Bram's snippets

Release -

April 21, 2016

Contents

http://snippets.bram.xyz/

Chrome snippets

CHAPTER 1

Google Maps: fast routes from a fixed point

Whenever I want to find the route to somewhere, say to Cherry Red's in Birmingham, I go to the address bar in Chrome and type

```
to Cherry Red's <enter>
```

This brings up the route from my university:

> https://www.google.co.uk/maps/search/from:+university+of+birmingham+to:+Cherry%20Red's

To get this, open chrome://settings/searchEngines (you'll have to copy-paste that link), and fill in a new entry:

> **New engine** *Route to*
>
> **Keyword** `to`
>
> **URL** `https://www.google.co.uk/maps/search/from:+university+of+birmingham+to:+%s`

To use your own starting place, replace `university+of+birmingham` with something of your choice without any special characters; replace spaces by plusses.

Shell snippets

CHAPTER 2

hlocate.fish: find in home directory

`locate` will not work if your home directory is encrypted. I use this instead:

```
function hlocate
        find ~  | grep -e $argv[1]  | sed "s|^$HOME|\~|"
end
```

Now:

```
bgeron@machine ~> hlocate test1234
~/tmp/test1234.txt
bgeron@machine ~>
```

This is a Fish script, but it should be easy to convert to bash syntax.

Birmingham computer science specific snippets

CHAPTER 3

How I set up my school printers

date 11 May 2015

I use Ubuntu 14.04 on my personal laptop at the moment. Here is how I add the printers.

1. Go to *System Settings*, then *Printers*, or open *Printers* from the Launcher.
2. Click Add.
3. Choose the printer you want from the printer list. Say you choose `dali`.
4. In the *New Printer* dialog, click *Network Printer* → *Windows Printer via SAMBA*. Add a printer from the URL `smb://socs-ad/sms-2.cs.bham.ac.uk/dali`.

   Use *Set authentication details now*, using your School username + password. Do not try *Verify...*, it will not work, but printing will work.
5. In the driver selection dialog, choose the recommended driver, and use the options recommended by the School. You can find these by clicking on the printer types in the printer list.
6. For both the printer name and description, use the printer name like `dali`. Copy the room number from the printer list to the *Location* field if you want.
7. Print a test page.

If you `ssh` to a School server, you can test if the printing worked, by running e.g. `lpq -P dali` repeatedly. Or run it in a loop with `watch -n .5 lpq -P dali`. Or, of course, you can actually walk to the printer.

The printers I add like this work on my personal laptop, but only if I connect from the wired network. The wireless network is separate in some ways and printing won't work; you might have luck using the VPN, or you can `scp` your PDFs to a School machine and `lpr` with the correct options from there. The `scp` approach works, but it's really not as handy as just setting up printers the normal way.

This guide is based on the School guide for printing on self maintained Windows PCs.

Other snippets

CHAPTER 4

LaTeX snippet: common-unicode.sty

If you would like to type Unicode in your `.tex` files, check out this Gist:

https://gist.github.com/bgeron/43ba383bc024791c245ce9abeb1b43b7

CHAPTER 5

Sphinx extension: skip.py

This extension gives you a `skip` directive that adds some vertical space.

```
r"""
Skip directive, by Bram Geron.

Adds a bit of vertical space.

Usage:

    .. skip:: big

This adds a \bigskip in the LaTeX, and a vertical space in the HTML. By HTML
standards, the space is not very big.
"""

from docutils import nodes
from docutils.parsers.rst import Directive
import docutils.parsers.rst.directives as directives
from sphinx.errors import SphinxError

def setup(app):
    app.add_node(skip,
                 html=(visit_skip_html, depart_skip_html),
                 latex=(visit_skip_tex, depart_skip_tex))
    app.add_directive('skip', SkipDirective)

    return {
        'version': "0.1",
        'parallel_read_safe': True,
        'parallel_write_safe': True,
    }

class skip(nodes.General, nodes.Element):
    pass

class SkipError(SphinxError):
    pass

class SkipDirective(Directive):

    has_content = True

    option_spec = {
```

```
    }

    def run(self):
        if self.content.data == []:
            raise self.error("missing skip size")
        elif not isinstance(self.content.data, list) or len(self.content.data) != 1:
            raise self.error("skip size decl not recognised: " + repr(self.content.data))

        size = self.content.data[0]

        if size in ['para', 'big', 'xxlarge']:
            return [skip(size=size)]
        else:
            raise self.error("skip size not recognised: " + repr(size))



def visit_skip_tex(self, node):
    if node['size'] == 'para':
        self.body.append("\n\n")
    elif node['size'] == 'big':
        self.body.append("\n\n\\bigskip\n\n")
    elif node['size'] == 'xxlarge':
        self.body.append("\n\n\\vfill\n\n")
    else:
        raise SkipError("cannot render skip size %r to latex"
            % (node['size'],))

def depart_skip_tex(self, node):
    pass

def visit_skip_html(self, node):
    if node['size'] in ['para']:
        pass # HTML already breaks enough paragraphs
    elif node['size'] in ['big', 'xxlarge']:

        self.body.append(
            self.starttag(node, 'div', **{'class': 'skip-%s' % (node['size'],)}))
        self.body.append("</div>")

    else:
        raise SkipError("cannot render skip size %s to HTML"
            % (node['size'],))


def depart_skip_html(self, node):
    pass
```

Others' scripts and programs I like and fanboy about

- Ack: like *grep* but with good defaults for programmers. Ignores version control and build directories; trivial to customise.

  If you use Ubuntu, `apt install ack-grep` *and define an* `alias ack=ack-grep` *in your shell profile.*

- Fish, the friendly interactive shell. Has better syntax, great defaults (syntax highlighting!), and better tab-completion (substrings, not just prefixes of filenames). Defining and saving an alias is as simple as this:

```
alias s sublime_text
funcsave s
```

  This persists the newly created function `s` to `~/.config/fish/functions/s.fish`, and it will automatically be loaded in all currently-running and future shells! No more manual fiddling around with `~/.profile`.

CHAPTER 6

Contact

Email: second letter of the alphabet @ my first name .xyz

Twitter: @bgeron