

Toolbox: Vim

Toolbox: Vim

Learn vim, a universal text editor that can be incredibly powerful when used properly. From basic text editing to editing of binary files, Vim can be an important arsenal in a security toolkit.



[Task 1] Task 1

Lets get started!

To check whether Vim is installed:

- Launch a Terminal Window
- Type "Vim"

If you're looking at the Vim splash page

```
VIM - Vi IMproved

        version 8.0.600
        by Bram Moolenaar et al.
Vim is open source and freely distributable


        Become a registered Vim user!
type  :help register<Enter>    for information


type  :q<Enter>                to exit
type  :help<Enter>  or  <F1>    for on-line help
type  :help version8<Enter>    for version info


                                0.0-1      All
```

Then you're in luck!

Otherwise type:

Debian-Based Distributions:

sudo apt install vim

Arch-Based Distributions:

sudo pacman -S install vim

Fedora-Based Distributions:

sudo dnf install vim-enhanced

Windows:

Go to: <https://www.vim.org/download.php#pc>

Download and install the "self-installing-executable"

#1
Install Vim

No answer needed

#2
Launch Vim

No answer needed

[Task 1] Task 2

Lets get used to using Vim for basic text editing...

Modes

There are three basic modes in Vim:

- **Command** mode is where you can run commands. This is the default mode in which Vim starts up
- **Insert** mode is where you insert i.e. write the text
- **Visual** mode is where you visually select a bunch of text so that you can run a command/operation only on that part

of the text.

These form the pillars of navigating and using Vim. Try and answer the questions and work out how to begin creating a basic text document in Vim.

Navigation

Now that you know how to start creating a text document, you're going to need to know how to navigate it.

- left
- right
- up
- down
- jump to the start of a word
- jump to the end of a word

Hint, focus on key clusters, not combinations of keys for this.

Inserting Text

Now you're comfortable in basic typing and navigating the document, lets get into using vim in a more powerful way! The shortcuts for appending and inserting are crucial, I suggest trying to implement them into your typing routine so that they really stick.

FOR HELP WITH THE QUESTIONS

The easiest way to ask for help is to start with executing `:help` during a Vim session. This will drop us into the main help file which has an overview of the basics.

To get help with a specific command, we can provide that command as an argument to the `:help` command. By invoking `:help gg`, we learn more details about `gg` including that `<C-home>` does the same thing and that by providing a `[count]`, we can use `gg` to jump anywhere in a file.

#1

How do we enter "INSERT" mode?

i

#2

How do we start entering text into our new Vim document?

typing

#3

How do we return to command mode?

esc

#4

How do we move the cursor left?

h

#5

How do we move the cursor right?

l

#6

How do we move the cursor up?

k

#7

How do we move the cursor down?

j

#8

How do we jump to the start of a word?

W

#9

How do we jump to the end of a word?

e

#10

How do we insert (before the cursor)

i

#11

How do we insert (at the beginning of the line)

I

#12

How do we append (after the cursor)

a

#13

How do we append (at the end of the line)

A

#14

How do we make a new line under the current line?

O

#15

No answer needed

[Task 1] Task 3

First of all,

Congratulations!

You can now type in, utilise functions of and move around in a Vim document! Congratulations. You're now an order of magnitude more capable than you were before! Remember, repetition is key so keep going!

Now lets learn how to exit vim...



how to quit

how to quit **vim**

how to quit **smoking**

how to quit **drinking**

how to quit **smoking weed**

how to quit **vaping**

how to quit **juul**

how to quit **sugar**

how to quit **facebook**

how to quit **nicotine**

how to quit **dipping**

Google Search

I'm Feeling Lucky

The hardest drug of them all

There are six ways to exit vim, hold on, don't run away. They are:

- write the file, but don't exit [equivalent to save]
- ^ as root
- write and quit [equivalent to save and exit]
- quit [fails if there is unsaved changes]
- force quit [will exit with unsaved changes]
- save and quit [all active tabs]

Guide

I suggest making a small learning directory, and writing a few small documents, in Vim of course, to test these out on- if you get stuck, there's the help guide outlined in the last task- or you can use google.

#1

How do we write the file, but don't exit?

:w

#2

How do we write the file, but don't exit- as root?

:w !sudo tee %

#3

How do we write and quit?

:wq

#4

How do we quit?

:q

#5

How do we force quit?

:q!

#6

How do we save and quit, for all active tabs?

:wqa

[Task 1] Task 4

Before we get going...

You're doing really well! Keep it up! With practice you'll be an extremely competent Vim user!

One of the most common usages for a text editor is to copy, cut and paste things in and out of the editor- knowing how to do this is essential to being able to collate and use text in Vim.

These questions will focus on three commands:

- cut
- copy
- paste

Tackle these questions, and remember- once you've completed them keep practising!

Guide

I suggest making a small learning directory, and writing a few small documents, in Vim of course, to test these out on- if you get stuck, there's the help guide outlined in the last task- or you can use google.

#1

How do we copy a line?

yy

#2

how do we copy 2 lines?

2yy

#3

How do we copy to the end of the line?

y\$

#4

How do we paste the clipboard contents after the cursor?

p

#5

How do we paste the clipboard contents before the cursor?

P

#6

How do we cut a line?

d

#7

How do we cut two lines?

2dd

#8

How do we cut to the end of the line?

D

#9

How do we cut a character?

x

[Task 1] Task 5

Onto one of the last key things about using Vim, finding and replacing patterns.

Keep going!

We can use Vim to search for any literal pattern in the text, allowing you to search through large text files with ease. As well as allowing for replacing of patterns, or just repeat characters / white space.

vimgrep

You can also use "vimgrep" to use grep inside vim, allowing you to search using grep syntax and regex if you so desire.

Guide

I suggest making a small learning directory, and writing a few small documents, in Vim of course, to test these out on- if you get stuck, there's the help guide outlined in the last task- or you can use google.

#1

How do we search forwards for a pattern (use "pattern" for your answer)

/pattern

#2

How do we search backwards for a pattern (use "pattern" for your answer)

?pattern

#3

How do we repeat this search in the same direction?

n

#4

How do we repeat this search in the opposite direction?

N

#5

How do we search for "old" and replace it with "new"

:%s/old/new/g

#6

How do we use "grep" to search for a pattern in multiple files?

:vimgrep