


## Vim Tips

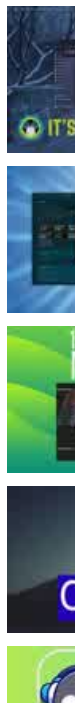
# Set Indentation Width to 2 or 4 Spaces (or Tab) in Vim

Prefer tabs to spaces? Here's how to set indentations to use spaces instead of tabs in Vim.

Jun 9, 2022 — Team LHB

### FEATURED VIDEOS

Powered by 



Vim is one of the most popular terminal-based text editors for decades.

But no matter how long you have been using Vim, there are always more things that you did not know about.

Subscribe

This one is about setting up indentation width in Vim to 2 spaces or 4 spaces. This is particularly helpful if you are a programmer, a Python one especially.

In your vimrc file (located at ~/.vimrc), add the following line to automatically use 2 spaces instead of tab in Vim.

```
set autoindent expandtab tabstop=2 shiftwidth=2
```

Don't worry. I'll go through it in detail.

## Set automatic indentation

Imagine you are writing a line of code and the next line of code needs to be indented, you press the Enter key to go to the next line but the indentation is not applied automatically.

This can be very annoying. To automatically indent lines, add the following line to your 'vimrc'.

```
set autoindent
```

Once you have this saved in your 'vimrc', it will enable automatic indentation in your vim session no matter the programming or scripting language that you use.

## Use spaces for indenting

If you want to use spaces for indenting your code, add the following lines to your 'vimrc' file.

```
set expandtab
set tabstop=<NUM OF SPACES>
set shiftwidth=<NUM OF SPACES>
```

- The first line enables `expandtab` option in Vim. This option makes sure that spaces

are used for indenting lines, even when you press the 'Tab' key.

- The second option `tabstop` takes a numerical value. Let's say I typed `set tabstop=2`, this will insert 2 spaces for a line indent.
- Finally, the third option, `shiftwidth` manages the indentation when you use the '>>' or '<<' operators to add or remove indentation to an already existing line/block of code.

I would suggest using 2 or 4 for `tabstop` and `shiftwidth` values.

I also recommend that you use the same values for `tabstop` and `shiftwidth`. Using different values might mess up your indentation.

Here is what indented code looks like in Vim:



```
0:3:zsh - "main.rs (~/.Downloads/dev_rust/guessing-game/src) - NVIM"
1 use colored::*;
2 use rand::Rng;
3 use std::cmp::Ordering;
4 use std::io;
5
6 fn read_str() -> String {
7     let mut mutable_temp: String = String::new();
8 }
```



In rare cases, if you need to use the tab character, pressing the 'Tab' key will not work with `expandtab` enabled. To use the tab character, use the 'Ctrl + V' key combination and then type the tab character.

## Vice versa: Use tabs for indentation

Serving to people standing on both sides of 'tabs vs spaces', let's now look at how you can

Serving to people standing on both sides of tabs vs spaces, let's now look at how you can use a tab character for indenting instead of spaces.

Add the following lines to your 'vimrc':

```
set noexpandtab
set tabstop=4
set shiftwidth=4
```

The only difference in these 3 lines for your vimrc to use tabs instead of spaces is the usage of `noexpandtab` instead of `expandtab`. The `noexpandtab` option prevents the conversion of tabs to spaces.

## Conclusion

All the above works for new files. To convert tabs to spaces in the currently opened file in Vim, enter the Normal mode by pressing Esc key. Now use the `retab` command by pressing the ':' (colon) character and Vim will convert the existing tabs to spaces.

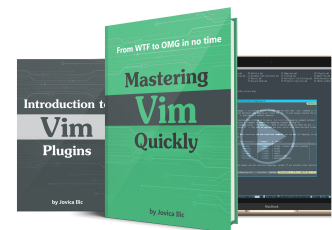
The war of tabs vs spaces aside, this tutorial aims at helping you set the indentation to whatever your preference might be.

If you are interested in learning more than just the Vim Basics, use this program by Jovica Ilic.

### Mastering Vim Quickly - Jovica Ilic

Exiting Mastering Vim Quickly From WTF to OMG in no time

Jovica Ilic



Linux Handbook  
@itsfoss

7  
VOTERS



## Test your knowledge of Popular desktop environments!

START QUIZ

**What's your reaction?**

**2 Comments** 22 ONLINE

[LOGIN](#)

We encourage polite and civil discussion.

Write your comment...

Top

J **Jojo** 13 days ago  
very cool thx

0 0 Reply



**Abhishek** Team LHB 13 days ago

You are welcome, Jojo :)

0 0 Reply



An independent, reader-supported publication focusing on  
Linux Command Line, Server, Self-hosting, DevOps and Cloud  
Learning

---

### Subscribe to LHB Linux Digest Newsletter

Get a curated assortment of Linux tips, tutorials and memes directly in your inbox. Over 14,000 Linux users enjoy it twice a month.

---

[Contact](#)[Privacy Policy](#)[Write for Us](#)[Pro Resources](#)[Newsletters](#)[Membership FAQ](#)

Linux Handbook © 2023 • Hosted on **DigitalOcean**, Published with **Ghost**, using **Spiritix** theme