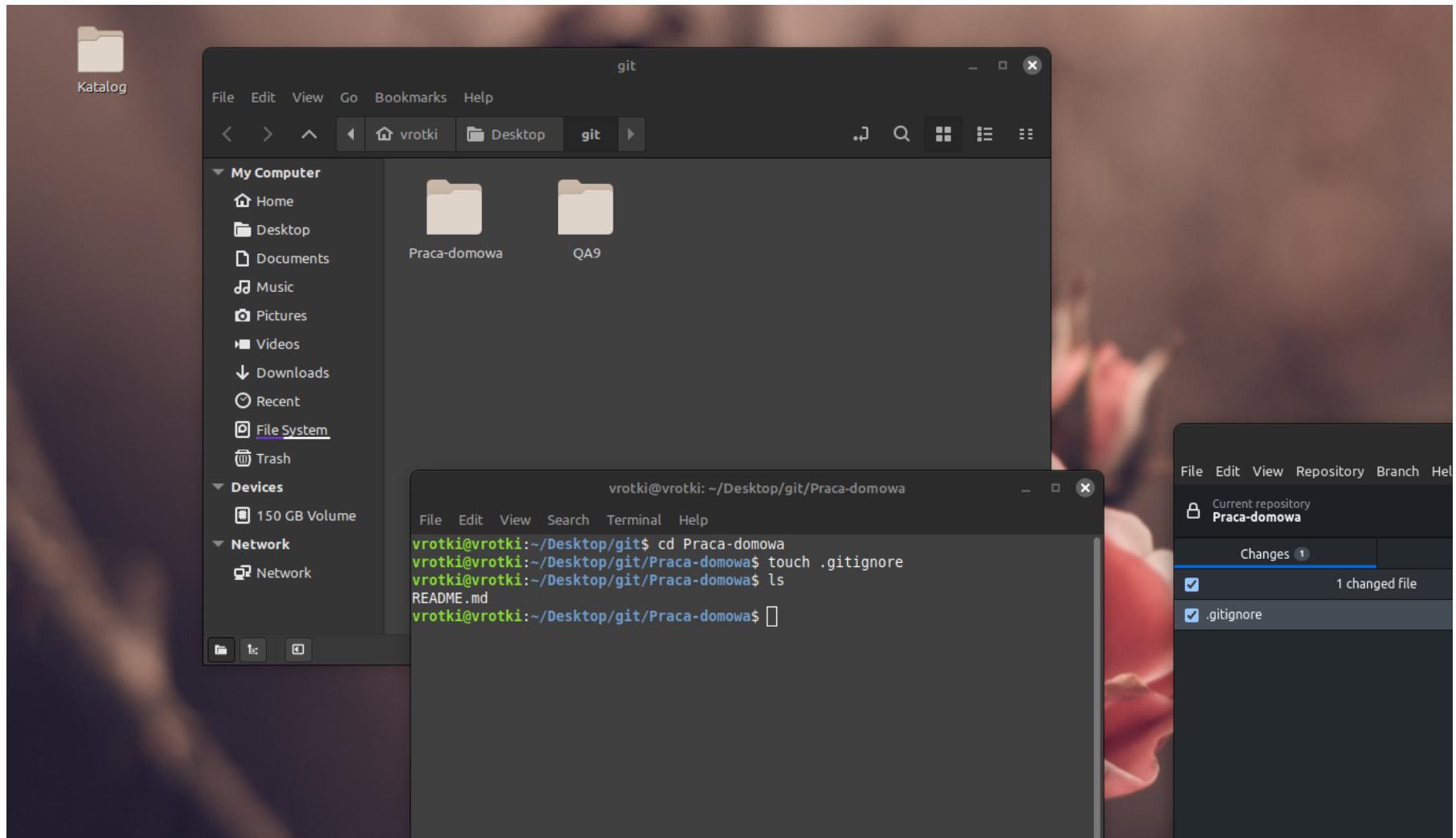
5-6.



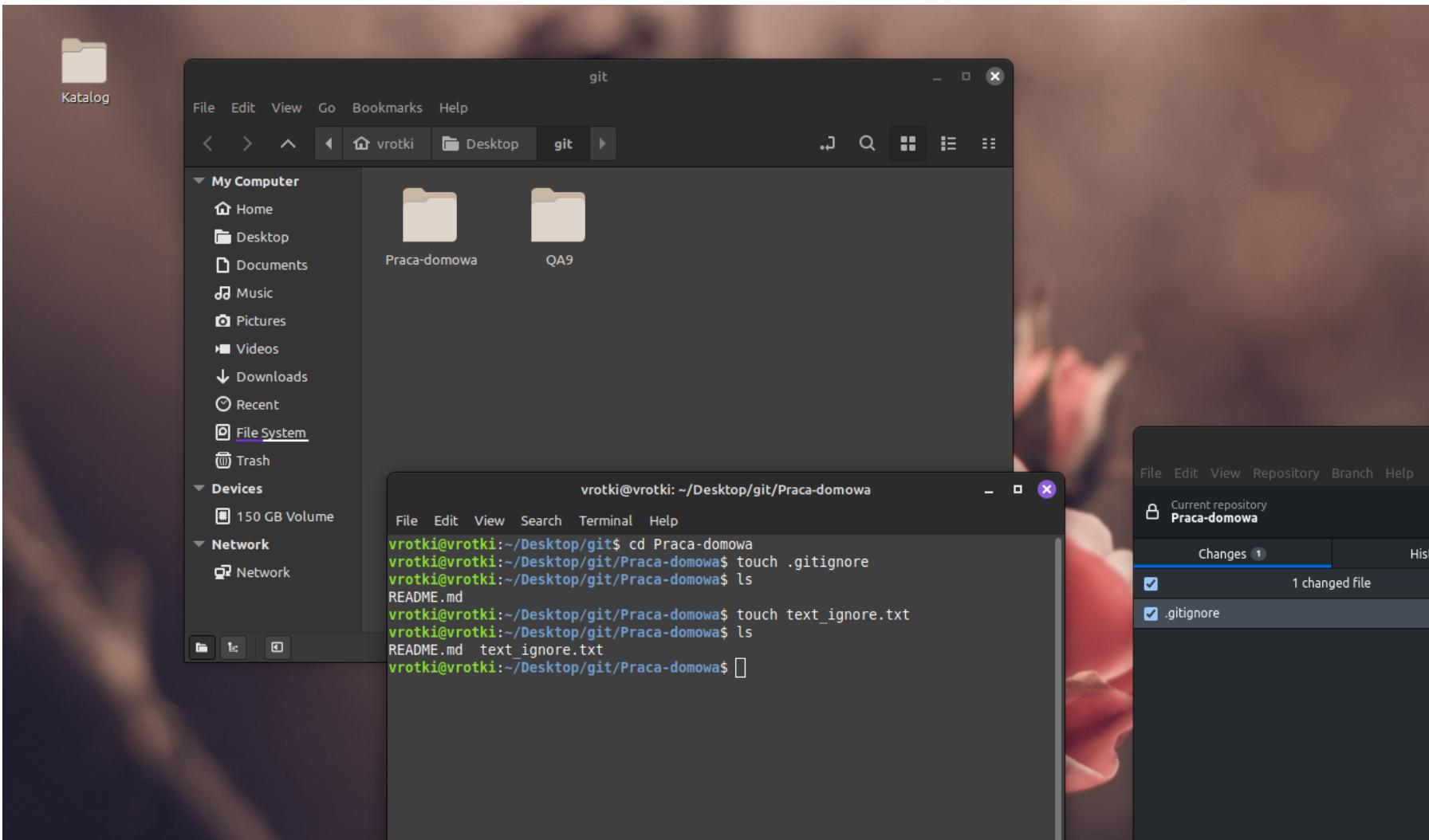
7.



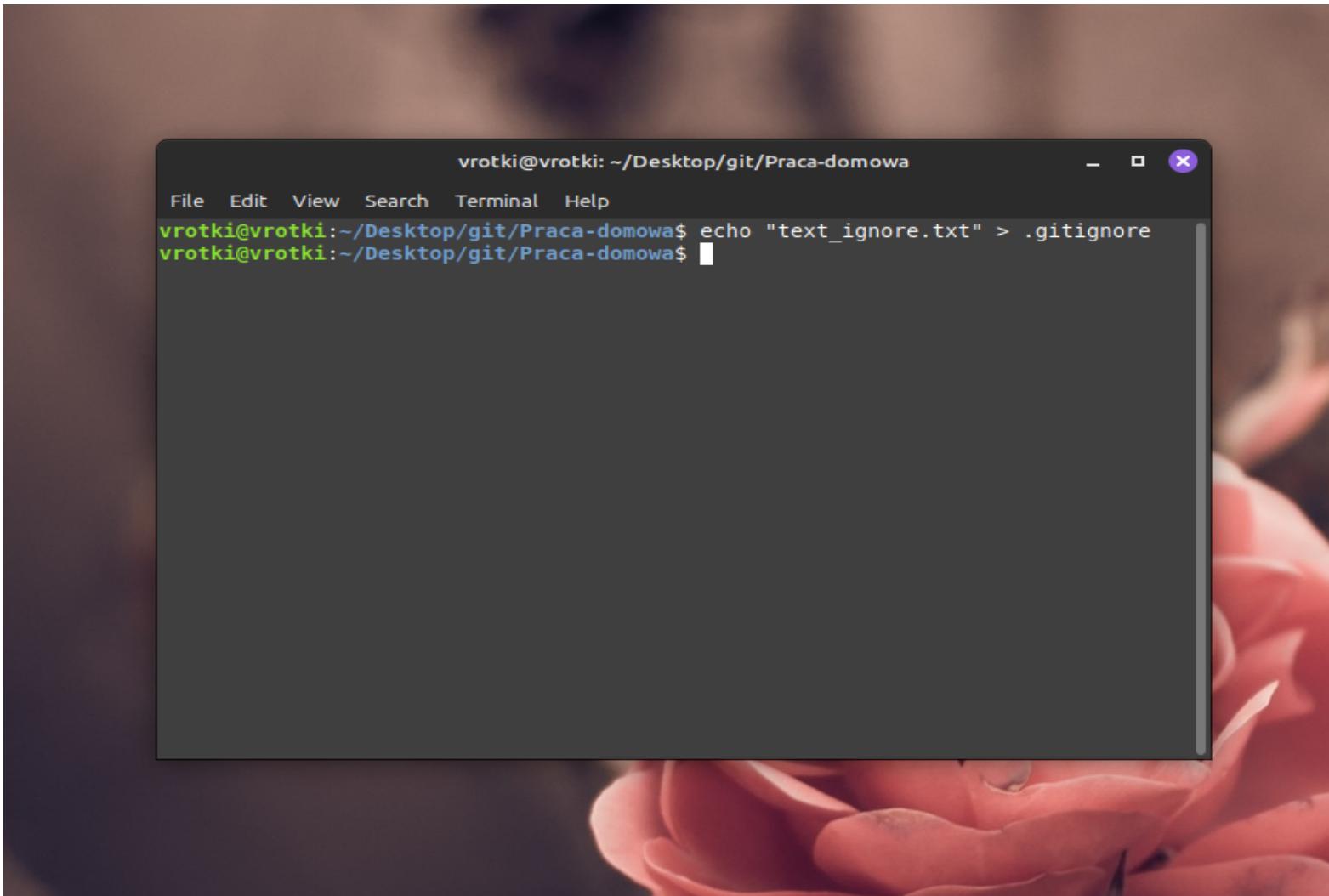
8.



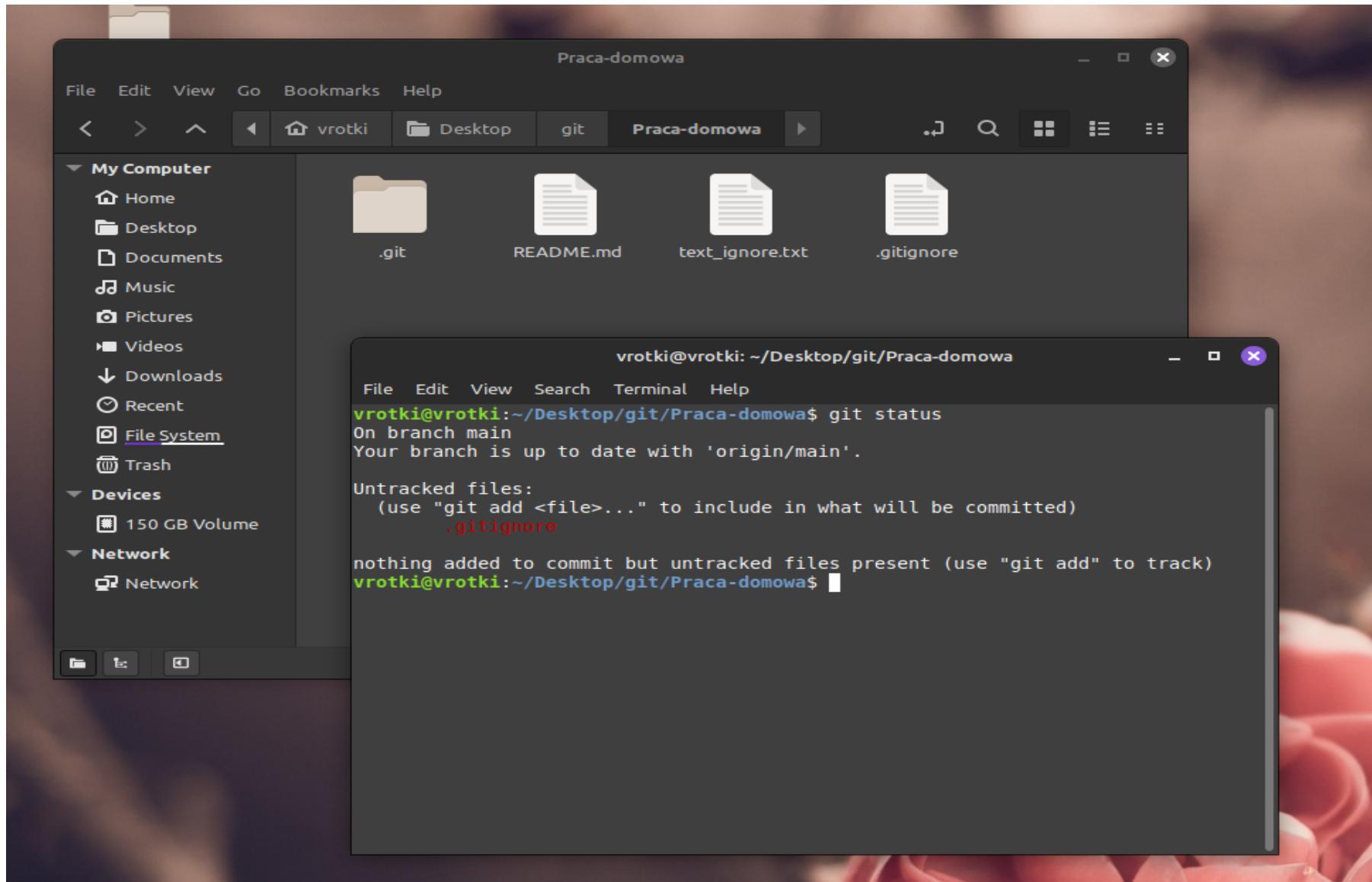
9.



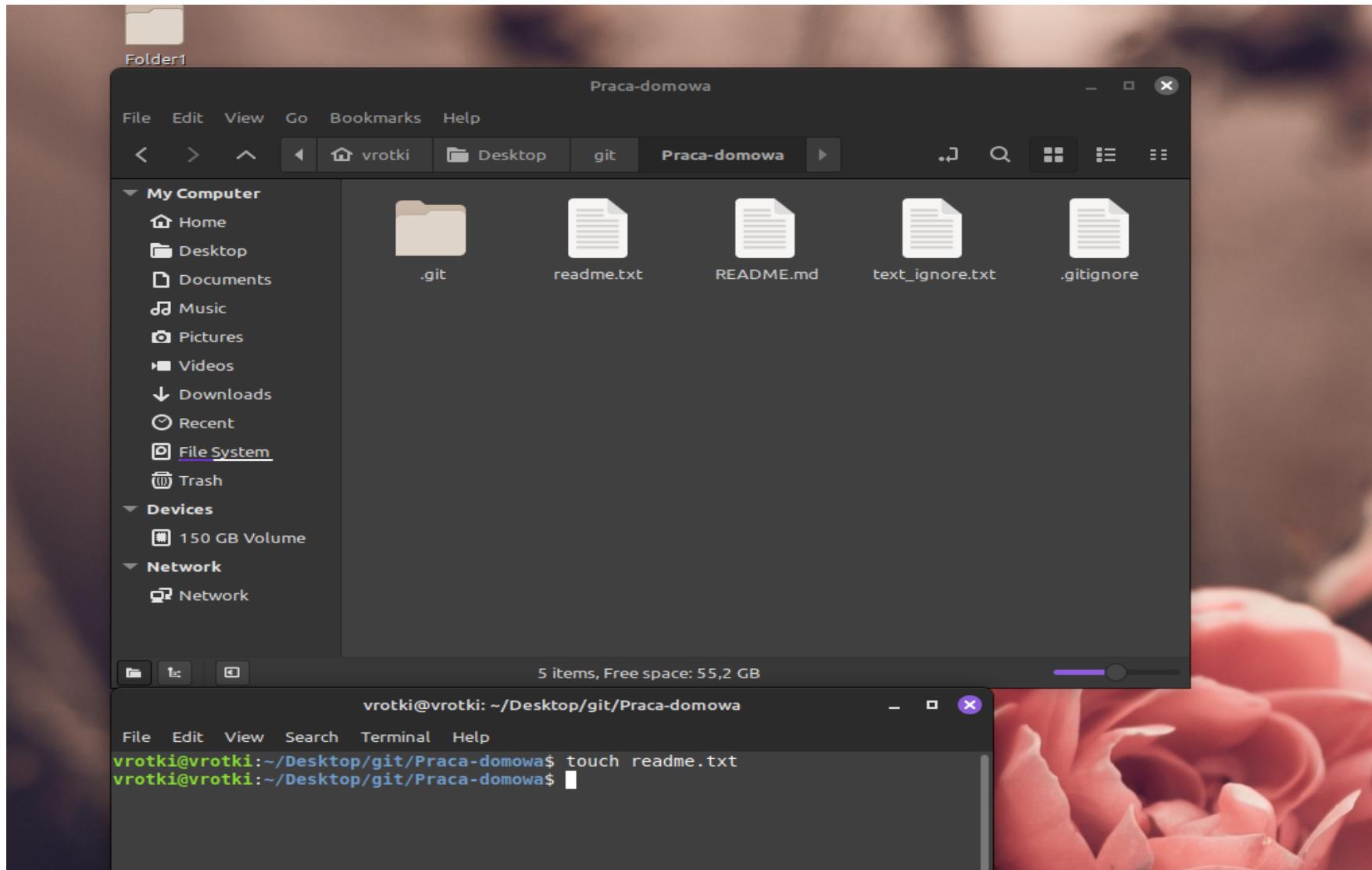
10.



11.



12.



13.

The image shows a Linux desktop environment with a terminal window open in the foreground and a file manager sidebar on the left.

File Manager Sidebar:

- Downloads
- Recent
- File System** (selected)
- Trash

Devices:

- 150 GB Volume

Network:

- Network

Terminal Window:

```
vrotki@vrotki: ~/Desktop/git/Praca-domowa
File Edit View Search Terminal Help
Your branch is up to date with 'origin/main'.

Untracked files:
(use "git add <file>..." to include in what will be committed)
.gitignore
readme.txt

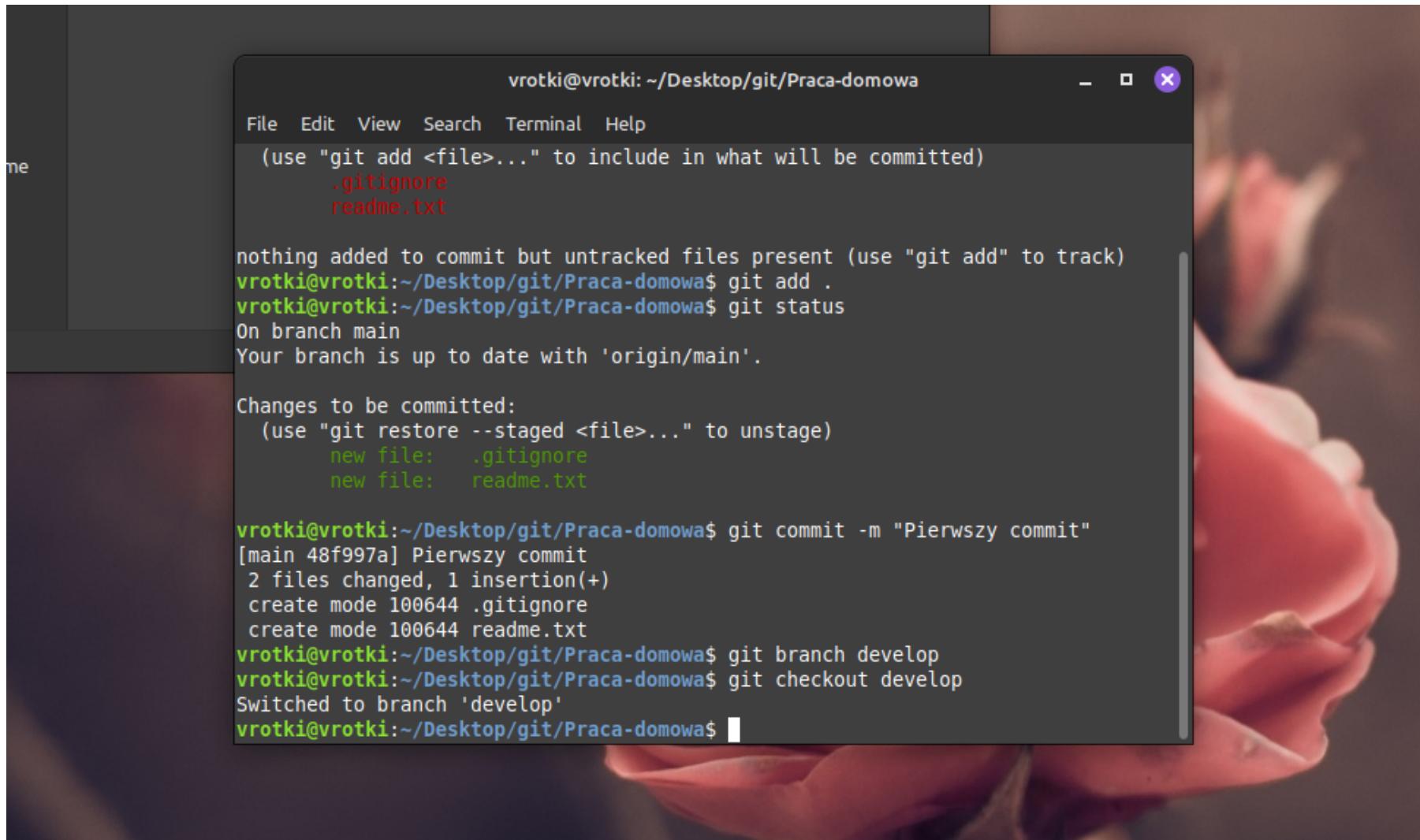
nothing added to commit but untracked files present (use "git add" to track)
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git add .
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
(use "git restore --staged <file>..." to unstage)
  new file:  .gitignore
  new file:  readme.txt

vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit -m "Pierwszy commit"
[main 48f997a] Pierwszy commit
 2 files changed, 1 insertion(+)
 create mode 100644 .gitignore
 create mode 100644 readme.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$
```

The terminal window title is `vrotki@vrotki: ~/Desktop/git/Praca-domowa`. The output shows the user navigating through a directory, adding untracked files to the staging area, and committing them with the message "Pierwszy commit". The commit successfully adds two files: `.gitignore` and `readme.txt`.

14.



```
vrotki@vrotki: ~/Desktop/git/Praca-domowa
File Edit View Search Terminal Help
(use "git add <file>..." to include in what will be committed)
.gitignore
readme.txt

nothing added to commit but untracked files present (use "git add" to track)
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git add .
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
(use "git restore --staged <file>..." to unstage)
  new file:   .gitignore
  new file:   readme.txt

vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit -m "Pierwszy commit"
[main 48f997a] Pierwszy commit
 2 files changed, 1 insertion(+)
 create mode 100644 .gitignore
 create mode 100644 readme.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git branch develop
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git checkout develop
Switched to branch 'develop'
vrotki@vrotki:~/Desktop/git/Praca-domowa$
```

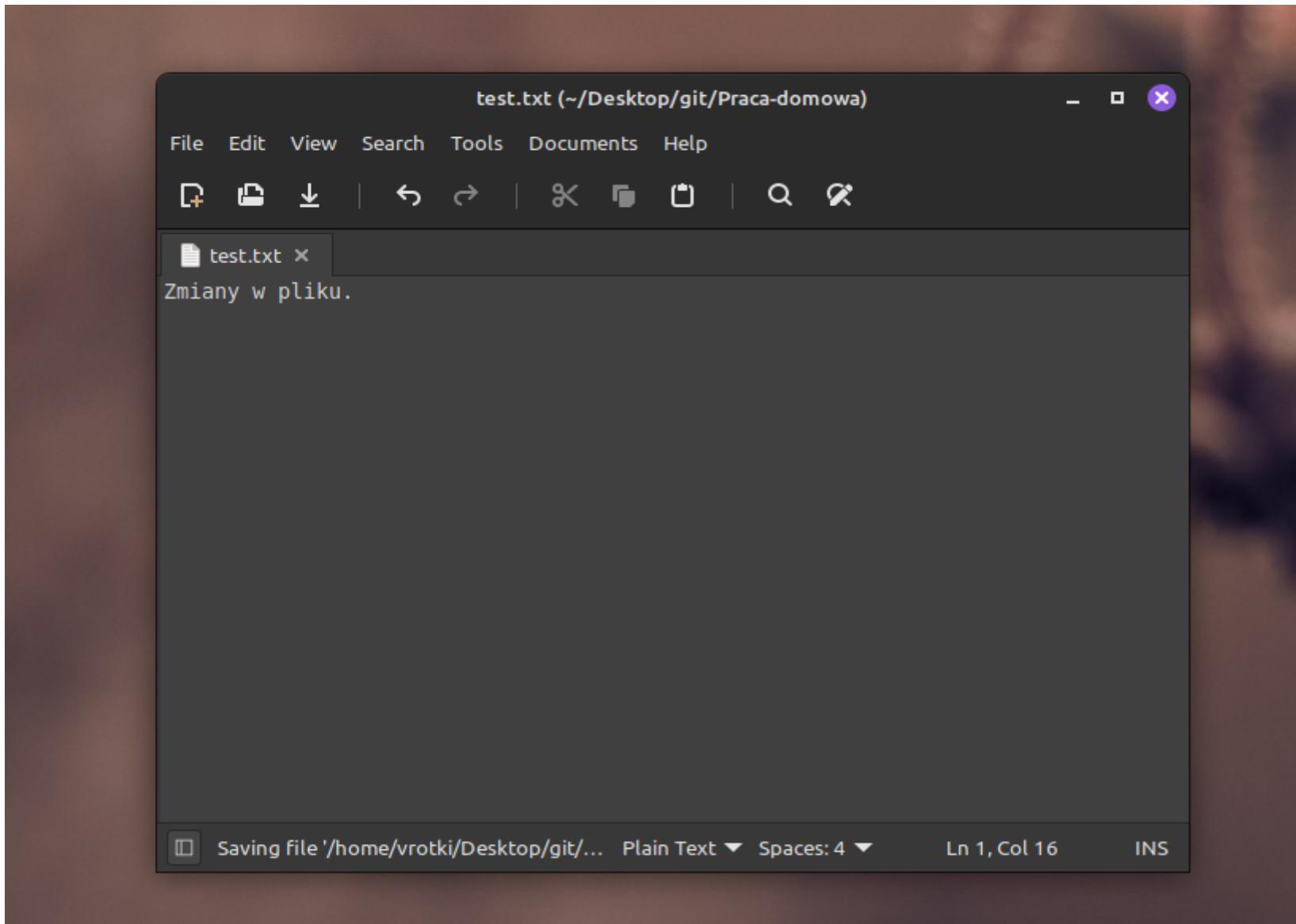
15.

```
.gitignore
vrotki@vrotki: ~/Desktop/git/Praca-domowa
File Edit View Search Terminal Help
create mode 100644 .gitignore
create mode 100644 readme.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git branch develop
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git checkout develop
Switched to branch 'develop'
vrotki@vrotki:~/Desktop/git/Praca-domowa$ touch test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git status
On branch develop
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    test.txt

nothing added to commit but untracked files present (use "git add" to track)
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git add test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git status
On branch develop
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   test.txt

vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit test.txt
Aborting commit due to empty commit message.
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit -m "Drugi commit"
[develop 18263b9] Drugi commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$
```

16.

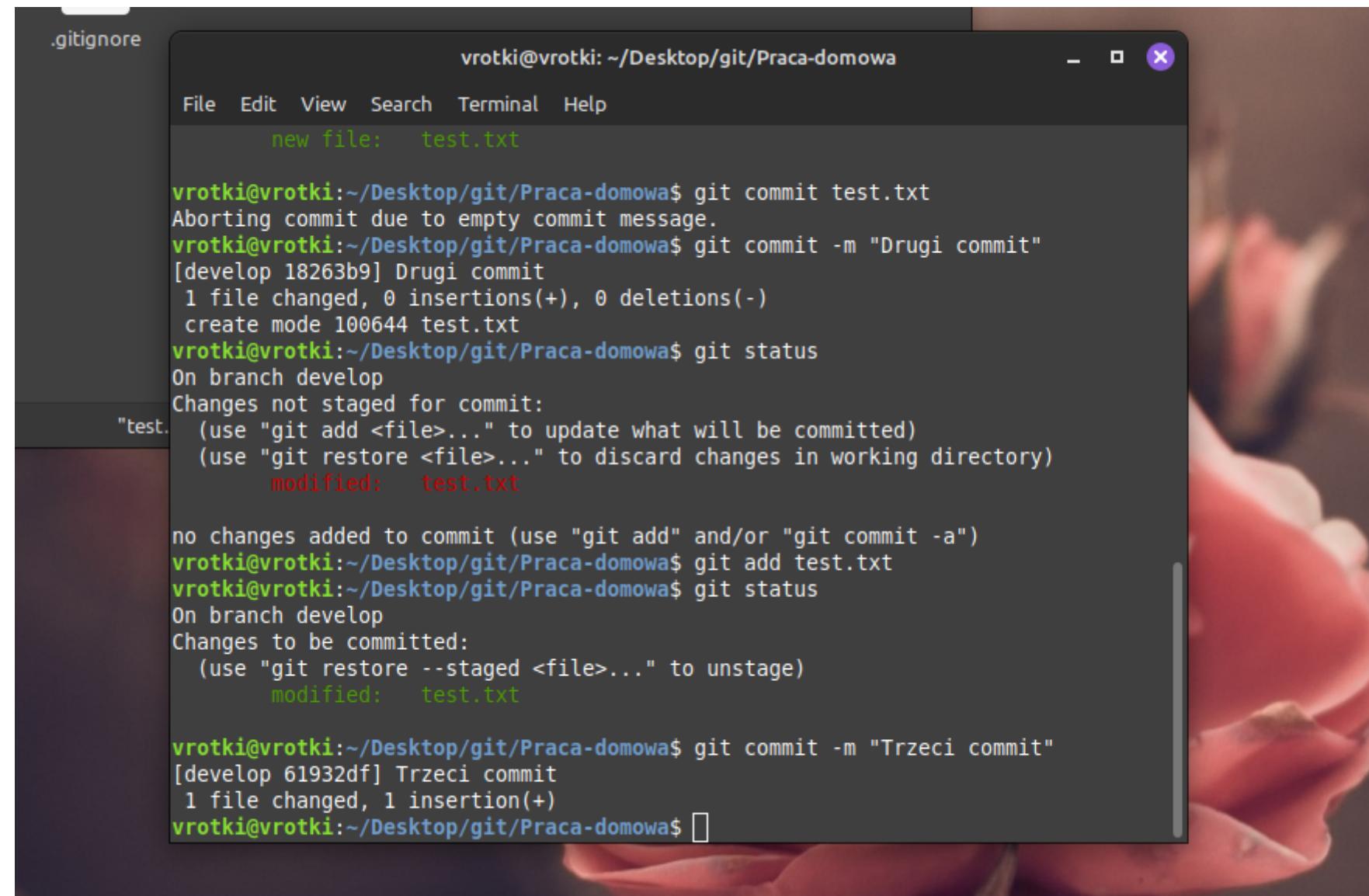


17.

```
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit -m "Drugi commit"
[develop 18263b9] Drugi commit
 1 file changed, 0 insertions(+), 0 deletions(-)
  create mode 100644 test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git status
On branch develop
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   test.txt

no changes added to commit (use "git add" and/or "git commit -a")
vrotki@vrotki:~/Desktop/git/Praca-domowa$
```

18.



.gitignore

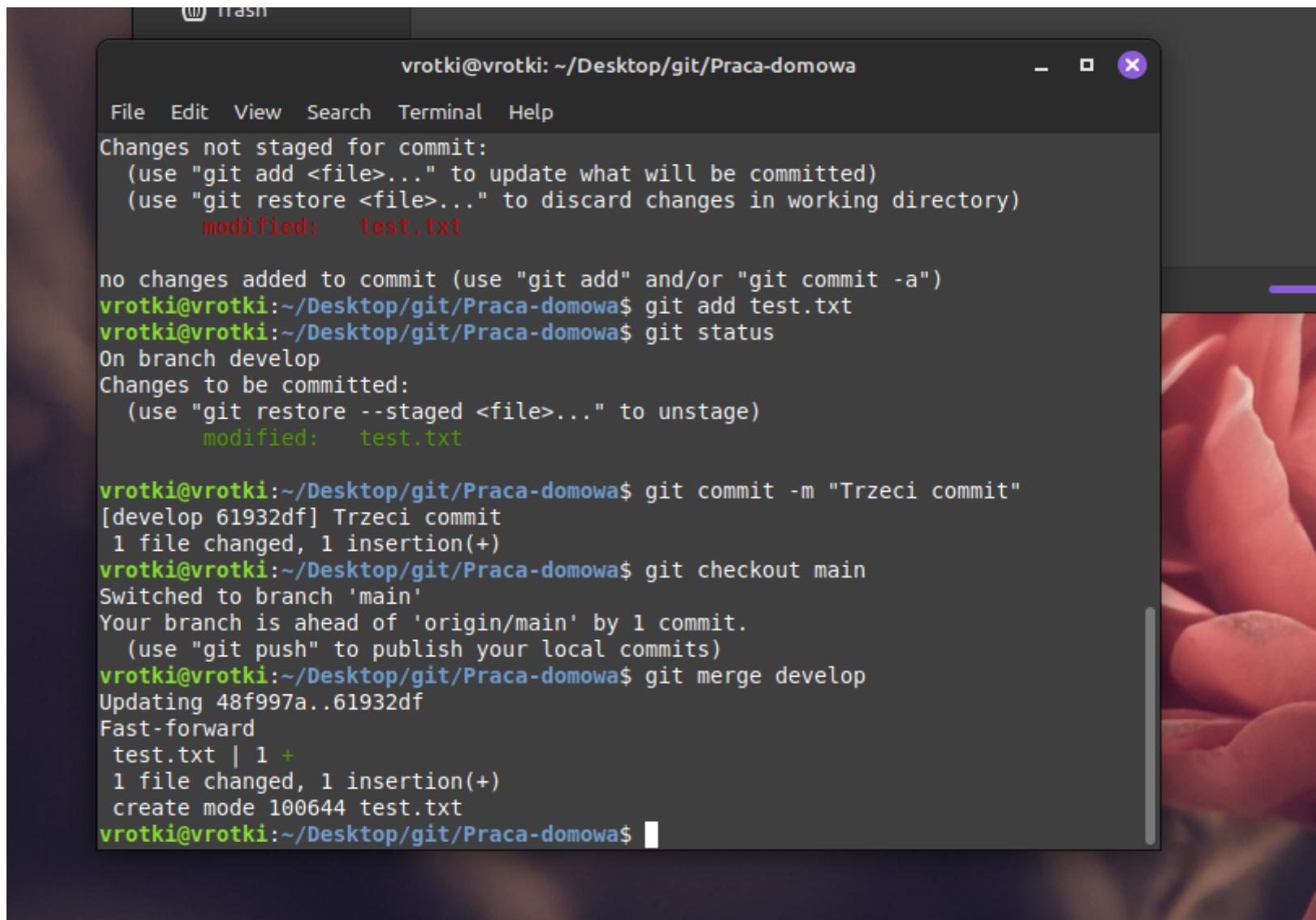
```
vrotki@vrotki: ~/Desktop/git/Praca-domowa
File Edit View Search Terminal Help
new file: test.txt

vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit test.txt
Aborting commit due to empty commit message.
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit -m "Drugi commit"
[develop 18263b9] Drugi commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git status
On branch develop
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified: test.txt

no changes added to commit (use "git add" and/or "git commit -a")
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git add test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git status
On branch develop
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified: test.txt

vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit -m "Trzeci commit"
[develop 61932df] Trzeci commit
 1 file changed, 1 insertion(+)
vrotki@vrotki:~/Desktop/git/Praca-domowa$ 
```

19.



```
vrotki@vrotki: ~/Desktop/git/Praca-domowa
File Edit View Search Terminal Help
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified: test.txt

no changes added to commit (use "git add" and/or "git commit -a")
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git add test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git status
On branch develop
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
modified: test.txt

vrotki@vrotki:~/Desktop/git/Praca-domowa$ git commit -m "Trzeci commit"
[develop 61932df] Trzeci commit
 1 file changed, 1 insertion(+)
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git merge develop
Updating 48f997a..61932df
Fast-forward
 test.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$
```

20.

vrotki@vrotki: ~/Desktop/git/Praca-domowa

```
File Edit View Search Terminal Help
1 file changed, 1 insertion(+)
create mode 100644 test.txt
vrotki@vrotki:~/Desktop/git/Praca-domowa$ git log
commit 61932df1fdd792a0da9cc38da9da5d8a1832371b (HEAD -> main, develop)
Author: vvrotki <asia.asior@gmail.com>
Date:   Sun Nov 26 20:40:47 2023 +0100

Trzeci commit

commit 18263b92d230179bb455331305a754f490329c5c
Author: vvrotki <asia.asior@gmail.com>
Date:   Sun Nov 26 20:36:40 2023 +0100

Drugi commit

commit 48f997a9e3d08520a9592c6d7bc57becff24f8a6
Author: vvrotki <asia.asior@gmail.com>
Date:   Sun Nov 26 20:03:06 2023 +0100

Pierwszy commit

commit 61a74b6daafaaa4749f149cb2ba0c16809bf9e2e (origin/main, origin/HEAD)
Author: vvrotki <143406949+vvrotki@users.noreply.github.com>
Date:   Sat Nov 25 20:37:17 2023 +0100

Initial commit
vrotki@vrotki:~/Desktop/git/Praca-domowa$
```