

# NEW POST

[waylonwalker.com/svim](http://waylonwalker.com/svim)

save  
**vim**  
Macro



@\_waylonwalker

# Vim Macros

If you are like me you have created a macro or two that is pure glory and you forget how you made it after a day or so

- Create
- Replay
- List all
- Save



[waylonwalker.com/svim](http://waylonwalker.com/svim)  
@\_waylonwalker

# Create A Macro

One of the earlies things we all learn to do in vim is to create macros, custom sets of functionality that get stored in a register and can be replayed later.

qq

Note: a common throw-away macro register is q because it's easy to hit qq from normal mode to start recording.



[waylonwalker.com/svim](http://waylonwalker.com/svim)  
@\_waylonwalker

# Replaying a Macro

Macros can be replayed using @ followed by the letter that you stored the macro under.

@q

Note: a common throw-away macro register is q because it's easy to hit qq from normal mode to start recording.



[waylonwalker.com/svim](http://waylonwalker.com/svim)  
@\_waylonwalker

# Registers

Registers are nothing more than a single character key mapping to a value of some text. As you yank, delete, or create macros in vim, it automatically stores text into these registers.

`:reg`

List the contents of all your registers with `:reg`



[waylonwalker.com/svim](http://waylonwalker.com/svim)  
[@\\_waylonwalker](https://twitter.com/_waylonwalker)

# making a macro into a shortcut

The magical shortcut that makes it easy is that control + r <C-R> followed by a register will paste that register wherever you currently are, including the command mode.

```
:nnoremap {binding} <C-R>{register}
```



[waylonwalker.com/svim](http://waylonwalker.com/svim)  
@\_waylonwalker



# Editing a Macro

Now that we understand that macros are simply strings of text placed into a register, it becomes a bit more intuitive to edit them after being created.

```
<C-R>q
```

Then edit the macro and add it back to that buffer and delete it.

```
"qdd
```



[@\\_waylonwalker](http://waylonwalker.com/svim)

# recap

```
" record a macro
q{register}

" play a macro
@{register}

" list registers
:reg

" map a macro to a keyboard shortcut
:noremap {binding} <C-R>{register}

" edit a macro
<C-R>{register}
"{register}dd

" make a macro recursive
<C-R>{register}A@q<esc>"{register}dd
```



[waylonwalker.com/svim](http://waylonwalker.com/svim)  
@\_waylonwalker



# full POST

[waylonwalker.com/svim](http://waylonwalker.com/svim)



@\_waylonwalker