

Лабораторная работа №8

Кочкарев “sakochkarev” Станислав

RUDN University

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

- Ознакомление с теоретическим материалом
- Ознакомление с редактором vi
- Выполнение упражнений с редактором vi
 - Создание нового файла с использованием vi
 - Редактирование существующего файла с использованием vi

Выполнение лабораторной работы

Первым делом мы ознакомились с приведенным теоретическим материалом.

Также перед лабораторной работой я ознакомился с
самом редактором vi.

Стоит упомянуть, что при выполнении данной лабораторной работы использовался редактор vi, отличия которого в рамках данной лабораторной работы несущественны.

После ознакомления со всем материалом я приступил
к выполнению основных заданий лабораторной
работы.

Первым заданием было создание нового файла с использованием редактора.

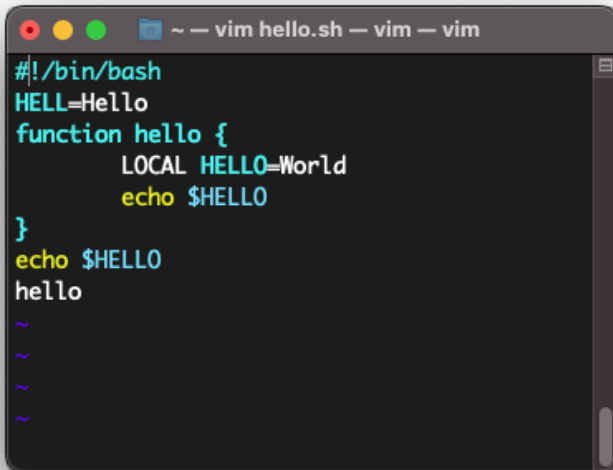
Создан каталог ~/work/os/lab06.

```
sakochkarev@sakochkarev [18:49:06] [~]  
[-> % mkdir -p ~/work/os/lab06
```

Выполнена команда `vim hello.sh`, создала и открыла в редакторе текстовый файл `hello.sh`.

```
sakochkarev@sakochkarev [18:49:19] [~/work/os/lab06]  
[-> % vim hello.sh
```

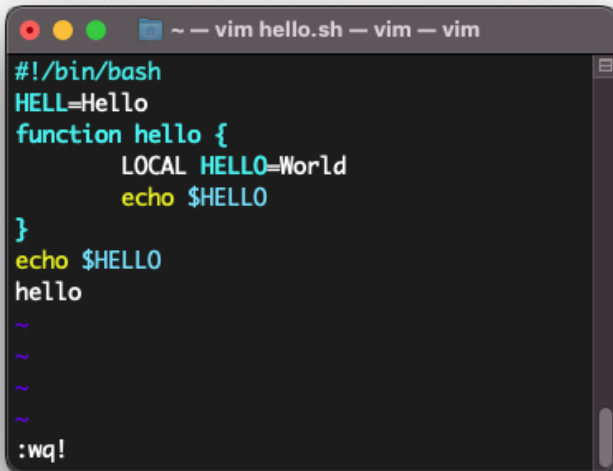
Переход в режим вставки и ввод приведенного текста.

A screenshot of a vim editor window. The title bar shows the window name as '~ — vim hello.sh — vim — vim'. The editor content is a shell script with the following lines:

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

 The text is color-coded: `#!/bin/bash` is white, `HELL=Hello` is white, `function hello {` is green, `LOCAL HELLO=World` is white, `echo $HELLO` is yellow, and `}` is green. There are four tilde (~) characters on the following lines. The window has standard macOS window controls (red, yellow, green buttons) and a scrollbar on the right.

Перевод редактора в командный режим и ввод команды **:wq!** для выхода из файла с сохранением.



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

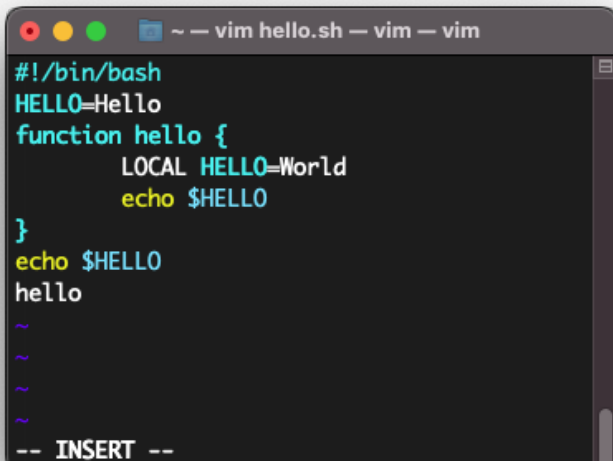
Изменение разрешений файла – выдача прав на выполнение файла командой `chmod +x hello.sh`.

```
sakochkarev@sakochkarev [18:50:28] [~/work/  
[os/lab06]  
-> % chmod +x hello.sh
```

Далее шли задания по редактированию существующего файла с помощью редактора vi.

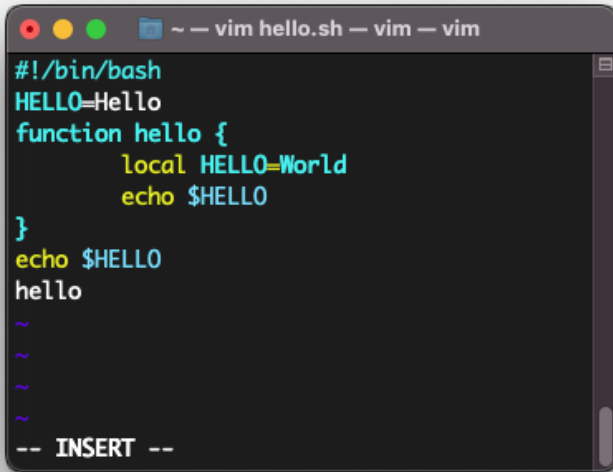
Открытие файла в редакторе командой `vi`
`~/work/os/lab06/hello.sh`.

Далее редактор был переведен в режим вставки и **HELL** было заменено на **HELLO**.

A screenshot of a terminal window with a vim editor. The title bar shows the file path and the editor name: '~ — vim hello.sh — vim — vim'. The script content is as follows:

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

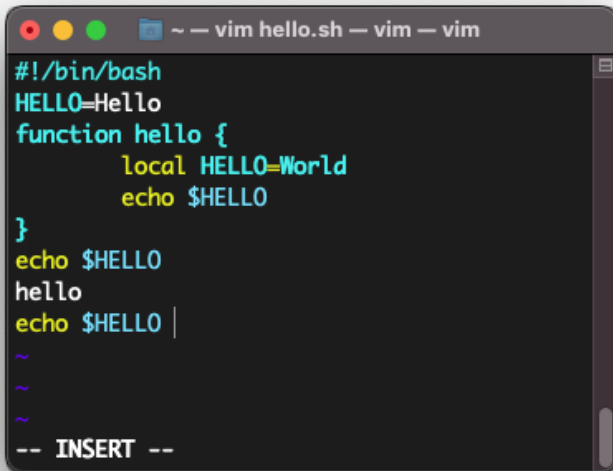

Курсор был установлен на четвертую строку и слово `LOCAL` было заменено на `local`.



The image shows a terminal window with the vim editor. The title bar indicates the file is `hello.sh` and the editor is in `vim` mode. The script content is as follows:

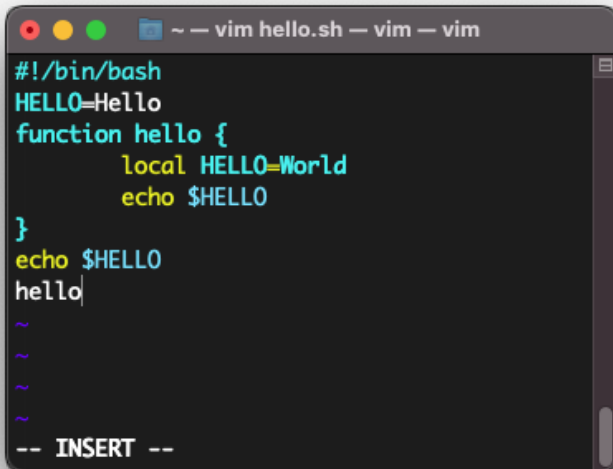
```
#!/bin/bash
HELLO=Hello
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello
~
~
~
~
-- INSERT --
```

Курсор был установлен на последнюю строку и после нее в режиме вставки была вставлена строка `echo $HELLO`.

A screenshot of a vim editor window. The title bar shows the file path '~ — vim hello.sh — vim — vim'. The editor content is a shell script with the following lines: `#!/bin/bash`, `HELLO=Hello`, `function hello {`, `local HELLO=World`, `echo $HELLO`, `}`, `echo $HELLO`, `hello`, and `echo $HELLO`. The cursor is positioned at the end of the last line. There are three tilde (~) symbols in the left margin. At the bottom, it says `-- INSERT --`.

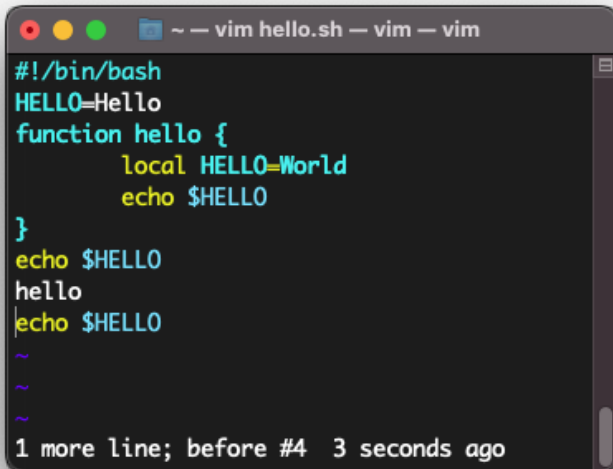
```
#!/bin/bash
HELLO=Hello
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO |
~
~
~
-- INSERT --
```

Удаление последней строки в командном режиме.



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

Отмена предыдущего действия

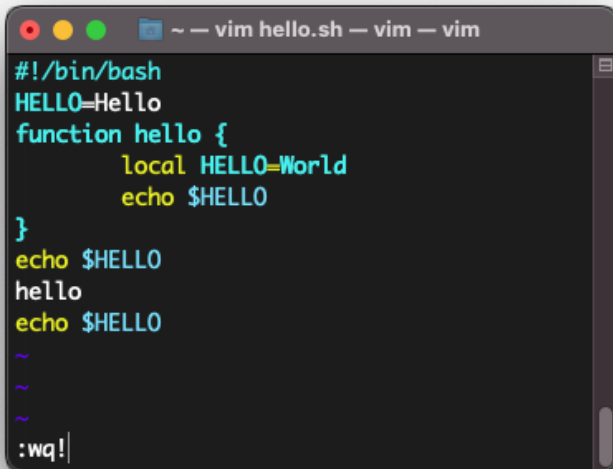
A screenshot of a vim editor window. The title bar shows the file path '~ — vim hello.sh — vim — vim'. The editor content is a shell script with the following lines:

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

 The cursor is positioned at the end of the last line. At the bottom of the window, a status line reads '1 more line; before #4 3 seconds ago'.

```
#!/bin/bash
HELLO=Hello
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO
~
~
~
1 more line; before #4 3 seconds ago
```

Выход из редактора с сохранением изменений

A screenshot of a vim editor window. The title bar shows the window name as '~ — vim hello.sh — vim — vim'. The editor content is a shell script with the following lines:

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

 The text is color-coded: shebangs in cyan, variable assignments in yellow, function definitions in cyan, and function calls in yellow. The cursor is at the end of the `:wq!` command on the last line.

По выполнении лабораторной работы мы получили практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах, а также немного дополнительно познакомились с операционной системой Linux.