

Лабораторная работа № 8

Операционные системы

Голованова Мария Константиновна

Содержание

1	Цель работы	5
2	Задание	6
2.1	Задание 2. Редактирование существующего файла	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	9
4.1	Создание нового файла с использованием vi	9
4.2	Редактирование существующего файла	12
5	Выводы	23
	Список литературы	24

Список иллюстраций

4.1	Создание каталога с именем ~/work/os/lab06	9
4.2	Переход в каталог ~/work/os/lab06	9
4.3	Вызов vi и создание файла hello.sh	9
4.4	Введение текста в файл hello.sh	10
4.5	Командный режим	10
4.6	Переход в режим последней строки	11
4.7	Введение команды для сохранения текста в файле и завершения работы	12
4.8	Разрешение на исполнение файла	12
4.9	Вызов vi на редактирование файла	13
4.10	Установка курсора в конец слова HELLO второй строки	13
4.11	Замена слова HELLO на HELLO и возвращение в командный режим	14
4.12	Установка курсора на четвертую строку	15
4.13	Удаление слова LOCAL	16
4.14	Введение слова local	17
4.15	Вставка после последней строки текста echo \$HELLO	18
4.16	Переход в командный режим	19
4.17	Удаление последней строки	20
4.18	Отмена последней команды	21
4.19	Переход в режим последней строки, запись произведённых изменений и выход из vi	22

Список таблиц

1 Цель работы

Познакомиться с операционной системой Linux. Получить практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.

2 Задание

##Задание 1. Создание нового файла с использованием vi

1. Создайте каталог с именем ~/work/os/lab06.
2. Перейдите во вновь созданный каталог.
3. Вызовите vi и создайте файл hello.sh
4. Нажмите клавишу i и вводите следующий текст:
 - #!/bin/bash
 - HELLO=Hello
 - function hello {
 - LOCAL HELLO=World
 - echo \$HELLO
 - }
 - echo \$HELLO
 - hello
5. Нажмите клавишу Esc для перехода в командный режим после завершения ввода текста.
6. Нажмите : для перехода в режим последней строки и внизу вашего экрана появится приглашение в виде двоеточия.
7. Нажмите w (записать) и q (выйти), а затем нажмите клавишу Enter для сохранения вашего текста и завершения работы.
8. Сделайте файл исполняемым

2.1 Задание 2. Редактирование существующего файла

1. Вызовите `vi` на редактирование файла
2. Установите курсор в конец слова `HELL` второй строки.
3. Перейдите в режим вставки и замените на `HELLO`. Нажмите `Esc` для возврата в командный режим.
4. Установите курсор на четвертую строку и сотрите слово `LOCAL`.
5. Перейдите в режим вставки и наберите следующий текст: `local`, нажмите `Esc` для возврата в командный режим.
6. Установите курсор на последней строке файла. Вставьте после неё строку, содержащую следующий текст: `echo $HELLO`.
7. Нажмите `Esc` для перехода в командный режим.
8. Удалите последнюю строку.
9. Введите команду отмены изменений `u` для отмены последней команды.
10. Введите символ `:` для перехода в режим последней строки. Запишите произведённые изменения и выйдите из `vi`.

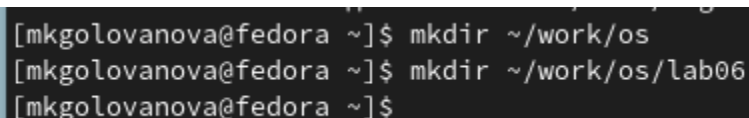
3 Теоретическое введение

В большинстве дистрибутивов Linux в качестве текстового редактора по умолчанию устанавливается интерактивный экранный редактор vi (Visual display editor). Он имеет три режима работы: - командный режим — предназначен для ввода команд редактирования и навигации по редактируемому файлу; для перехода в данный режим используется клавиша **Esc** ; - режим вставки — предназначен для ввода содержания редактируемого файла; - режим последней (или командной) строки — используется для записи изменений в файл и выхода из редактора. Для вызова редактора vi необходимо указать команду vi и имя редактируемого файла: vi . При этом в случае отсутствия файла с указанным именем будет создан такой файл. Для выхода из редактора vi необходимо перейти в режим последней строки: находясь в командном режиме, нажать **Shift-** (:), затем: - набрать символы **wq**, если перед выходом из редактора требуется записать изменения в файл; - набрать символ **q** (или **q!**), если требуется выйти из редактора без сохранения.

4 Выполнение лабораторной работы

4.1 Создание нового файла с использованием vi

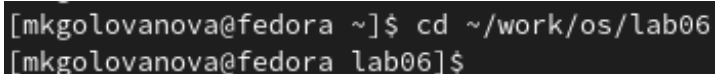
1. Я создала каталог с именем ~/work/os/lab06 (рис. 4.1).

A terminal window showing the execution of two 'mkdir' commands. The first command creates the directory ~/work/os, and the second command creates the subdirectory ~/work/os/lab06. The prompt changes from ~ to lab06 after the second command.

```
[mkgolovanova@fedora ~]$ mkdir ~/work/os  
[mkgolovanova@fedora ~]$ mkdir ~/work/os/lab06  
[mkgolovanova@fedora ~]$
```

Рис. 4.1: Создание каталога с именем ~/work/os/lab06

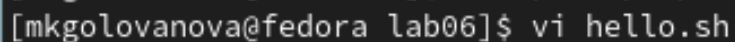
2. Я перешла во вновь созданный каталог (рис. 4.2).

A terminal window showing the execution of a 'cd' command to move into the directory ~/work/os/lab06. The prompt changes from ~ to lab06.

```
[mkgolovanova@fedora ~]$ cd ~/work/os/lab06  
[mkgolovanova@fedora lab06]$
```

Рис. 4.2: Переход в каталог ~/work/os/lab06

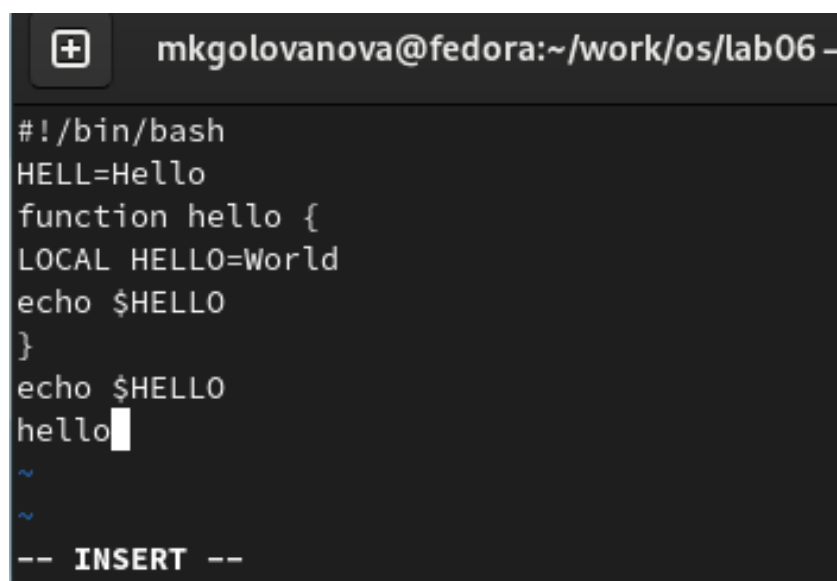
3. Я вызвала vi и создала файл hello.sh (рис. 4.3).

A terminal window showing the execution of the 'vi' command to create a new file named hello.sh in the current directory (lab06).

```
[mkgolovanova@fedora lab06]$ vi hello.sh
```

Рис. 4.3: Вызов vi и создание файла hello.sh

4. Я нажала клавишу i и ввела текст (рис. 4.4).

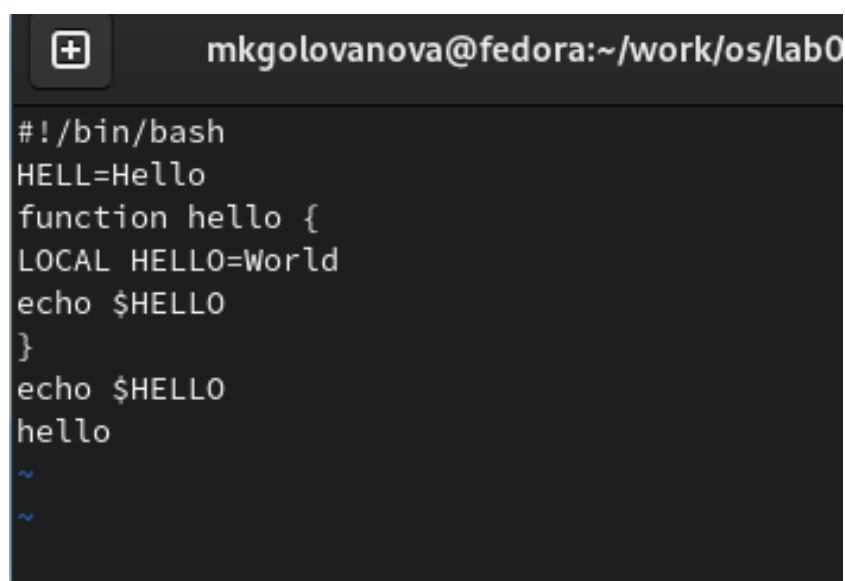
A terminal window with a dark background. The title bar shows a plus icon and the text 'mkgolovanova@fedora:~/work/os/lab06'. The terminal content is as follows:

```
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
```

A white cursor is positioned at the end of the 'hello' line. Below this line, there are two tilde '~' characters and the text '-- INSERT --' at the bottom.

Рис. 4.4: Введение текста в файл hello.sh

5. После завершения ввода текста я нажала клавишу **Esc** и перешла в командный режим (рис. 4.5).

A terminal window with a dark background. The title bar shows a plus icon and the text 'mkgolovanova@fedora:~/work/os/lab06'. The terminal content is as follows:

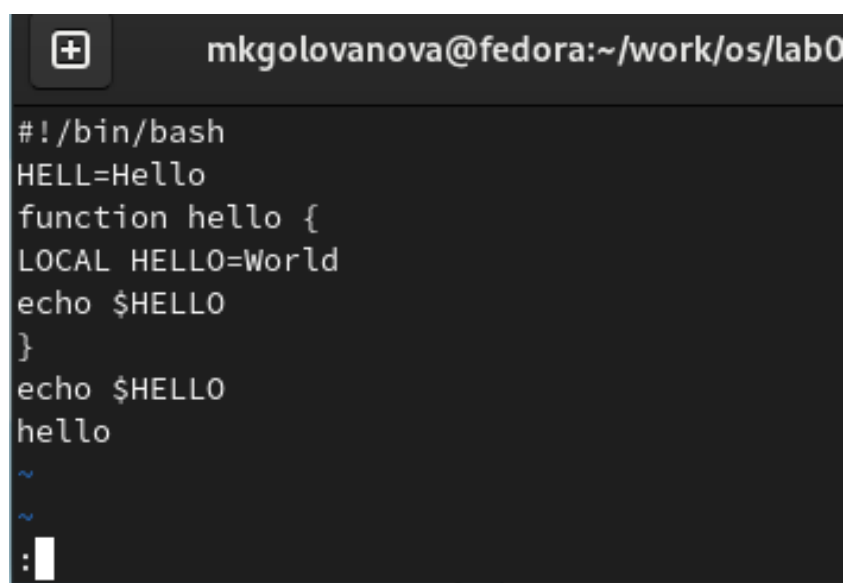
```
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
```

Below the 'hello' line, there are two tilde '~' characters. The text '-- INSERT --' is no longer visible.

Рис. 4.5: Командный режим

6. Я нажала **:** для перехода в режим последней строки, внизу моего экрана

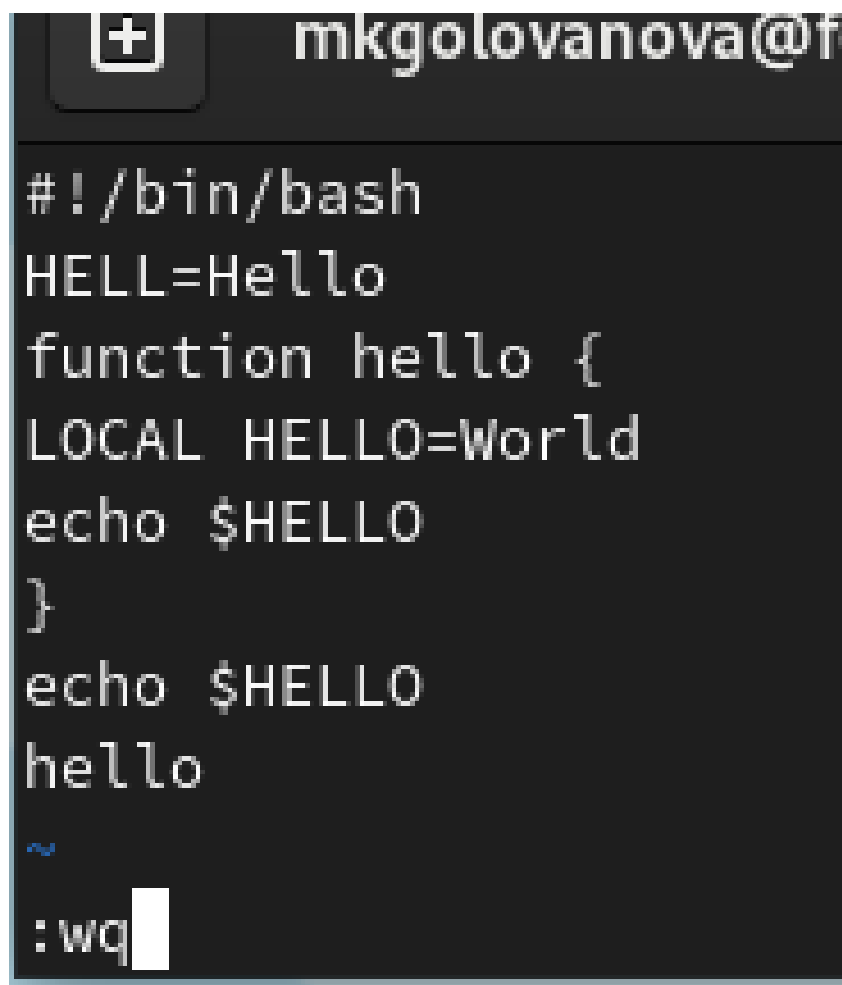
появилось приглашение в виде двоеточия (рис. 4.6).

A terminal window with a dark background. The title bar shows a window icon and the text 'mkgolovanova@fedora:~/work/os/lab0'. The terminal content shows a bash script being executed: '#!/bin/bash', 'HELL=Hello', 'function hello {', 'LOCAL HELLO=World', 'echo \$HELLO', '}', 'echo \$HELLO', and 'hello'. Below the script, there are two tilde '~' characters and a colon ':' followed by a cursor, indicating the prompt for the next command.

```
mkgolovanova@fedora:~/work/os/lab0
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
~
~
:
```

Рис. 4.6: Переход в режим последней строки

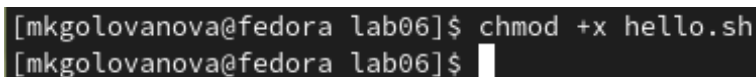
7. Я нажала **w** (записать) и **q** (выйти), а затем нажала клавишу **Enter** для сохранения текста и завершения работы (рис. 4.7).



```
mkgolovanova@f
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
~
:wq
```

Рис. 4.7: Введение команды для сохранения текста в файле и завершения работы

8. Я сделала файл исполняемым (рис. 4.8).

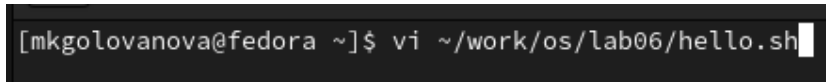


```
[mkgolovanova@fedora lab06]$ chmod +x hello.sh
[mkgolovanova@fedora lab06]$
```

Рис. 4.8: Разрешение на исполнение файла

4.2 Редактирование существующего файла

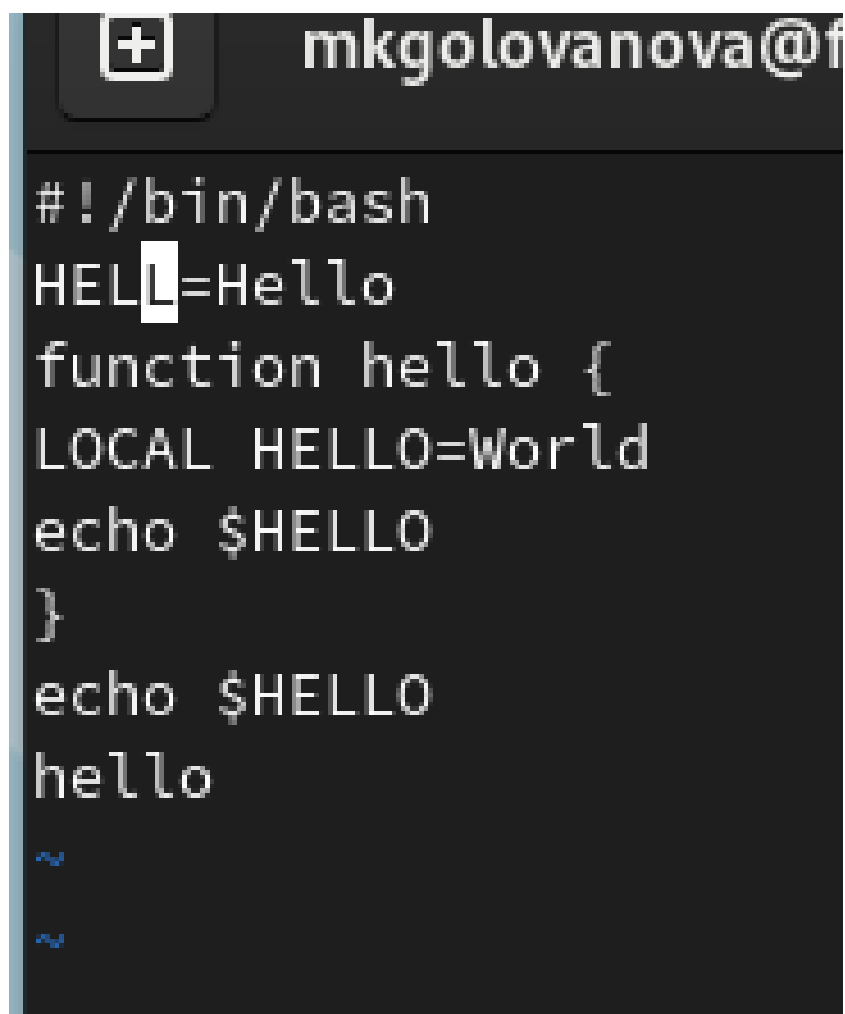
1. Я вызвала vi на редактирование файла (рис. 4.9).



```
[mkgolovanova@fedora ~]$ vi ~/work/os/lab06/hello.sh
```

Рис. 4.9: Вызов vi на редактирование файла

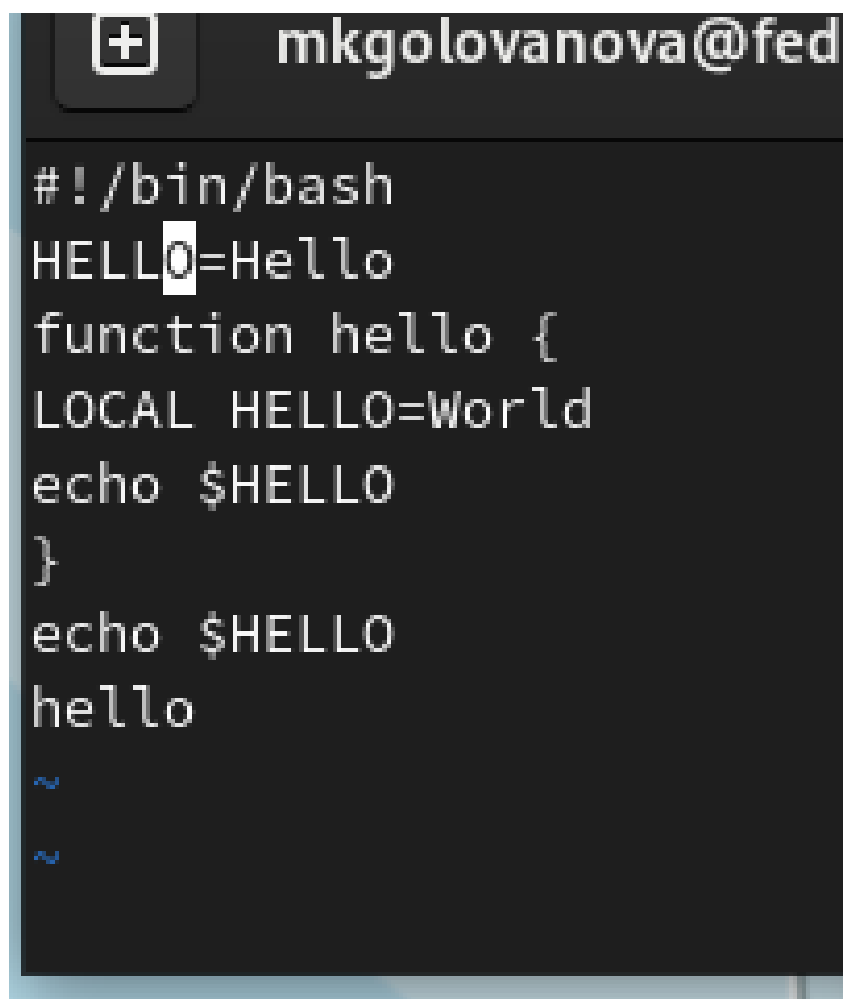
2. Я установила курсор в конец слова HELL во второй строке (рис. 4.10).



```
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
~
~
..
```

Рис. 4.10: Установка курсора в конец слова HELL второй строки

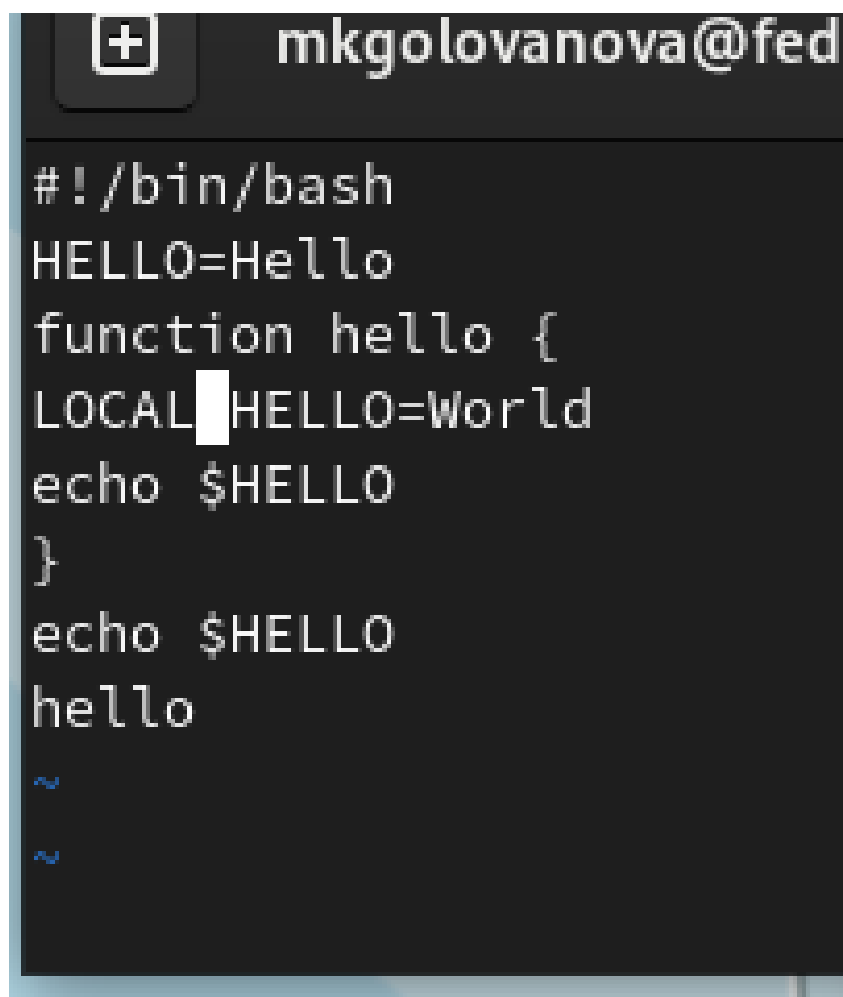
3. Я перешла в режим вставки и заменила слово HELL на HELLO, а затем нажала **Esc** для возврата в командный режим (рис. 4.11).



```
mkgolovanova@fed
#!/bin/bash
HELLO=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
~
~
```

Рис. 4.11: Замена слова HELL на HELLO и возвращение в командный режим

4. Я установила курсор на четвертую строку и стёрла слово LOCAL с помощью команды **dw** (рис. 4.12, рис. 4.13).



A terminal window titled 'mkgolovanova@fed' with a '+' icon in the top-left corner. The terminal displays a bash script with the following content:

```
#!/bin/bash
HELLO=Hello
function hello {
LOCALHELLO=World
echo $HELLO
}
echo $HELLO
hello
```

The cursor is positioned at the end of the fourth line, 'LOCALHELLO=World'. Below the script, there are two tilde characters '~' on separate lines.

Рис. 4.12: Установка курсора на четвертую строку

```
#!/bin/bash
HELLO=Hello
function hello {
HELLO=World
echo $HELLO
}
echo $HELLO
hello
~
~
```

Рис. 4.13: Удаление слова LOCAL

5. Я перешла в режим вставки и набрала слово local (рис. 4.14), затем нажала **Esc** для возврата в командный режим .


```
#!/bin/bash
HELLO=Hello
function hello {
local HELLO=World
echo $HELLO
}
echo $HELLO
hello
~
-- INSERT --
```

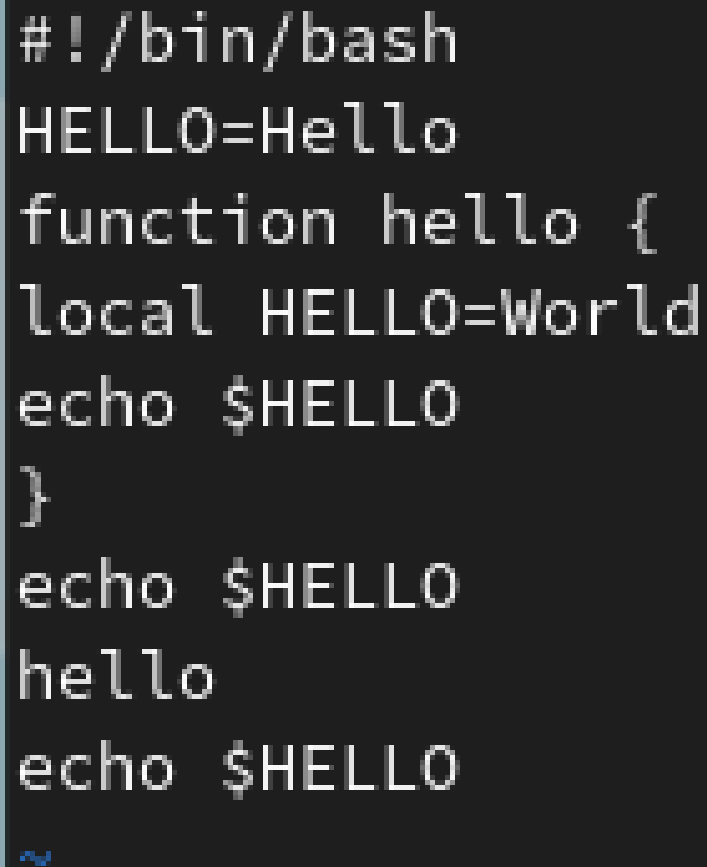
Рис. 4.14: Введение слова local

6. Я установила курсор на последней строке файла и вставила после неё строку, содержащую текст echo \$HELLO (рис. 4.15).

```
#!/bin/bash
HELLO=Hello
function hello {
local HELLO=World
echo $HELLO
}
echo $HELLO
hello
echo $HELLO
~
-- INSERT --
```

Рис. 4.15: Вставка после последней строки текста echo \$HELLO

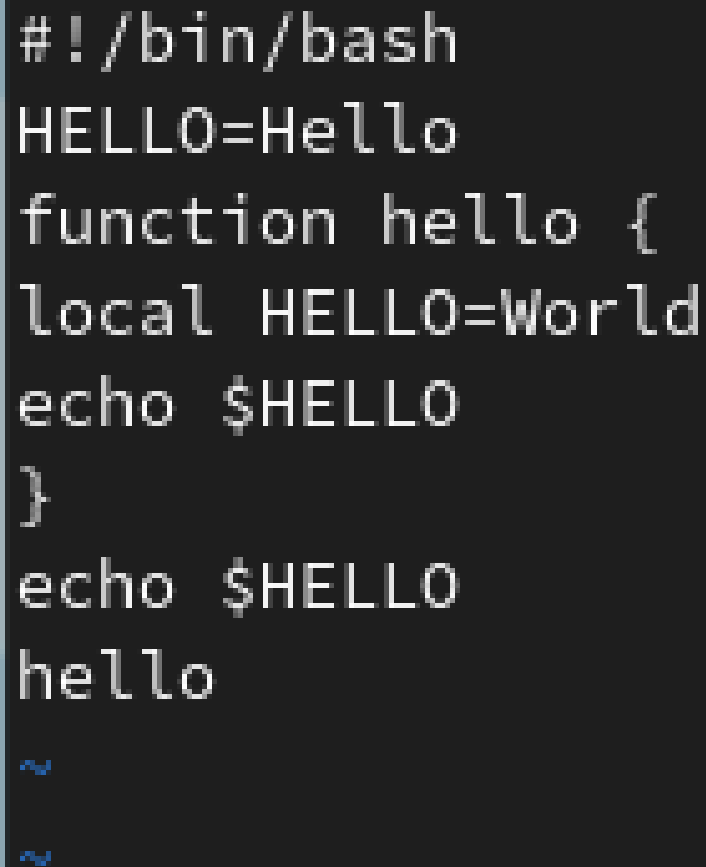
7. Я нажала **Esc** для перехода в командный режим (рис. 4.16).

A terminal window with a dark background and light blue text. The text shows a shell script being executed. The script starts with a shebang, sets a global variable, defines a function, and then calls the function. The function sets a local variable and echoes it. After the function call, the global variable is echoed again. The prompt is a blue tilde character.

```
#!/bin/bash
HELLO=Hello
function hello {
  local HELLO=World
  echo $HELLO
}
echo $HELLO
hello
echo $HELLO
~
```

Рис. 4.16: Переход в командный режим

8. Удалите последнюю строку с помощью команды **dd** (рис. 4.17).

A terminal window with a dark background and light blue text. The text shows a bash script being executed. The script defines a variable HELLO=Hello, a function hello that sets a local variable HELLO=World and echoes \$HELLO, and then echoes \$HELLO. The function is called with the argument 'hello'. The output shows 'Hello' on the first line and 'World' on the second line.

```
#!/bin/bash
HELLO=Hello
function hello {
  local HELLO=World
  echo $HELLO
}
echo $HELLO
hello
~
~
```

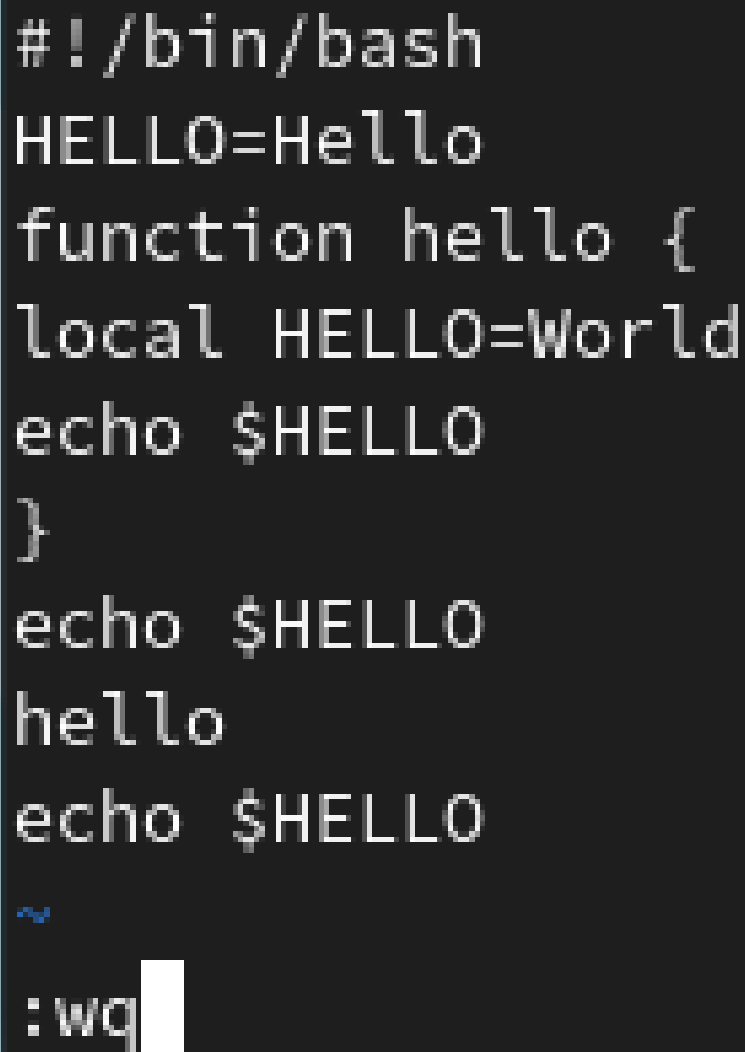
Рис. 4.17: Удаление последней строки

9. Я ввела команду отмены изменений **u** для отмены последней команды (рис. 4.18).

```
#!/bin/bash
HELLO=Hello
function hello {
local HELLO=World
echo $HELLO
}
echo $HELLO
hello
echo $HELLO
1 more line; before #10  11 seconds ago
```

Рис. 4.18: Отмена последней команды

10. Я ввела символ `:` для перехода в режим последней строки, записала произведённые изменения и вышла из `vi` (рис. 4.19).



```
#!/bin/bash
HELLO=Hello
function hello {
local HELLO=World
echo $HELLO
}
echo $HELLO
hello
echo $HELLO
~
:wq
```

Рис. 4.19: Переход в режим последней строки, запись произведённых изменений и выход из vi

5 Выводы

Я получила практические навыки работы с редактором vi.

Список литературы