

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 9

дисциплина: Операционные системы

Студент: Ким Реачна

Группа: НПИбд-02-20

Москва

2021г.

Цель работы:

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

Теоретические сведения:

Редактор vi имеет три режима работы:

- командный режим — предназначен для ввода команд редактирования и навигации по редактируемому файлу
- 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 <имя_файла>` .

При этом в случае отсутствия файла с указанным именем будет создан такой файл.

Переход в командный режим осуществляется нажатием клавиши `Esc` .

Для выхода из редактора vi необходимо перейти в режим последней строки: находясь в командном режиме, нажать Shift-; (по сути символ : — двоеточие), 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.

Основные группы команд редактора:

- Команды управления курсором
- Команды позиционирования
- Команды позиционирования
- Команды перемещения по файлу
- Команды перемещения по словам
- Команды редактирования:
 - Вставка текста
 - Вставка строки
 - Удаление текста
 - Отмена и повтор произведённых изменений
 - Копирование текста в буфер
 - Вставка текста из буфер
 - Замена текста
 - Поиск текста
- Команды редактирования в режиме командной строки:
 - Копирование и перемещение текста
 - Запись в файл и выход из редактора

Выполнение работы:

1. Ознакомиться с теоретическим материалом. [\[1\]](#)
2. Ознакомиться с редактором vi .
3. Выполнить упражнения, используя команды vi .

Задание 1. Создание нового файла с использованием vi

1. Создайте каталог с именем `~/work/os/lab09` :

Рисунок 1: создание каталог `~/work/os/lab09`

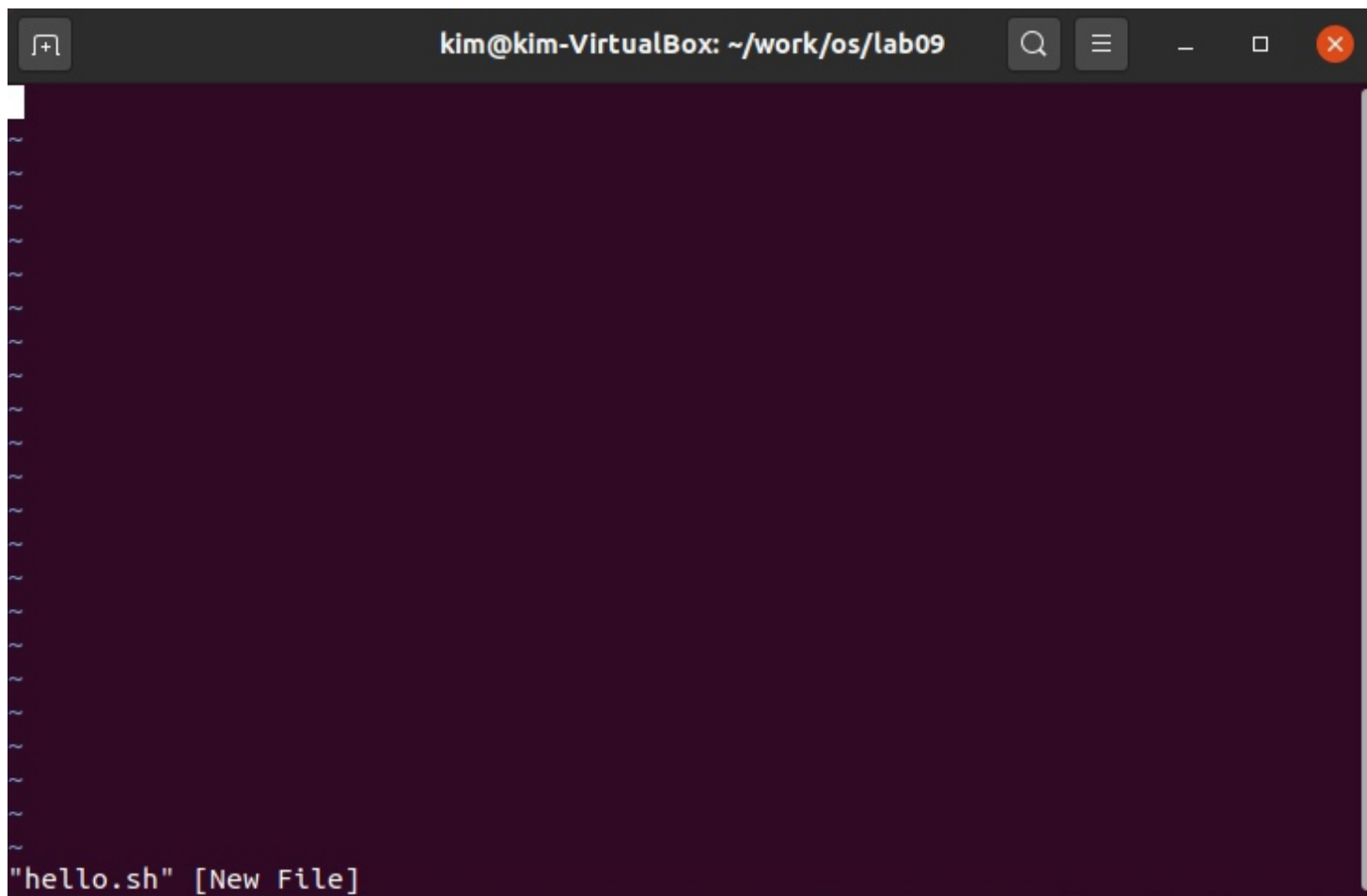
```
kim@kim-VirtualBox:~$ cd work
kim@kim-VirtualBox:~/work$ mkdir os
kim@kim-VirtualBox:~/work$ cd ~/work/os
kim@kim-VirtualBox:~/work/os$ mkdir lab09
kim@kim-VirtualBox:~/work/os$ cd ~/work/os/lab09
kim@kim-VirtualBox:~/work/os/lab09$
```

2. Перейдите во вновь созданный каталог `~/work/os/lab09` (*Рисунок 1*)
3. Вызовите `vi` и создайте файл `hello.sh` : `vi hello.sh` : В работе здесь после того, как мы запустим команду `vi hello.sh` затем мы используем сочетание клавиш не работает для вставки текста, поэтому я установил VIM, чтобы решить проблему с помощью команды `sudo apt-get install vim`

Рисунок 2: вызов `vi` и создание файла `hello.sh`

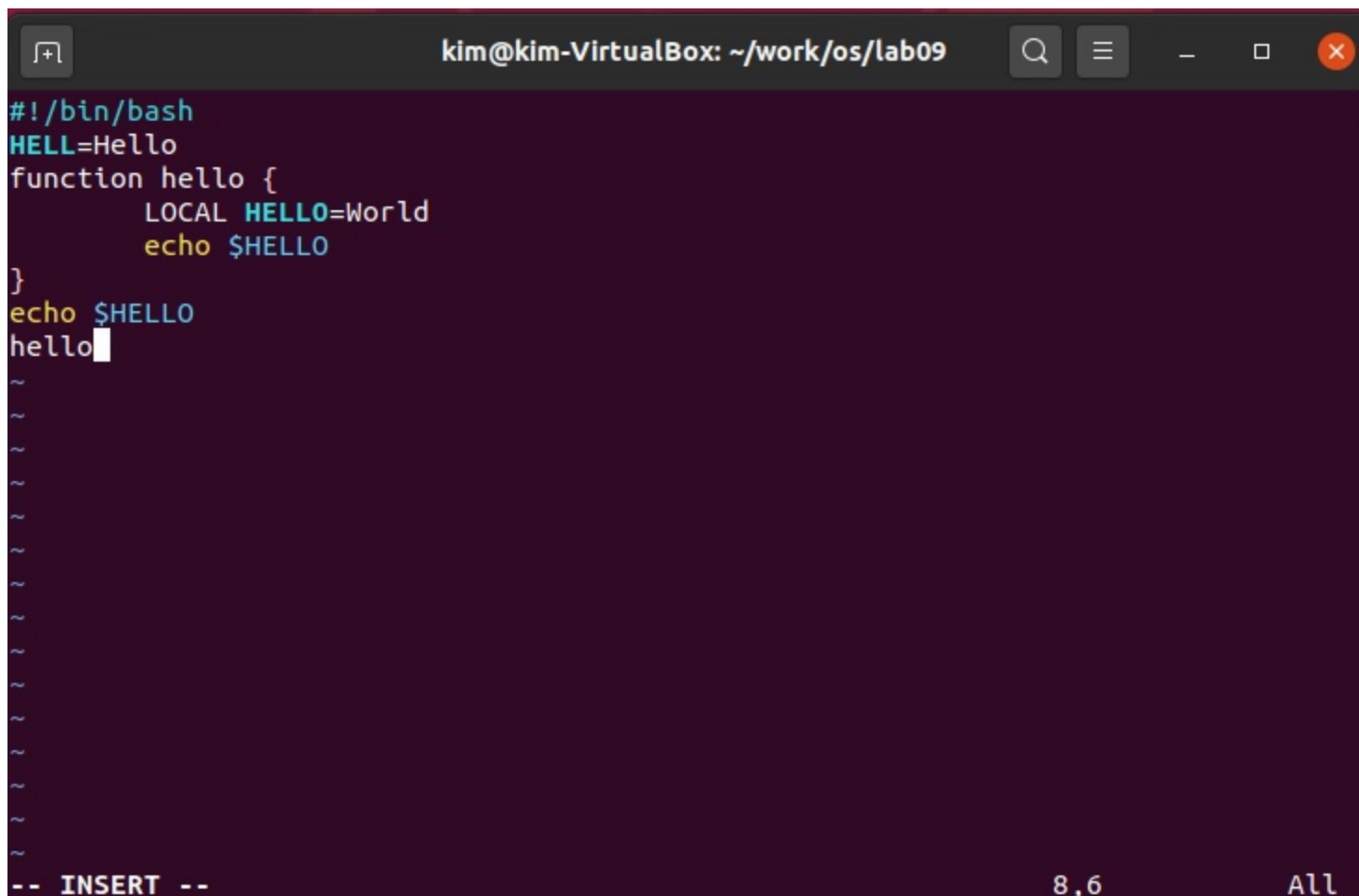
```
kim@kim-VirtualBox:~/work/os/lab09$ vi hello.sh
```

Рисунок 3: Открыть файл `hello.sh`



4. Нажмите клавишу `i` и вводите следующий текст.

Рисунок 4: редактирование файла и ввод текста



```
kim@kim-VirtualBox: ~/work/os/lab09
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
```

5. Нажмите клавишу `Esc` для перехода в командный режим после завершения ввода текста.

Рисунок 5: нажмите клавишу `Esc`

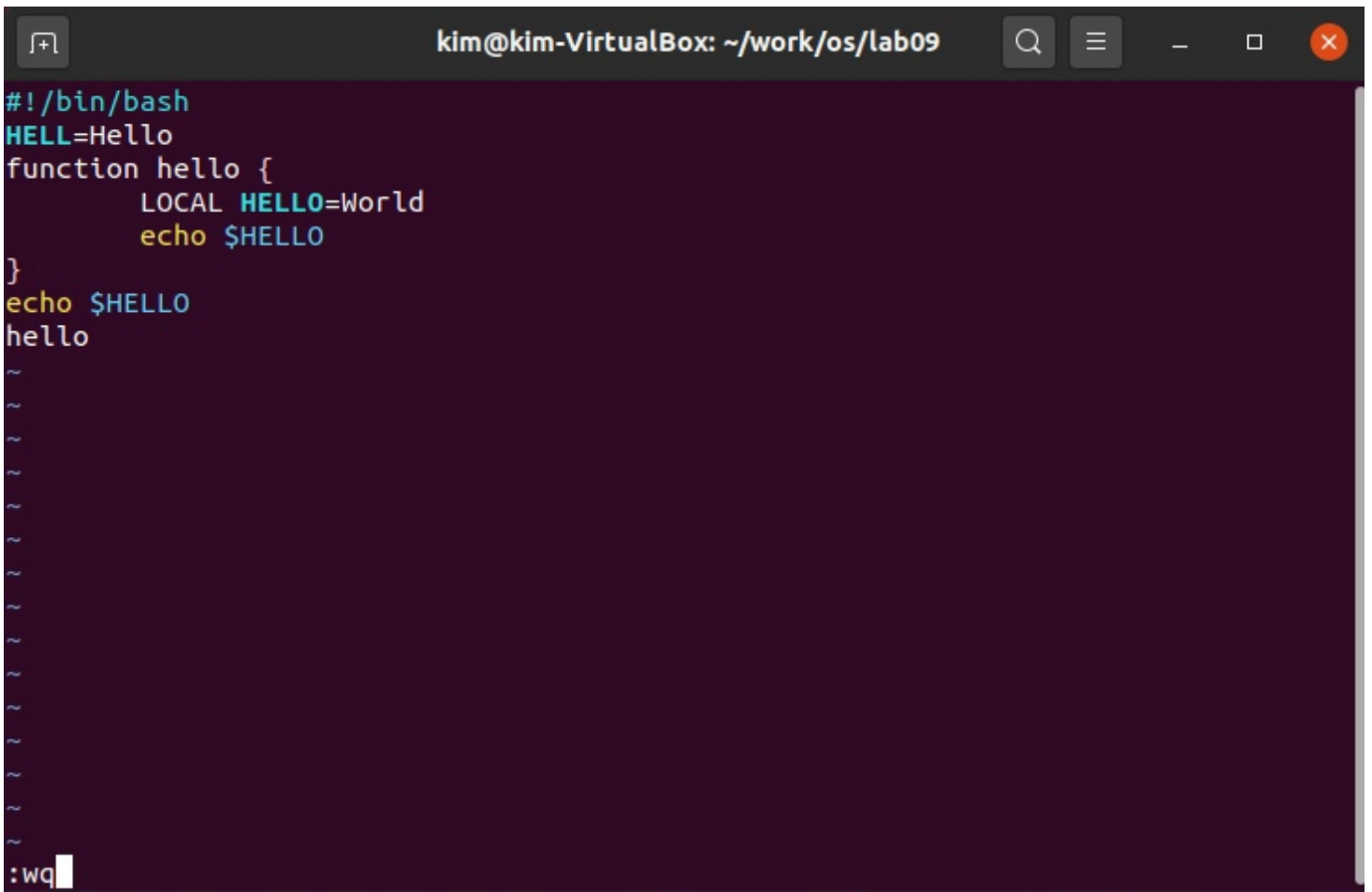
A screenshot of a terminal window titled "kim@kim-VirtualBox: ~/work/os/lab09". The terminal has a dark purple background. It shows a Bash shell prompt "#!/bin/bash" followed by a script:

```
HELL=Hello  
function hello {  
    LOCAL HELLO=World  
    echo $HELLO  
}  
echo $HELLO  
hello
```

The cursor is at the end of the "hello" command. Below the script, there are several tilde (~) characters representing output or continuation lines. At the bottom right of the terminal, the numbers "8,5" and the word "All" are visible.

6. Нажмите : для перехода в режим последней строки и внизу вашего экрана появится приглашение в виде двоеточия. (Рисунок 6)
7. Нажмите `w` (записать) и `q` (выйти) , а затем нажмите клавишу `Enter` для сохранения вашего текста и завершения работы.

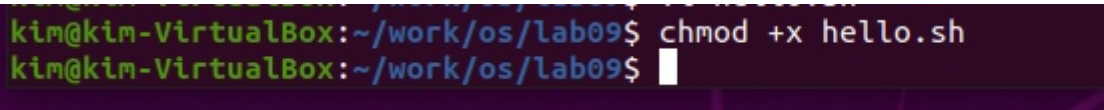
Рисунок 6: режим последней строки



```
kim@kim-VirtualBox: ~/work/os/lab09
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
~
~
~
~
~
~
~
~
~
~
~
:wq
```

8. Сделайте файл исполняемым : `chmod +x hello.sh`

Рисунок 7: делаем файл исполняемым

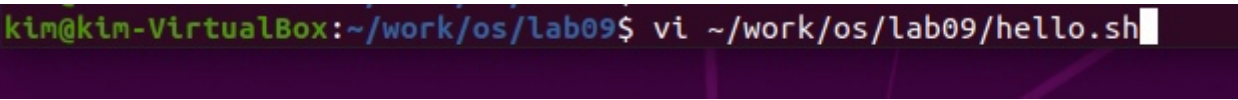


```
kim@kim-VirtualBox:~/work/os/lab09$ chmod +x hello.sh
kim@kim-VirtualBox:~/work/os/lab09$
```

Задание 2. Редактирование существующего файла

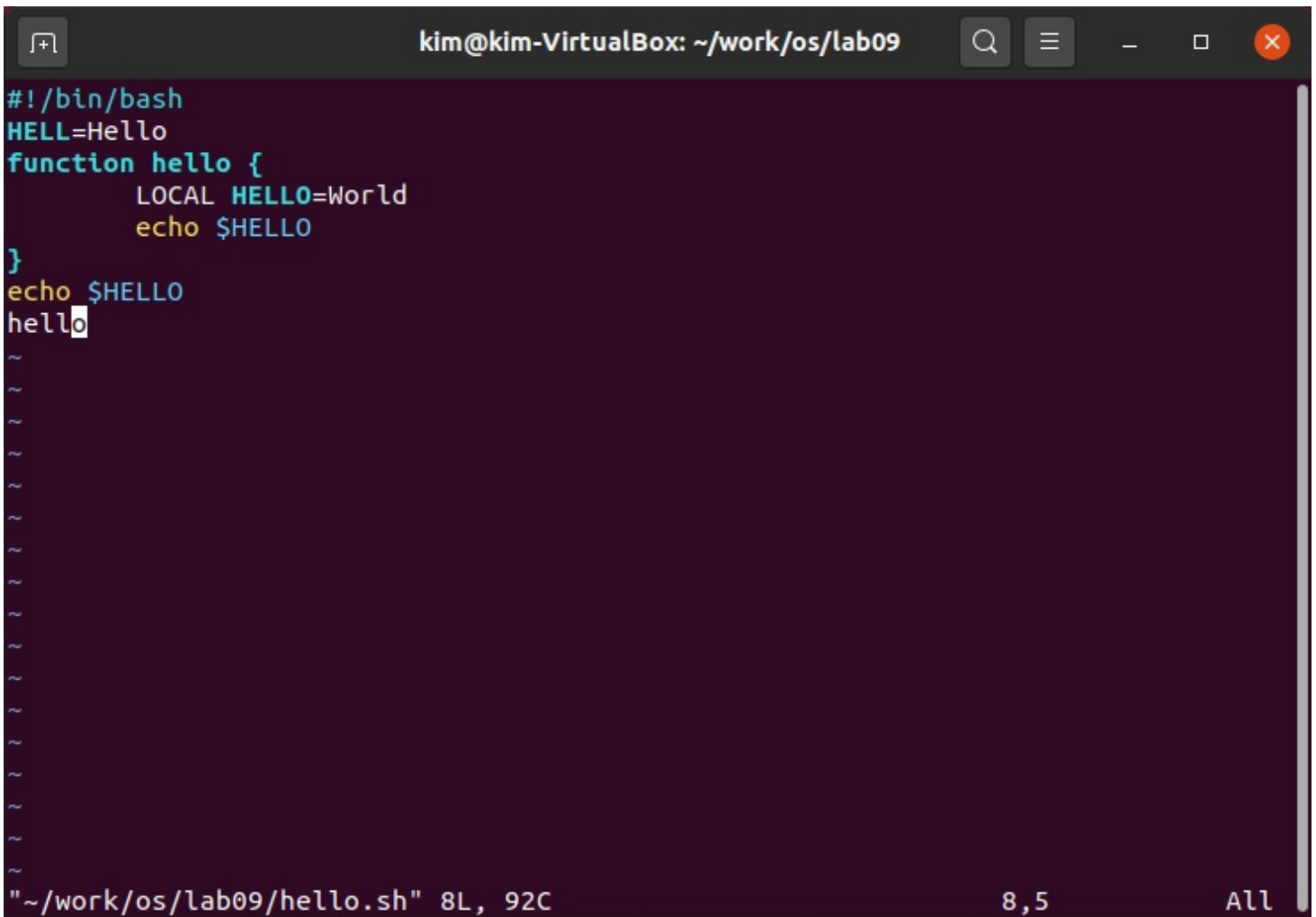
1. Вызовите vi на редактирование файла : `vi ~/work/os/lab09/hello.sh`

Рисунок 8: вызов vi на редактирование файла



```
kim@kim-VirtualBox:~/work/os/lab09$ vi ~/work/os/lab09/hello.sh
```

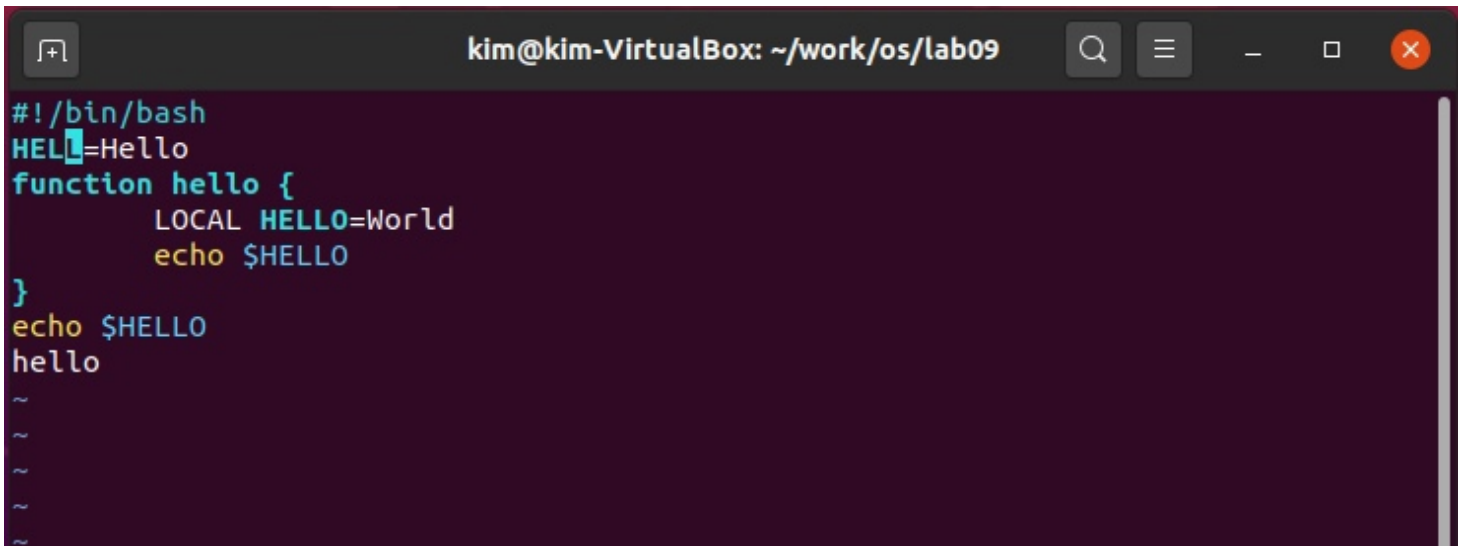
Рисунок 9: вызов vi на редактирование файла



```
kim@kim-VirtualBox: ~/work/os/lab09
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
~
~
~
~
~
~
~
~
~
~
~/work/os/lab09/hello.sh" 8L, 92C 8,5 All
```

2. Установите курсор в конец слова HELL второй строки

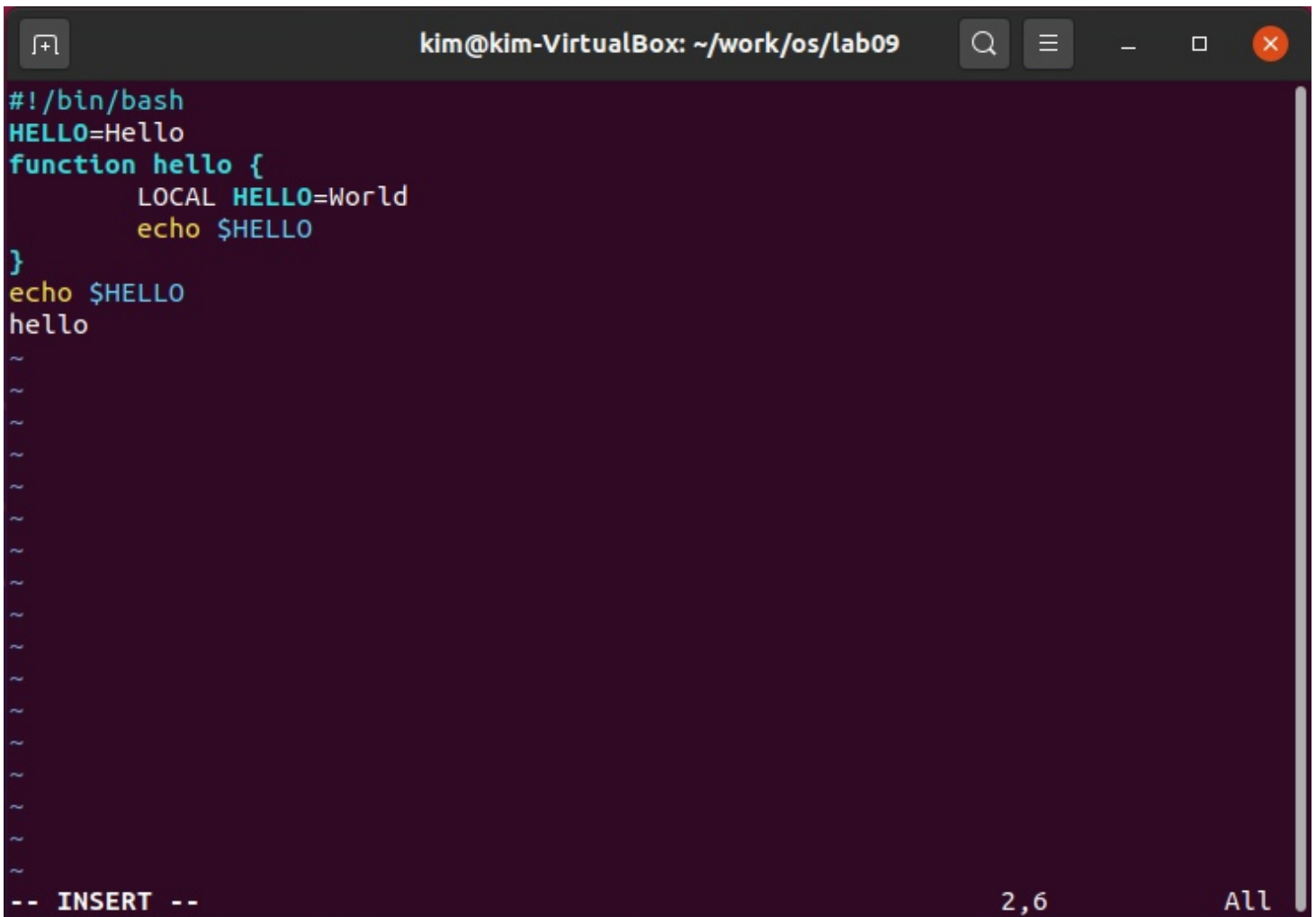
Рисунок 10: Установите курсор в конец слова HELL



```
kim@kim-VirtualBox: ~/work/os/lab09
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
~
~
~
~
~
```

3. Перейдите в режим вставки и замените на HELLO. Нажмите Esc для возврата в командный режим.

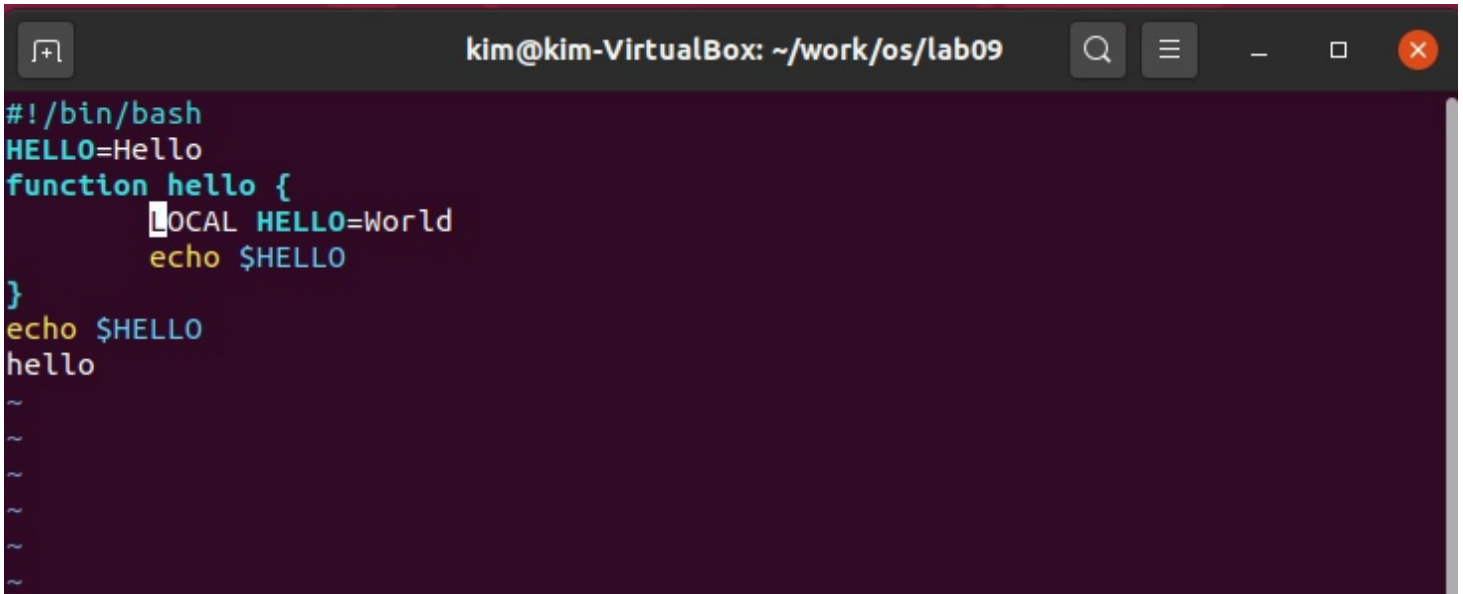
Рисунок 11: замена HELL на HELLO



```
kim@kim-VirtualBox: ~/work/os/lab09
#!/bin/bash
HELLO=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```

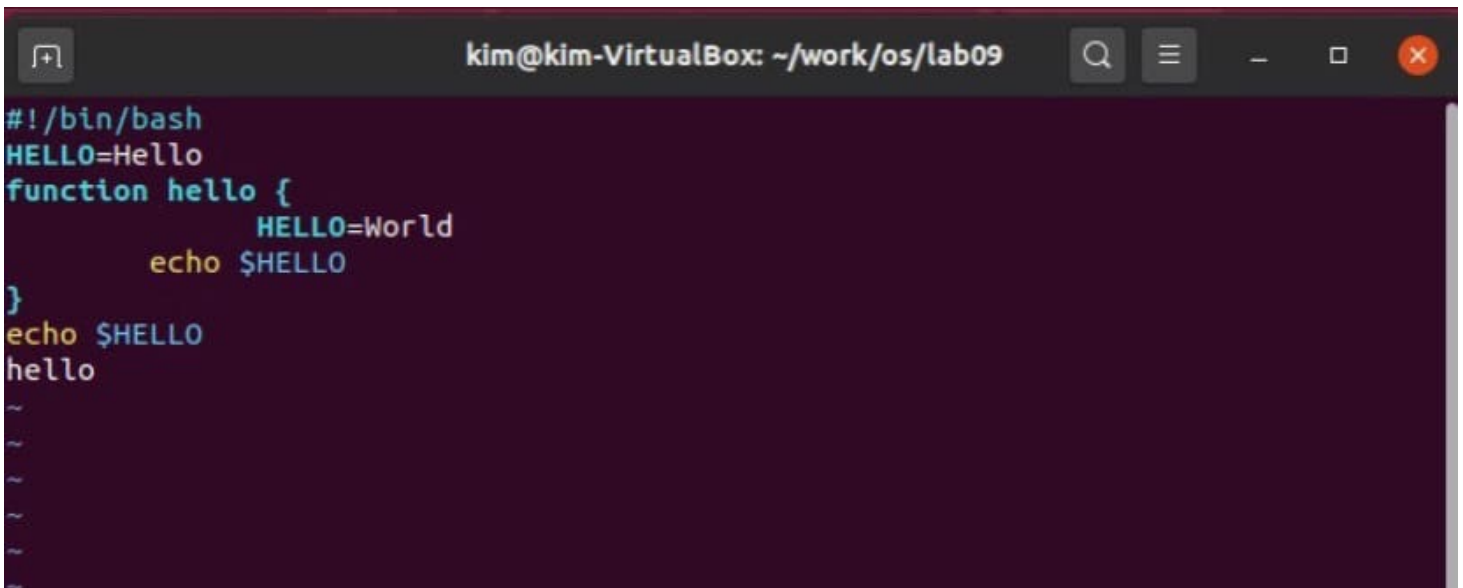
4. Установите курсор на четвертую строку и сотрите слово LOCAL : командой `d+w`

Рисунок 12: курсор на четвертую строку



```
kim@kim-VirtualBox: ~/work/os/lab09
#!/bin/bash
HELLO=Hello
function hello {
    HELLO=World
    echo $HELLO
}
echo $HELLO
hello
~
~
~
~
~
~
~
~
~
~
```

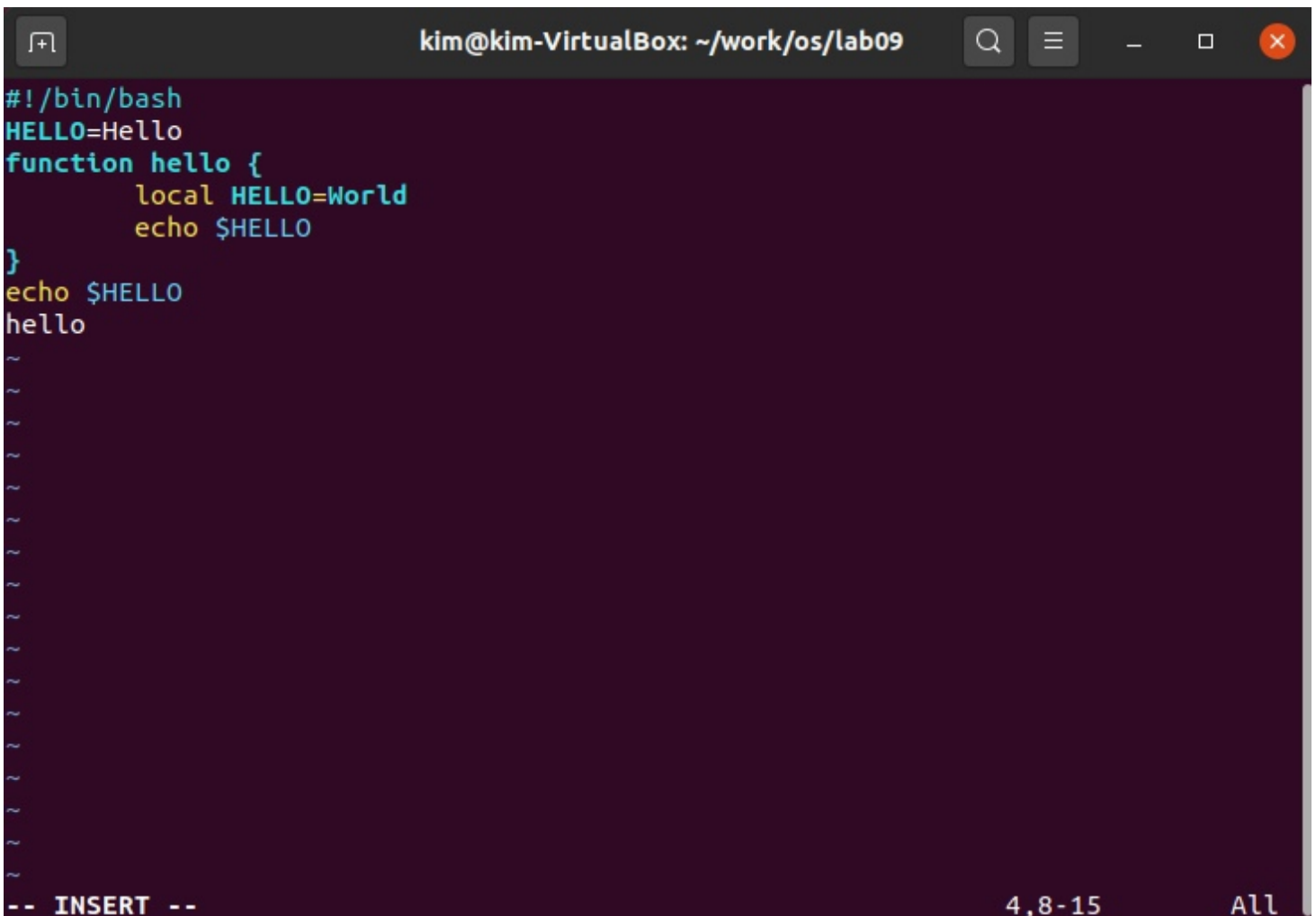
Рисунок 13: сотрите слово LOCAL



```
kim@kim-VirtualBox: ~/work/os/lab09
#!/bin/bash
HELLO=Hello
function hello {
    HELLO=World
    echo $HELLO
}
echo $HELLO
hello
~
~
~
~
~
```

5. Перейдите в режим вставки `i` и наберите следующий текст: `local`, нажмите `Esc` для возврата в командный режим.

Рисунок 14: ввод `local`




```
kim@kim-VirtualBox: ~/work/os/lab09
#!/bin/bash
HELLO=Hello
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```


7. Нажмите `Esc` для перехода в командный режим.

8. Удалите последнюю строку: Для этого мы используем команду `dd` для удаления строки.

Рисунок 17: уделим последнюю строку



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

The image shows a terminal window with a dark purple background. The text is displayed in a monospaced font with syntax highlighting: blue for keywords, yellow for variables, and white for other text. The cursor is positioned at the end of the word 'hello' on the last line of the script. Below the script, there are several tilde (~) characters representing prompt lines.

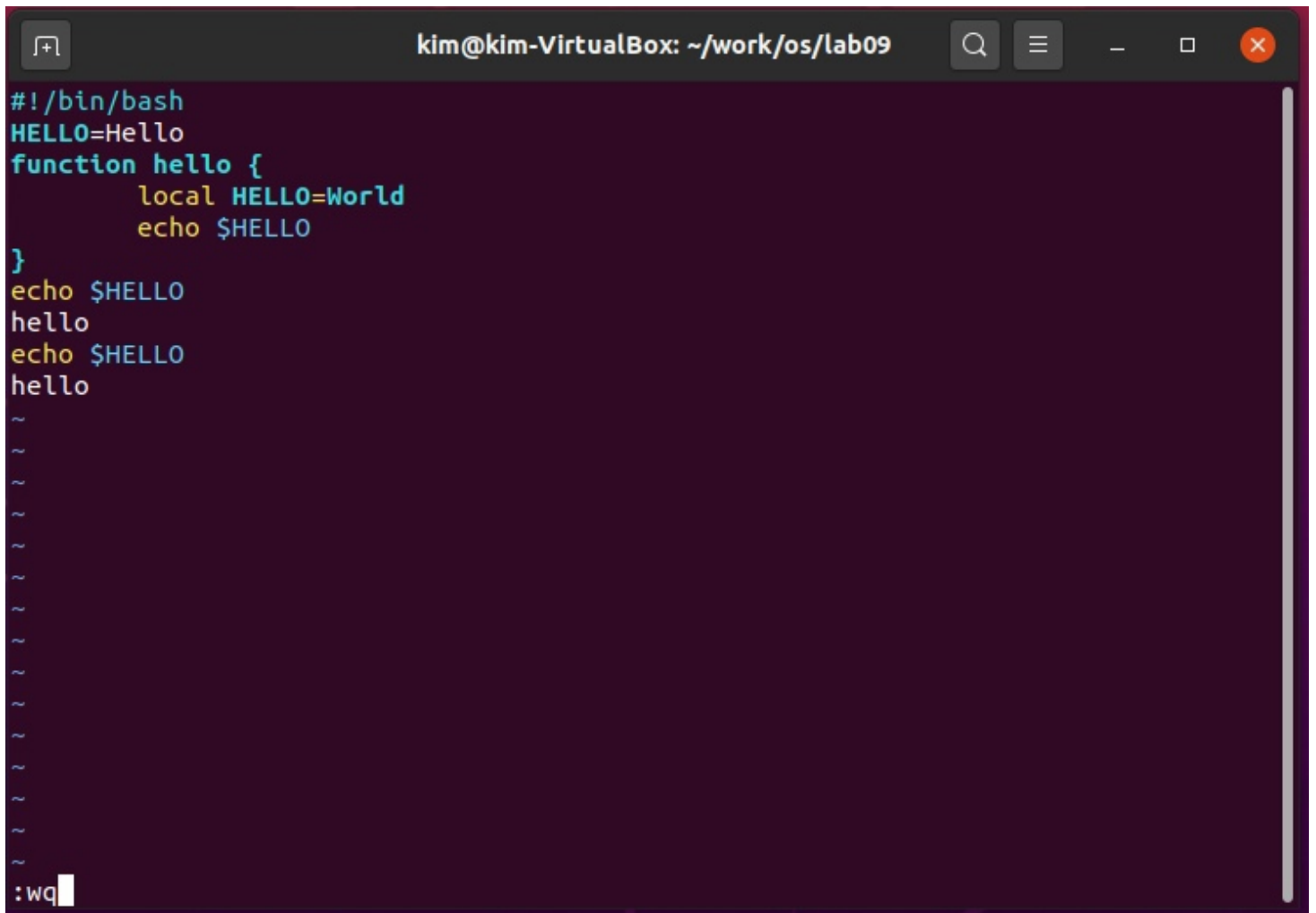
9. Введите команду отмены изменений `u` для отмены последней команды.

Рисунок 18: отмены последней команды

[illegible]

10. Введите символ `:` для перехода в режим последней строки, мы используем команды `w`(запись) и `q`(выход) после `:` (Рисунок 19) и нажимаем `Enter`. Запишите произведённые изменения и выйдите из `vi` (Рисунок 20).

Рисунок 19: режим последней строки



A terminal window titled "kim@kim-VirtualBox: ~/work/os/lab09". The window contains a bash script and its execution output. The script defines a function "hello" that sets a local variable "HELLO" to "World" and prints it. The script also prints the global variable "HELLO" which is "Hello". The output shows the script being executed twice, each time printing "Hello" and "World" on separate lines. The prompt ":wq" is visible at the bottom left.

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

Рисунок 20: выйдите из vi

```
kim@kim-VirtualBox: ~/work/os/lab09
ags.vim-tiny by vim-runtime'
Unpacking vim-runtime (2:8.1.2269-1ubuntu5) ...
Selecting previously unselected package vim.
Preparing to unpack .../vim_2%3a8.1.2269-1ubuntu5_amd64.deb ...
Unpacking vim (2:8.1.2269-1ubuntu5) ...
Setting up vim-runtime (2:8.1.2269-1ubuntu5) ...
Setting up vim (2:8.1.2269-1ubuntu5) ...
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vim (vim) in a
uto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vimdiff (vimdi
ff) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/rvim (rvim) in
auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/rview (rview)
in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vi (vi) in aut
o mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/view (view) in
auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/ex (ex) in aut
o mode
Processing triggers for man-db (2.9.1-1) ...
kim@kim-VirtualBox:~/work/os/lab09$ vi hello.sh
kim@kim-VirtualBox:~/work/os/lab09$ chmod +x hello.sh
kim@kim-VirtualBox:~/work/os/lab09$ vi ~/work/os/lab09/hello.sh
kim@kim-VirtualBox:~/work/os/lab09$
```

Вывод:

Я познакомилась с операционной системой Linux. Получила практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах

Библиография:

[1][Лабораторная работа №9](#)