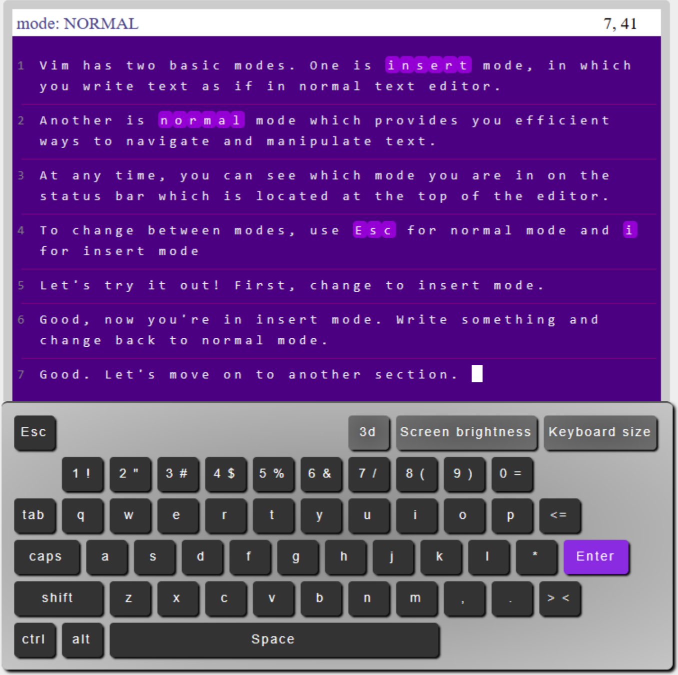
Two modes, insert and normal



mode: NORMAL 7, 41

1 Vim has two basic modes. One is insert mode, in which you write text as if in normal text editor.

2 Another is normal mode which provides you efficient ways to navigate and manipulate text.

3 At any time, you can see which mode you are in on the status bar which is located at the top of the editor.

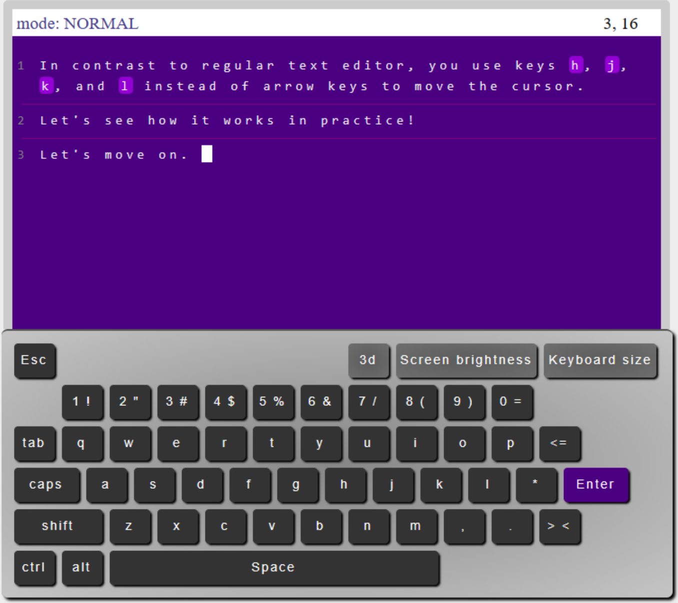
4 To change between modes, use Esc for normal mode and i for insert mode

5 Let's try it out! First, change to insert mode.

6 Good, now you're in insert mode. Write something and change back to normal mode.

7 Good. Let's move on to another section.

Basic movement: h, j, k, and l



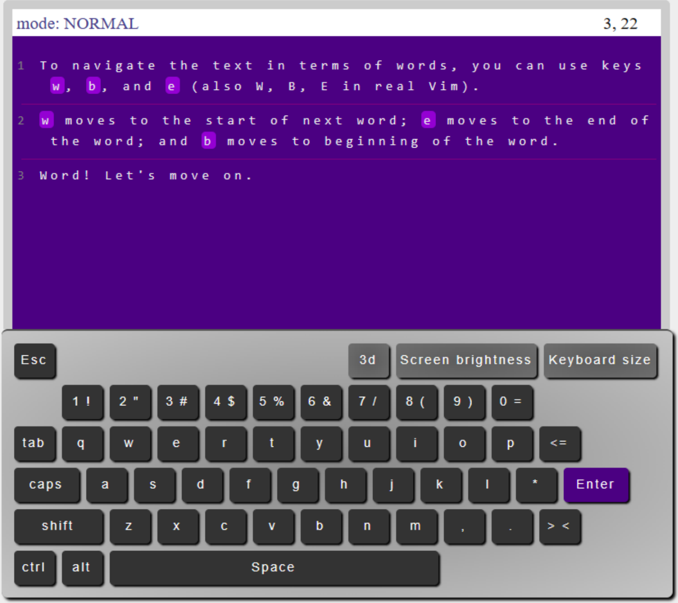
mode: NORMAL 3, 16

```
In contrast to regular text editor, you use keys h, j,
k, and l instead of arrow keys to move the cursor.

Let's see how it works in practice!

Let's move on.
```

Word movement: w, e, b



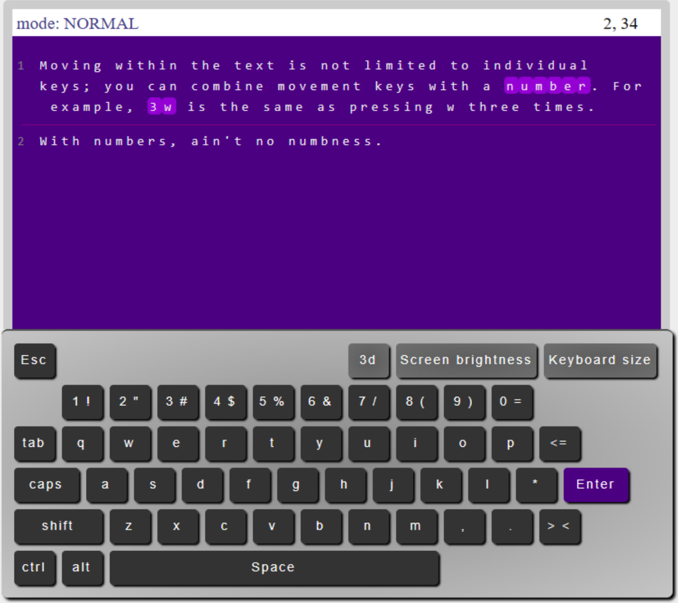
mode: NORMAL 3, 22

To navigate the text in terms of words, you can use keys w, b, and e (also W, B, E in real Vim).

w moves to the start of next word; e moves to the end of the word; and b moves to beginning of the word.

Word! Let's move on.

Number powered movement, e.g. 5w

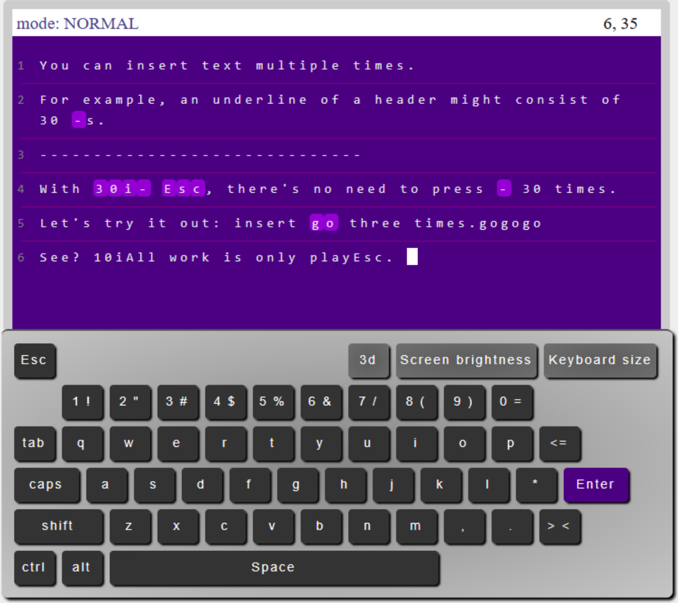


mode: NORMAL 2, 34

Moving within the text is not limited to individual keys; you can combine movement keys with a number. For example, 3w is the same as pressing w three times.

With numbers, ain't no numbness.

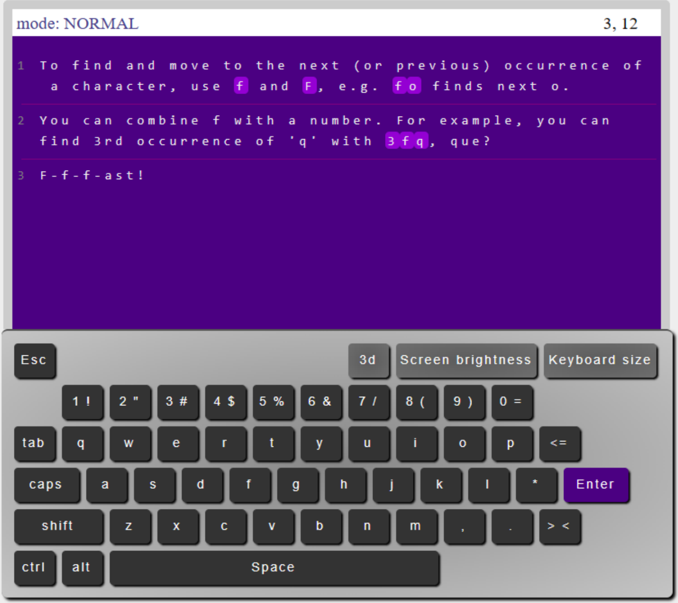
Insert text repeatedly, e.g. 3iYes



mode: NORMAL 6, 35

```
You can insert text multiple times.
For example, an underline of a header might consist of 30 -s.
------------------------------
With 30i- Esc, there's no need to press - 30 times.
Let's try it out: insert go three times.gogogo
See? 10iAll work is only playEsc.
```

Find a character, f and F



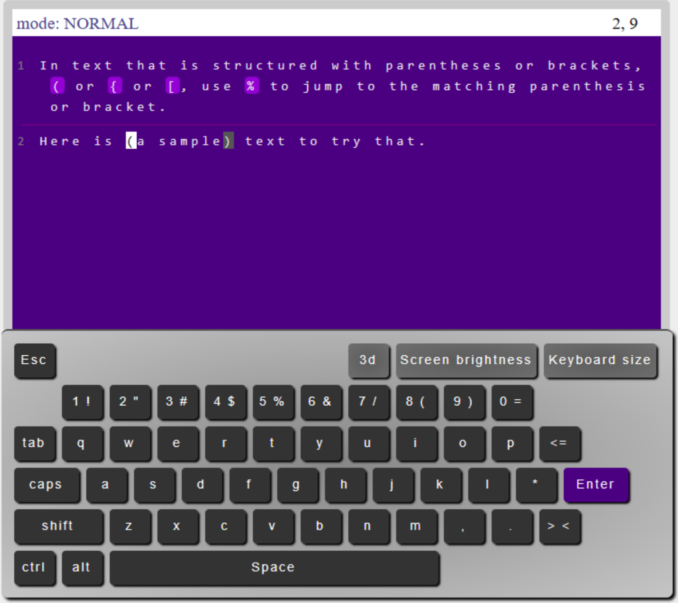
mode: NORMAL 3, 12

To find and move to the next (or previous) occurrence of a character, use f and F, e.g. fo finds next o.

You can combine f with a number. For example, you can find 3rd occurrence of 'q' with 3fq, que?

F-f-f-ast!

Go to matching parentheses, %

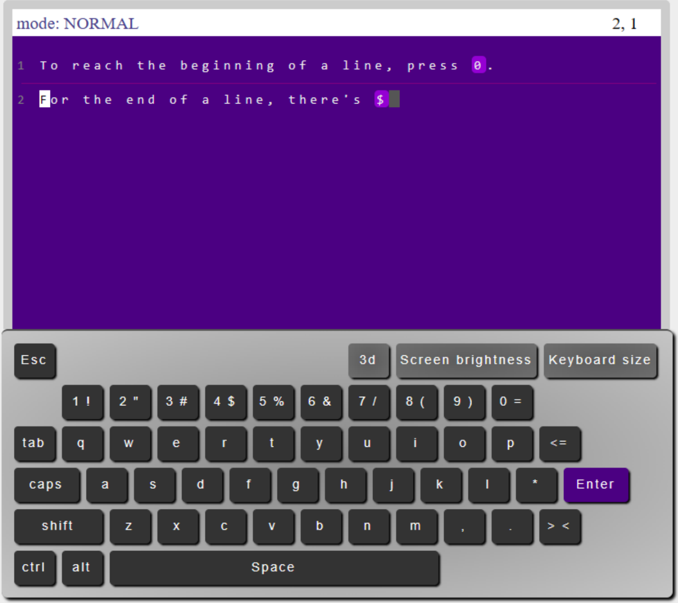


mode: NORMAL 2, 9

1 In text that is structured with parentheses or brackets, (or { or [, use % to jump to the matching parenthesis or bracket.

2 Here is (a sample) text to try that.

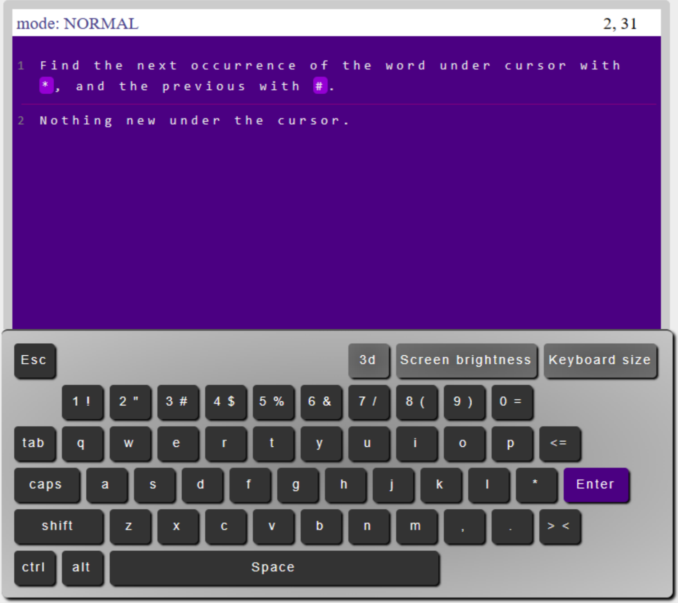
Go to start/end of line, 0 and $



mode: NORMAL 2, 1

```
To reach the beginning of a line, press 0.
For the end of a line, there's $
```

Find word under cursor, * and

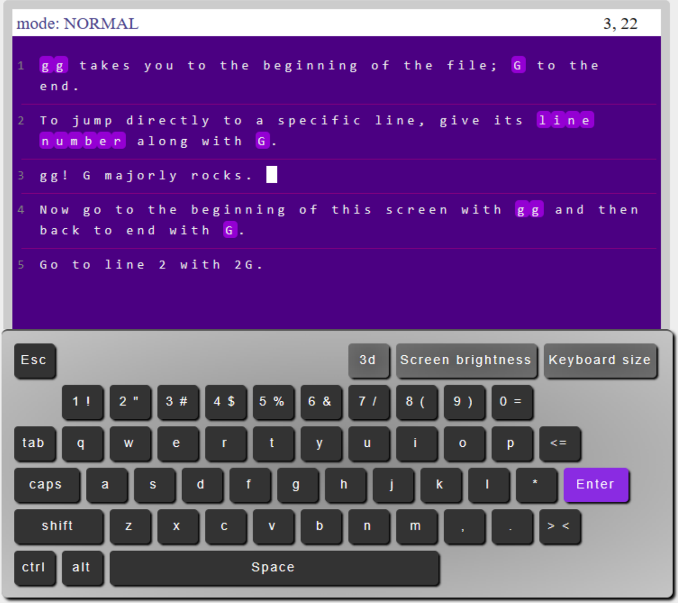


mode: NORMAL 2, 31

Find the next occurrence of the word under cursor with *, and the previous with #.

Nothing new under the cursor.

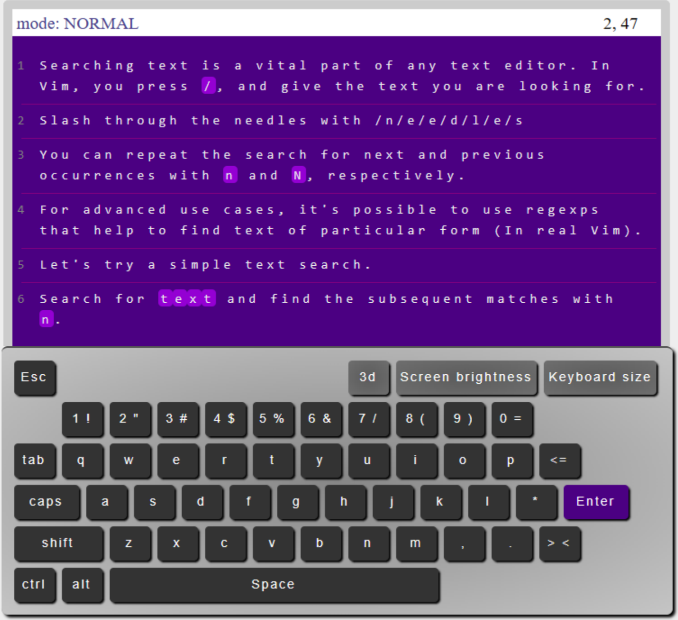
Goto line, g and G



mode: NORMAL 3, 22

```
1 gg takes you to the beginning of the file; G to the
  end.
2 To jump directly to a specific line, give its line
  number along with G.
3 gg! G majorly rocks.
4 Now go to the beginning of this screen with gg and then
  back to end with G.
5 Go to line 2 with 2G.
```

Search, /text with n and N



mode: NORMAL 2, 47

1 Searching text is a vital part of any text editor. In Vim, you press /, and give the text you are looking for.

2 Slash through the needles with /n/e/e/d/l/e/s

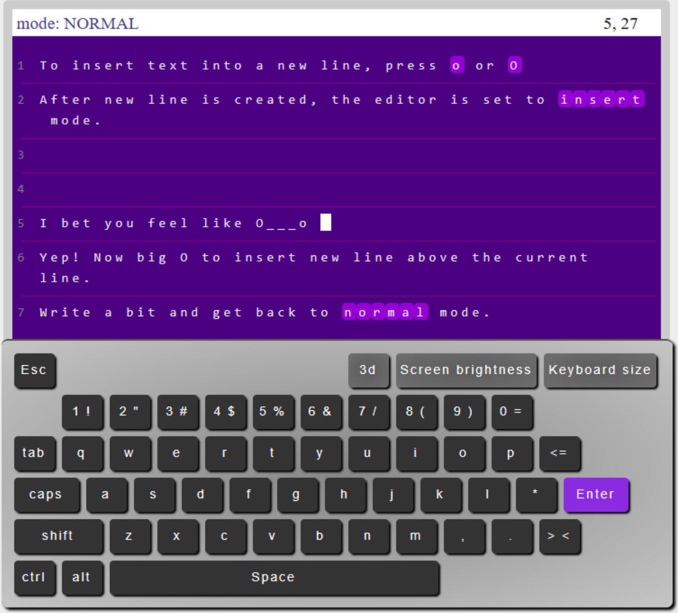
3 You can repeat the search for next and previous occurrences with n and N, respectively.

4 For advanced use cases, it's possible to use regexps that help to find text of particular form (In real Vim).

5 Let's try a simple text search.

6 Search for text and find the subsequent matches with n.

Insert new line, o and O

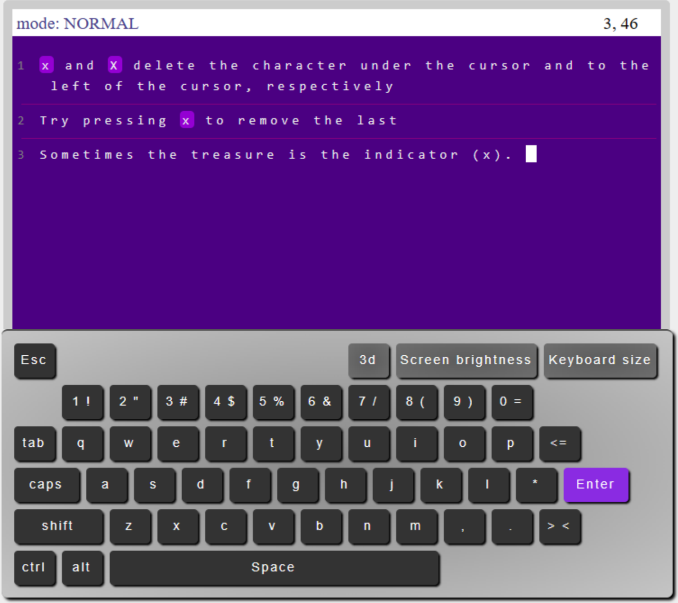


mode: NORMAL 5, 27

```
To insert text into a new line, press o or O
After new line is created, the editor is set to insert
 mode.


I bet you feel like O___o
Yep! Now big O to insert new line above the current
line.
Write a bit and get back to normal mode.
```

Removing a character, x and X



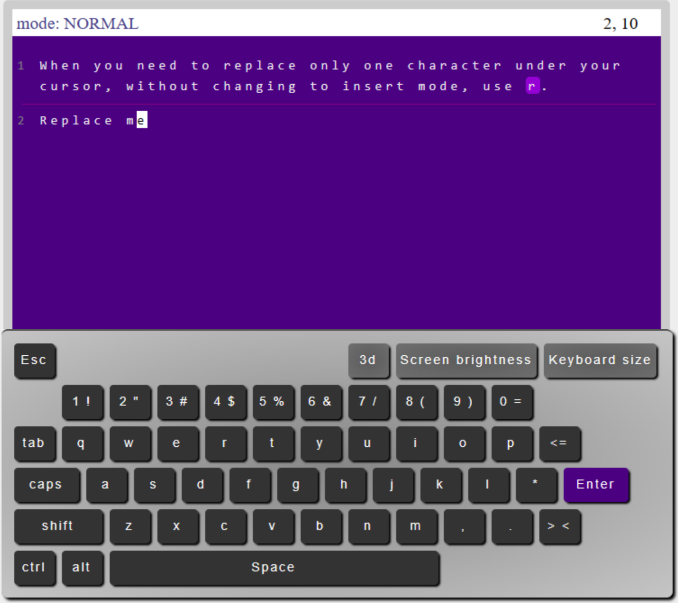
mode: NORMAL 3, 46

x and X delete the character under the cursor and to the left of the cursor, respectively

Try pressing x to remove the last

Sometimes the treasure is the indicator (x).

Replacing letter under cursor, r

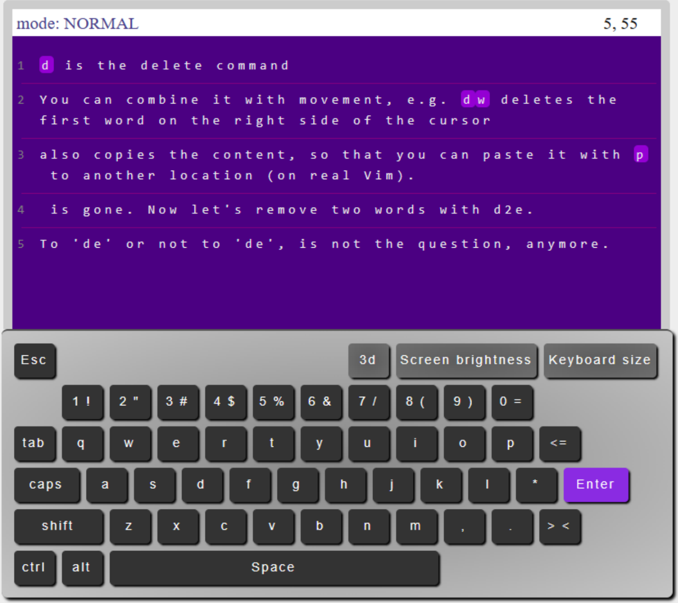


mode: NORMAL 2, 10

```
When you need to replace only one character under your
cursor, without changing to insert mode, use r.

Replace me
```

Deleting, d



mode: NORMAL 5, 55

d is the delete command

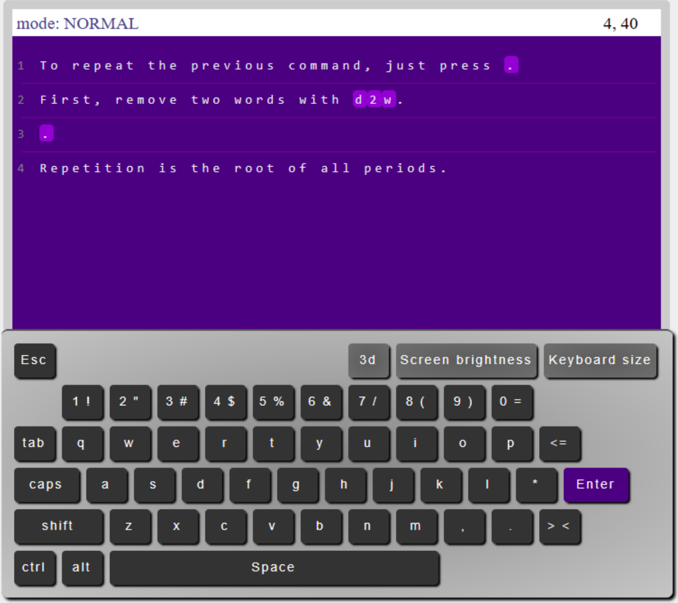
You can combine it with movement, e.g. dw deletes the first word on the right side of the cursor

also copies the content, so that you can paste it with p to another location (on real Vim).

is gone. Now let's remove two words with d2e.

To 'de' or not to 'de', is not the question, anymore.

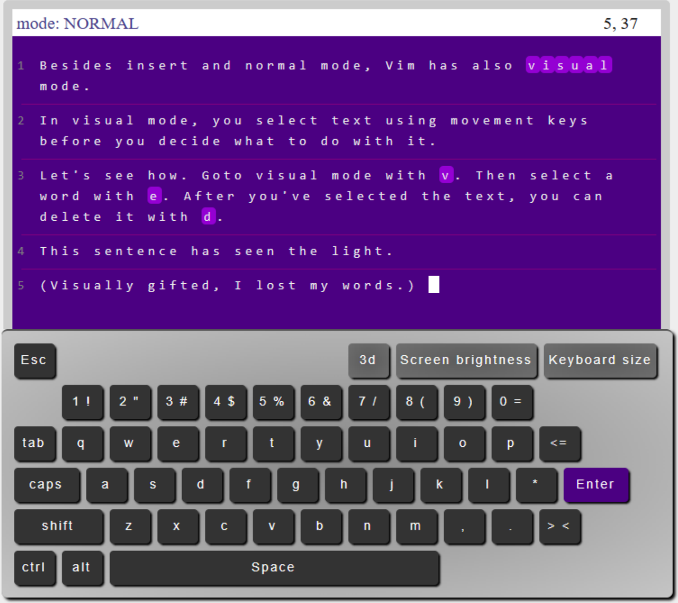
Repetition with .



mode: NORMAL 4, 40

```
To repeat the previous command, just press .
First, remove two words with d2w.
.
Repetition is the root of all periods.
```

Visual mode, v



mode: NORMAL 5, 37

Besides insert and normal mode, Vim has also visual mode.

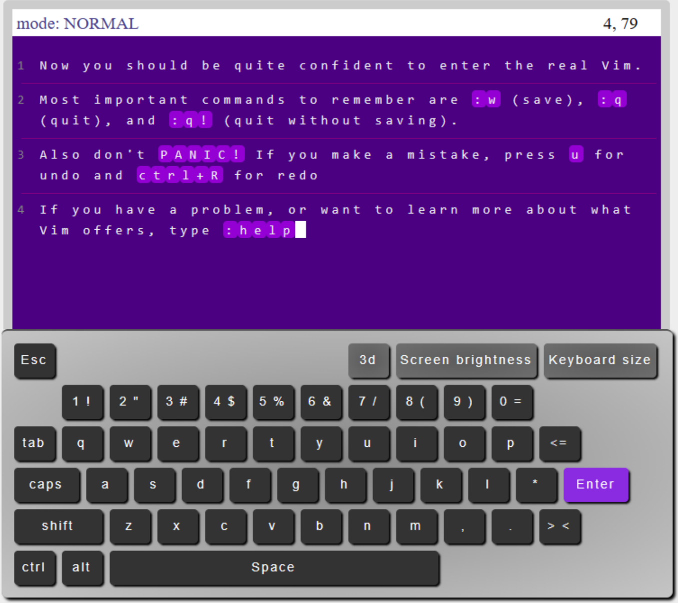
In visual mode, you select text using movement keys before you decide what to do with it.

Let's see how. Goto visual mode with v. Then select a word with e. After you've selected the text, you can delete it with d.

This sentence has seen the light.

(Visually gifted, I lost my words.)

Real Vim awaits



mode: NORMAL 4, 79

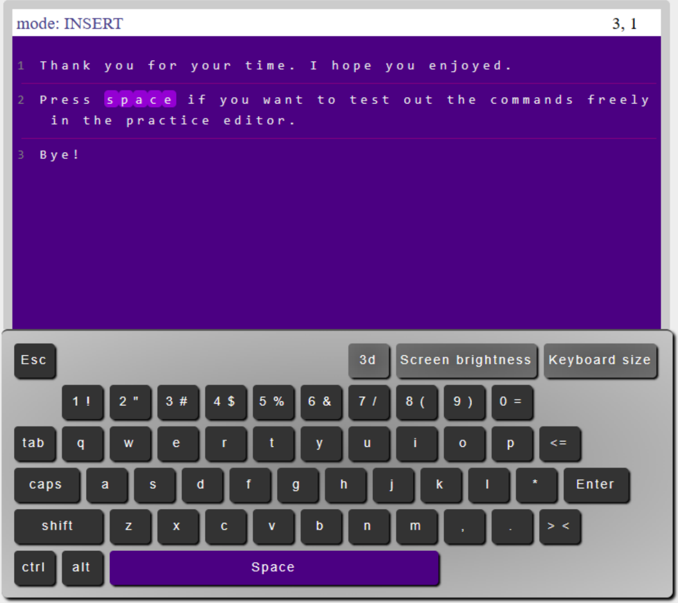
Now you should be quite confident to enter the real Vim.

Most important commands to remember are :w (save), :q (quit), and :q! (quit without saving).

Also don't PANIC! If you make a mistake, press u for undo and ctrl+R for redo

If you have a problem, or want to learn more about what Vim offers, type :help

The end



mode: INSERT 3, 1

```
Thank you for your time. I hope you enjoyed.
Press space if you want to test out the commands freely
 in the practice editor.
Bye!
```