

Buffers

Goal:

The goal of this exercise is to give you practice working with multiple buffers.

Instructions:

Open the buf* files

First, start a command line session on your local machine. Next, use vim to open all the files that start with "buf" that came in the course downloads. To do that, navigate to the location of the file. Remember this could be different for you depending on where you extracted the contents of the course archive. This example assumes the course download was saved into your Downloads folder and extracted from there.

```
cd Downloads
cd vimclass
vim buf*
```

View the buffers

Look at your buffer list with either the **:buffers** command. You can also use **:files** or **:ls** as they are all the same command. Here is what you should see when you execute the command:

```
:ls
 1 %a      "buf-ant.txt"          line 1
 2         "buf-bed.txt"          line 0
 3         "buf-cat.txt"          line 0
 4         "buf-dad.txt"          line 0
```

Switch buffers

The **:buffer** command, or **:b** for short, allows you to switch buffers. Open the buffer associated with the "buf-bed.txt" file by using it's unique number. To do that, type **:b2<ENTER>**.

Now switch to the buffer associated with the "buf-cat.txt" file by using it's file name. To do that, type **:b buf-cat.txt<ENTER>**.

Use tab completion to open the buffer associated with the "buf-dad.txt" file. To do that, type **:b<SPACE><TAB><TAB><TAB><TAB><ENTER>**.

Quickly return to the previously opened buffer by typing **Ctrl-^**. You should now be looking at the contents of the "buf-cat.txt" file.

Look at your buffers with **:ls<ENTER>** and note the indicators. Buffer 3 displays "%a" which means it's the buffer in the current window and it's active. Buffer 4 displays "#" which makes it the alternative buffer.

```
:ls
1      "buf-ant.txt"          line 1
2      "buf-bed.txt"          line 1
3 %a   "buf-cat.txt"          line 0
4 #    "buf-dad.txt"          line 1
```

Switch back to buffer 4 with **Ctrl-^**.

Move to buffer 3 with the **:bprevious** command. If you prefer, you can use the short version of the command by typing **:bp<ENTER>**.

Move to buffer 2 by typing **:bp<ENTER>**.

Now reverse the direction and move to the next buffer which is buffer 3 with the **:bnext** command or the short version, **:bn<ENTER>**.

Go to the first buffer in your buffer list with the **:bfirst** command. Of course, you can use the short version by typing **:bf<ENTER>**.

No just to the last buffer in the buffer list with **:blast** or **:bl<ENTER>**.

Edit multiple buffers at once

Make a change to the current buffer. For example, you could enter insert mode with **i** and then type some text such as **dad** and then return to normal mode by typing **<ESCAPE>**.

Try to switch to buffer 2 by typing **:b2<ENTER>**. If you do not have the hidden option enabled, you will see an error message such as this:

```
E37: No write since last change (add ! to override)
```

Follow the instructions and type **:b!2<ENTER>**. Now buffer 2 is loaded into the window. Look at your buffers with **:ls<ENTER>**. You'll notice the previous buffer has the indicators **h** and **+** which mean it is a hidden buffer with modifications.

```
:ls
1      "buf-ant.txt"          line 1
2 %a   "buf-bed.txt"         line 1
3      "buf-cat.txt"         line 0
4 #h + "buf-dad.txt"         line 1
```

Enable the hidden option by typing **:set hidden<ENTER>**. Now make a change to this buffer. For example, you could enter insert mode with **i** and then type some text such as **bed** and then return to normal mode by typing **<ESCAPE>**.

Switch to buffer 1 by typing **:b1<ENTER>**. Notice that you did not get an error message. Look at your buffers with **:ls<ENTER>**. You'll now notice buffers 2 and 4 have the indicators **h** and **+** which mean they are hidden buffers with modifications.

```
:ls
1 %a   "buf-ant.txt"          line 1
2 #h + "buf-bed.txt"         line 1
3      "buf-cat.txt"         line 0
4  h + "buf-dad.txt"         line 1
```

Add and delete buffers

Open up the file named "nav.txt" for editing. To do that type **:e<SPACE>nav.txt<ENTER>**. Now the contents of the file "nav.txt" are loaded in the buffer display in your window. You can list your buffers by typing **:ls<ENTER>**. You should see the following:

```
:ls
1 #h   "buf-ant.txt"          line 1
2  h + "buf-bed.txt"         line 1
3      "buf-cat.txt"         line 1
4  h + "buf-dad.txt"         line 1
5 %a   "nav.txt"             line 1
```

Let's say you decided you don't want to edit the "nav.txt" file, so delete it by typing **:bd<ENTER>**. Examine your buffers with **:ls<ENTER>**. Now your buffers will look like so:

```
:ls
1 %a   "buf-ant.txt"          line 1
2  h + "buf-bed.txt"         line 1
3      "buf-cat.txt"         line 1
4  h + "buf-dad.txt"         line 1
```

Delete the buffer with an ID of 3 by typing **:bd3<ENTER>**. Examine your buffers with **:ls<ENTER>**. Now your buffers will look like so:

```
:ls
 1 %a      "buf-ant.txt"          line 1
 2  h +    "buf-bed.txt"          line 1
 4  h +    "buf-dad.txt"          line 1
```

Use the Explorer

Use the explorer to open the "help.txt" file. Start the explorer by typing **:E<ENTER>**. Now, use the vim navigation commands you know to place your cursor under the file named "help.txt." Finally, press **<ENTER>** to load the buffer into the current window.

Now discard this buffer with **:bd<ENTER>**.

Execute a command in all buffers

Perform a global substitution in all the buffers. Remember that substitute command is **:[range]s[substitute]/{pattern}/{string}/[flags]** Also remember that the range of % is the entire file and the **g** flag replaces all occurrences in the line. So the global substitution command is effectively **:%s/{old}/{new}/g**. Use this in combination with the **:bufdo** command to replace every # character with the @ character.

To do that, type **:bufdo %s/#/@/g<ENTER>**. Examine your buffers with **:ls<ENTER>**. You should see that they've all been modified, indicated by the + character. Quickly check that the substitution occurred in another buffer by typing **:bf<ENTER>**, which loads the first buffer. It should look like the following:

```
  @      @      @  @@@@@@@@
  @ @      @@      @      @
  @      @  @ @      @      @
  @      @  @  @      @      @
  @@@@@@@@  @  @ @      @
  @      @  @      @@      @
  @      @  @      @      @
```

This training is provided by [LinuxTrainingAcademy.com](http://www.LinuxTrainingAcademy.com).

Abandon your changes to all buffers

If you want to try this exercise again in the future, abandon all your changes with the **:qall!<ENTER>** command. If you want to do the opposite and save all your changes you can use the **:wall** command.