Installing and Running decksh/pdfdeck

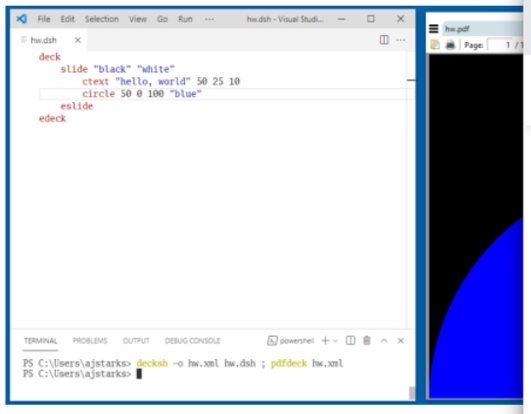
Installing

f.pdf - 12/17 (72 dpi)

Running decksh: Windows

editor: VSCode

viewer: Sumatra PDF



terminal: VSCode

install-run.dsh - Visual Studio Code

File Edit Selection View Go Run Terminal Help

install-run.dsh U

```
    eslide

    slide bgcolor fgcolor
        ctext "Running decksh: Mac OS" mid top ts
        image "images/mac-decksh03.png" 50 45 2909 1102 30
        ctext "editor: VSCode"      25 75 capsize capfont
        ctext "terminal: VSCode"    25 10 capsize capfont
        ctext "viewer: Preview"     75 75 capsize capfont
    eslide

    slide bgcolor fgcolor
        ctext "Running decksh: Windows" mid top ts
        image "images/win-decksh00.png" 50 45 1356 616 55
        ctext "editor: VSCode"          25 75 capsize capfont
        ctext "terminal: VSCode"        25 10 capsize capfont
        ctext "viewer: Sumatra PDF"     75 75 capsize capfont
    eslide
```

TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE bash

```
$ dr install-run.dsh -pagesize 900,600 -sans GillSans -mono Inconsolata-Medium -serif Charter-Italic
```

making this guide

Installing using go and git

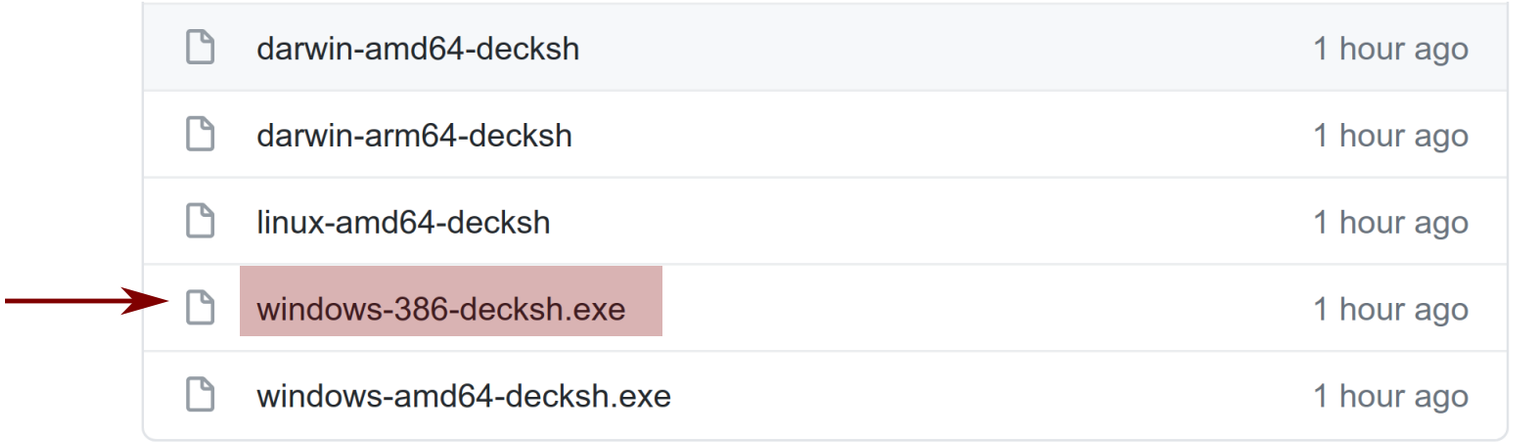
> Install the latest versions of decksh and pdfdeck,
> download the deckfonts in $HOME/deckfonts.
> Test.

```
$ go install github.com/ajstarks/decksh/cmd/decksh@latest
...
$ go install github.com/ajstarks/deck/cmd/pdfdeck@latest
...
$ cd $HOME
$ git clone https://github.com/ajstarks/deckfonts
...
$ decksh -? ; pdfdeck -?
```

Installing decksh and pdfdeck binaries

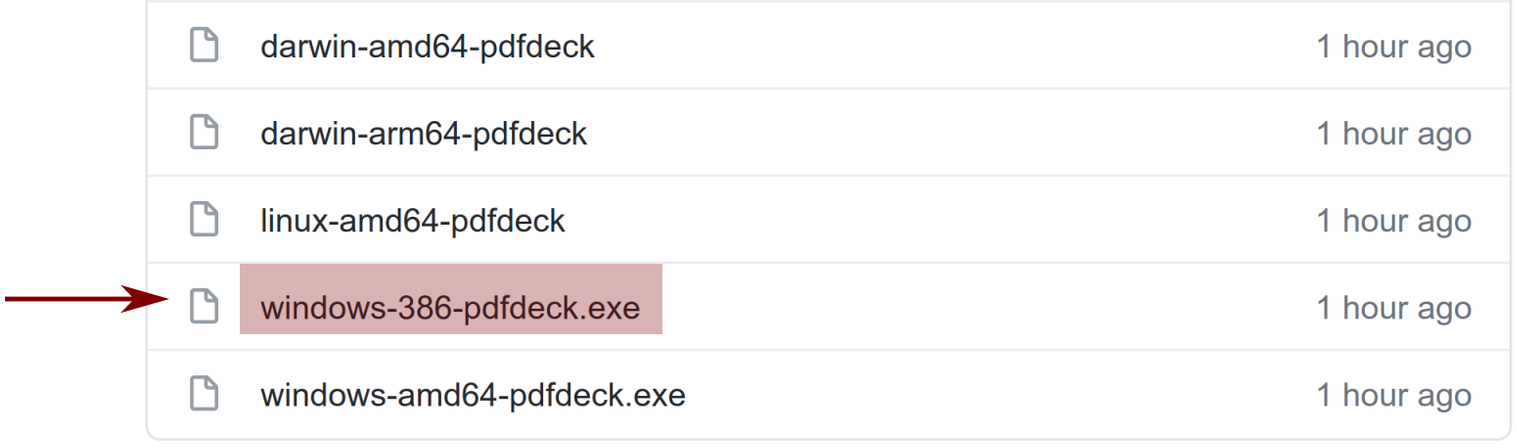
Mac (M1, Intel), Linux (Intel), Windows (32, 64-bit)

Pick your type, download, rename to 'decksh', place where your apps live.



https://github.com/ajstarks/decksh/tree/master/cmd/decksh/binaries

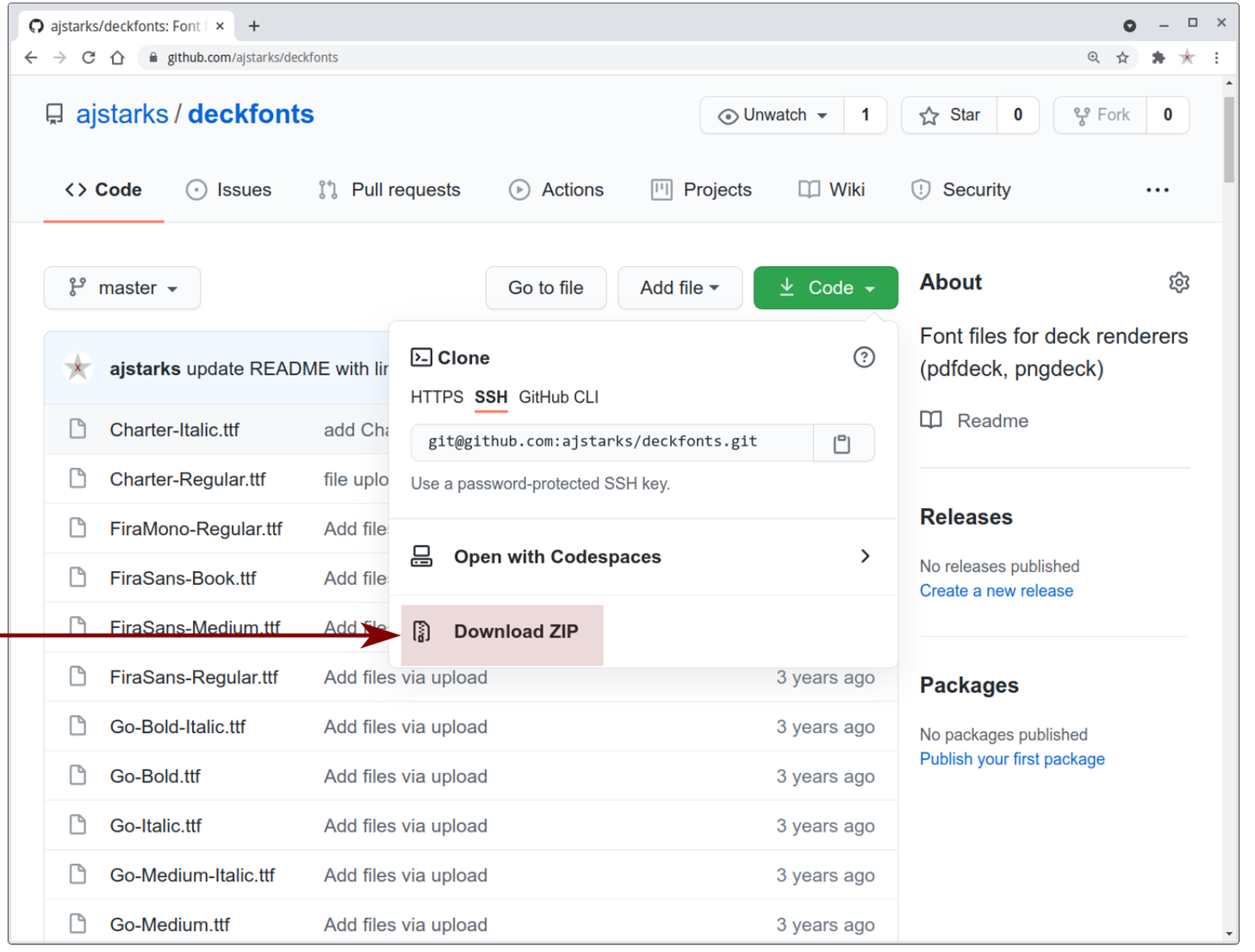
Pick your type, download, rename to 'pdfdeck', place where your apps live.



https://github.com/ajstarks/deck/tree/master/cmd/pdfdeck/binaries

Downloading the fonts

Download, unzip to your home directory.



Default Fonts

Times Roman

times
timesi
timesb
timesbi

Hamburgevons

Helvetica

helvetica
helveticai
helveticab
helveticabi

Hamburgevons

Courier

courier
courieri
courierb
courierbi

Hamburgevons

Alternative serif, sans, mono

Charter

Charter-Regular
Charter-Italic

Fira Sans

FiraSans-Book
FiraSans-Medium
FiraSans-Regular

Inconsolata

Inconsolata-Regular
Inconsolata-Bold
Inconsolata-Medium
Inconsolata-Condensed
Inconsolata-Black

Sans fonts

IBM Plex

IBMPlexSans-Regular
IBMPlexMono-Regular

Hamburgevons

Noto Sans

NotoSans-Regular
NotoMono-Regular

Hamburgevons

Public Sans

PublicSans-Italic
PublicSans-Light
PublicSans-Medium
PublicSans-Regular
PublicSans-SemiBold
PublicSans-BoldItalic
PublicSans-Bold

Hamburgevons

Symbol fonts

Zapf Dingbats

zapfdingbats

Gophers

Gophers

State Face

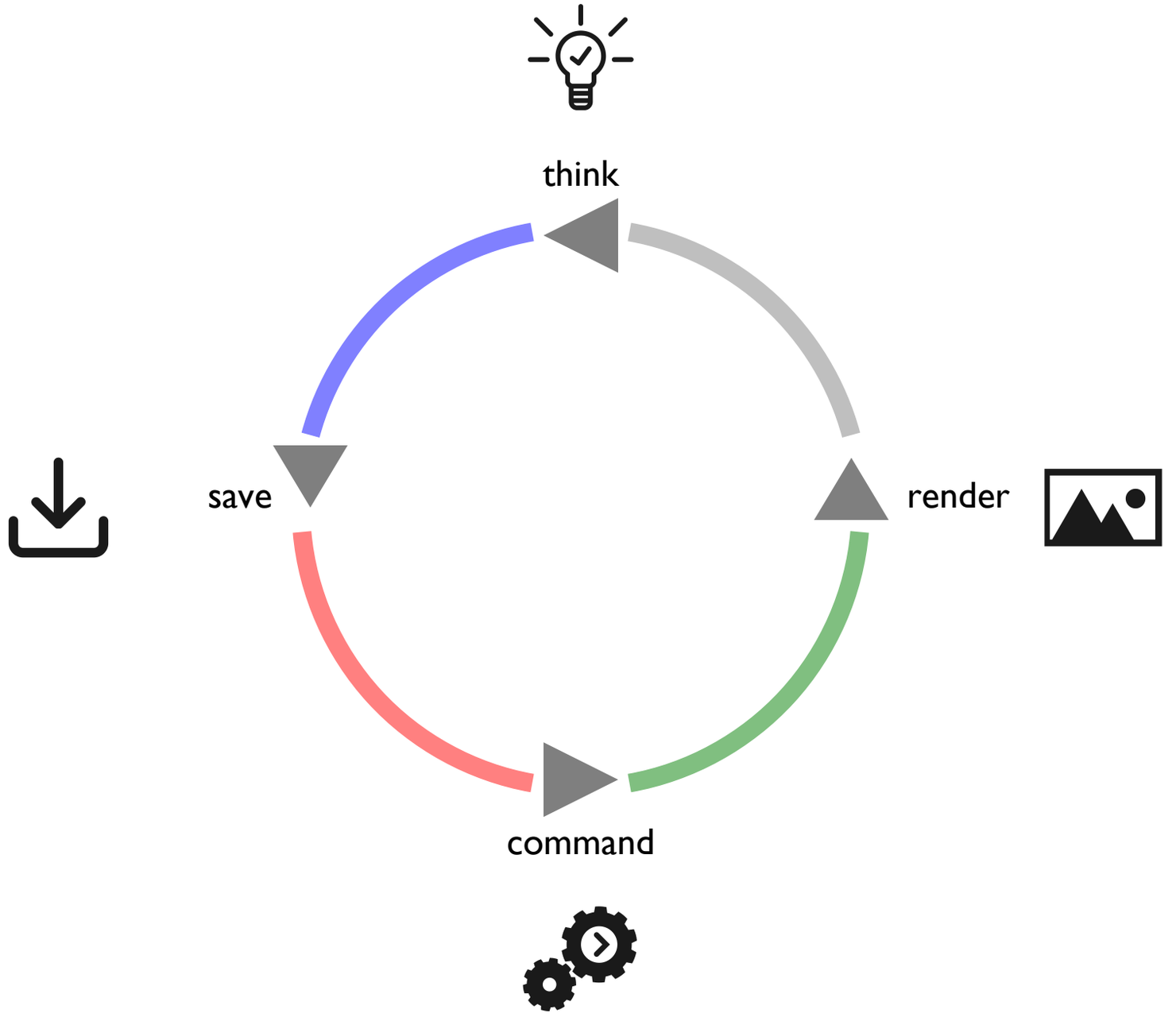
stateface

Wee People

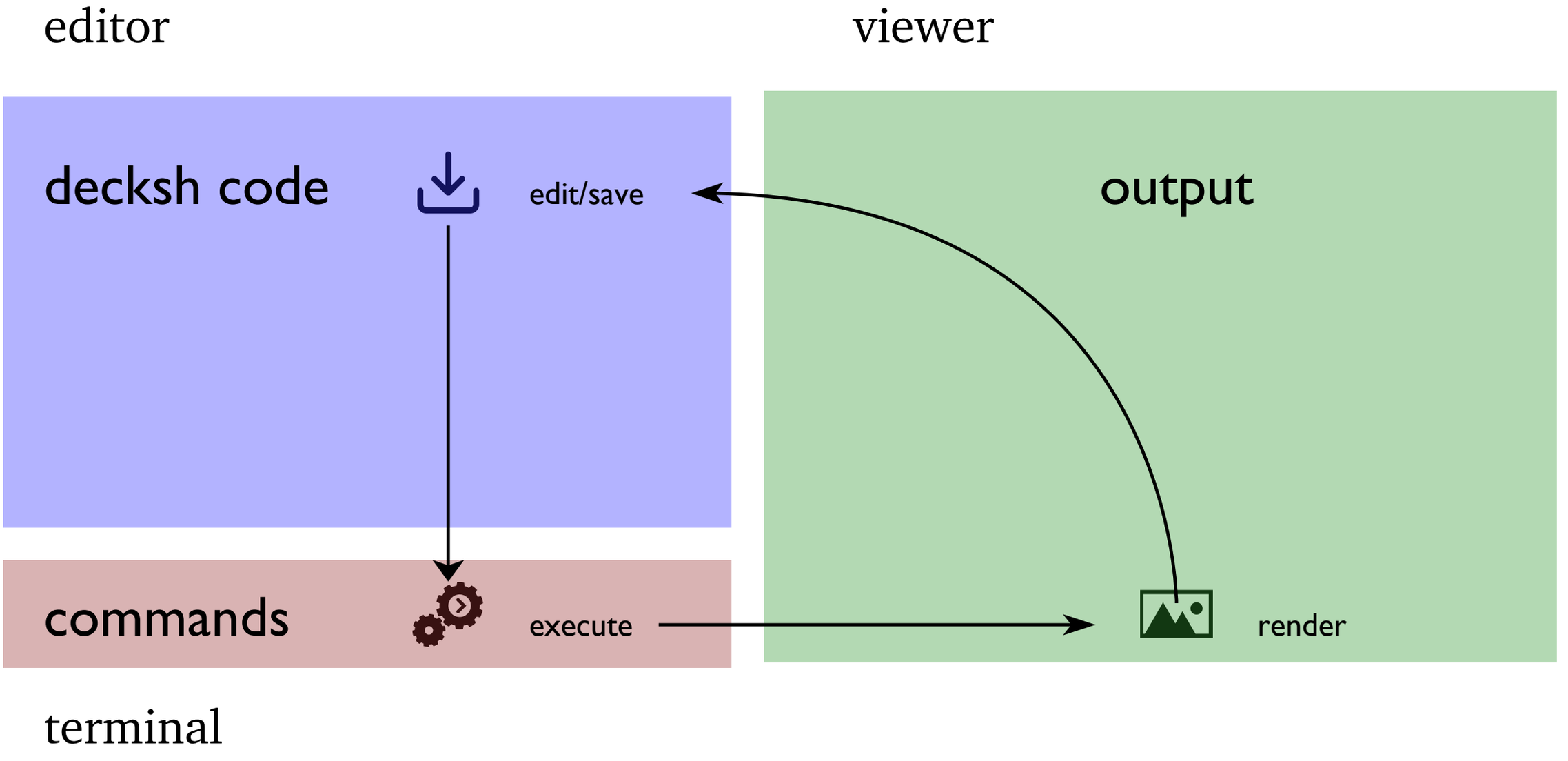
weepeople

Running

workflow



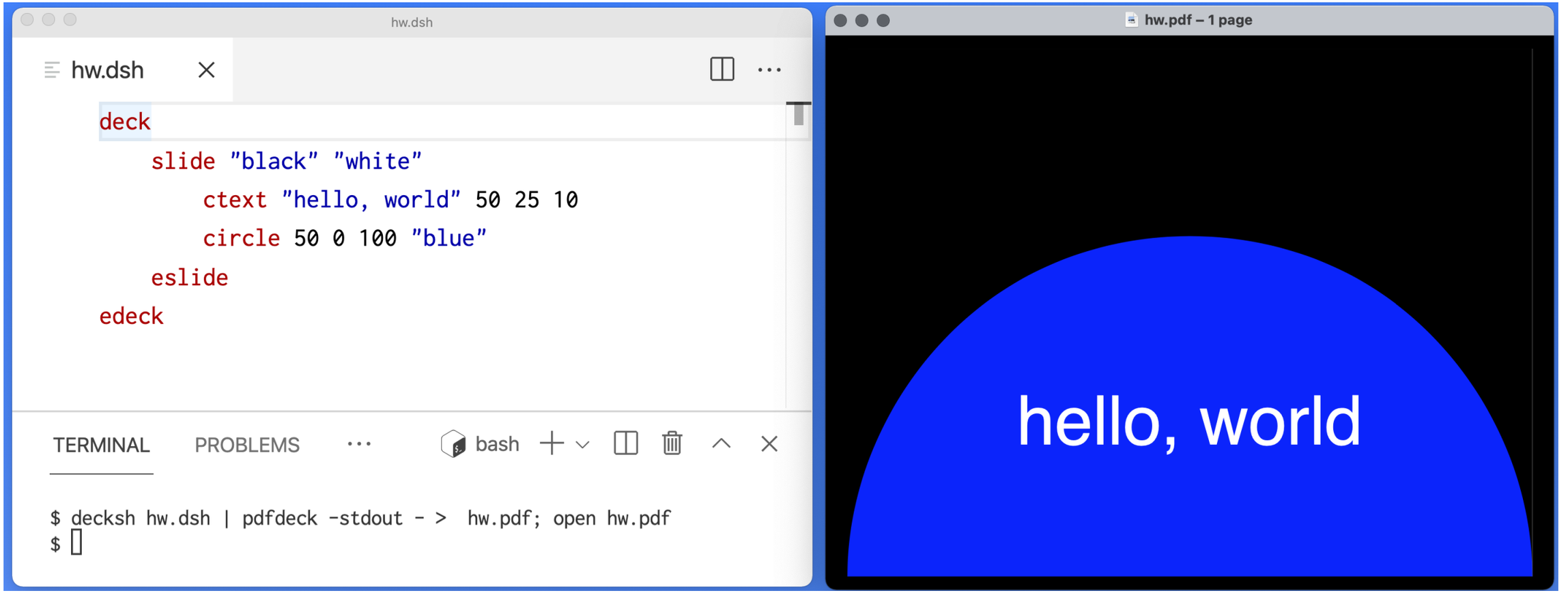
Working setup



Mac OS

editor: VSCode

viewer: Preview

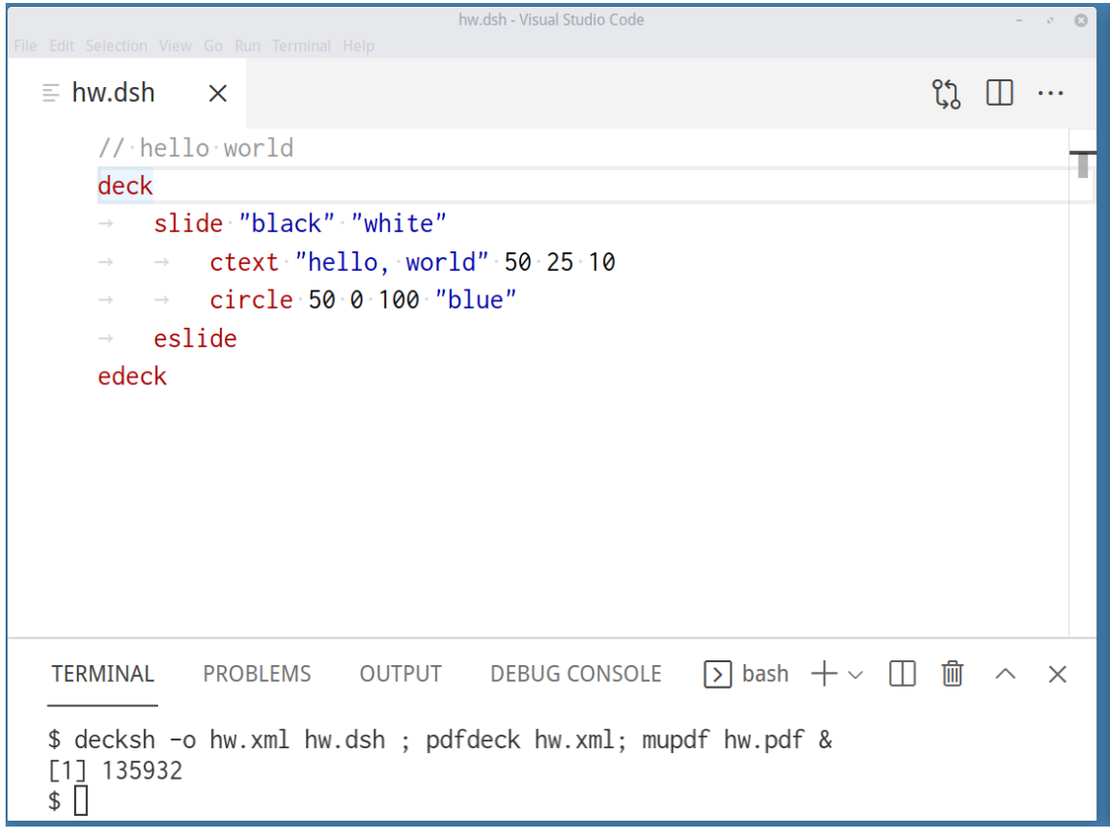


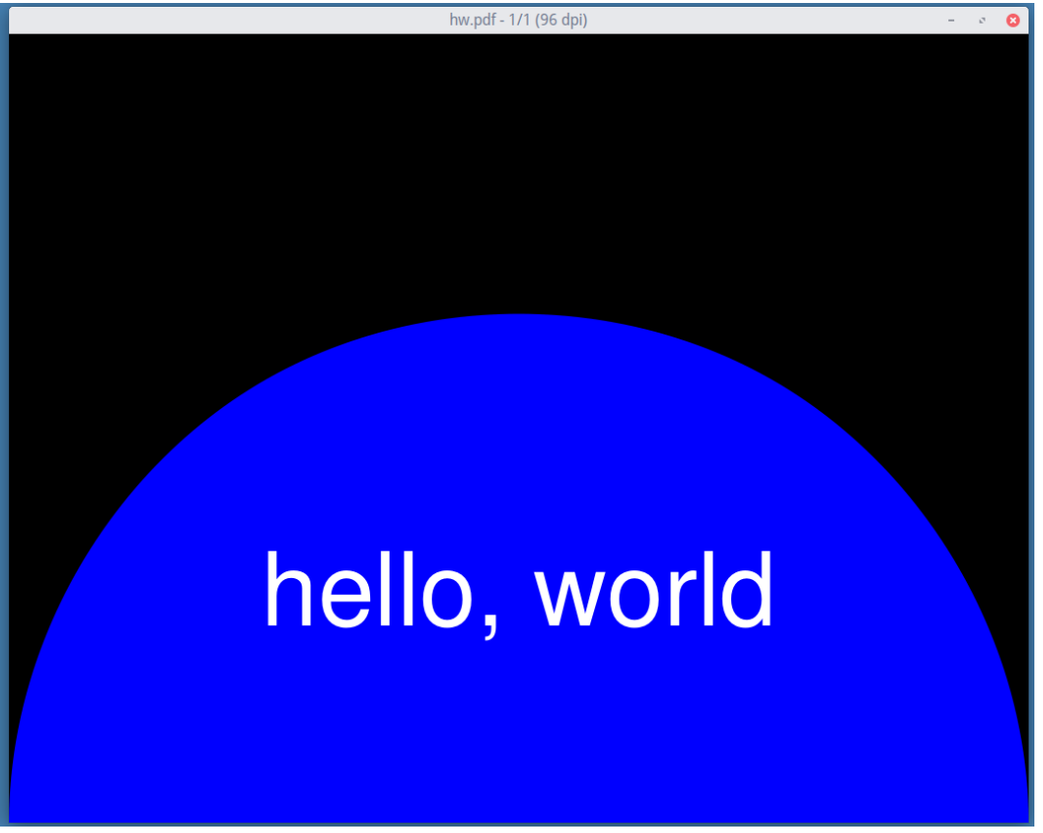
terminal: VSCode

Linux

editor: VSCode

viewer: mupdf



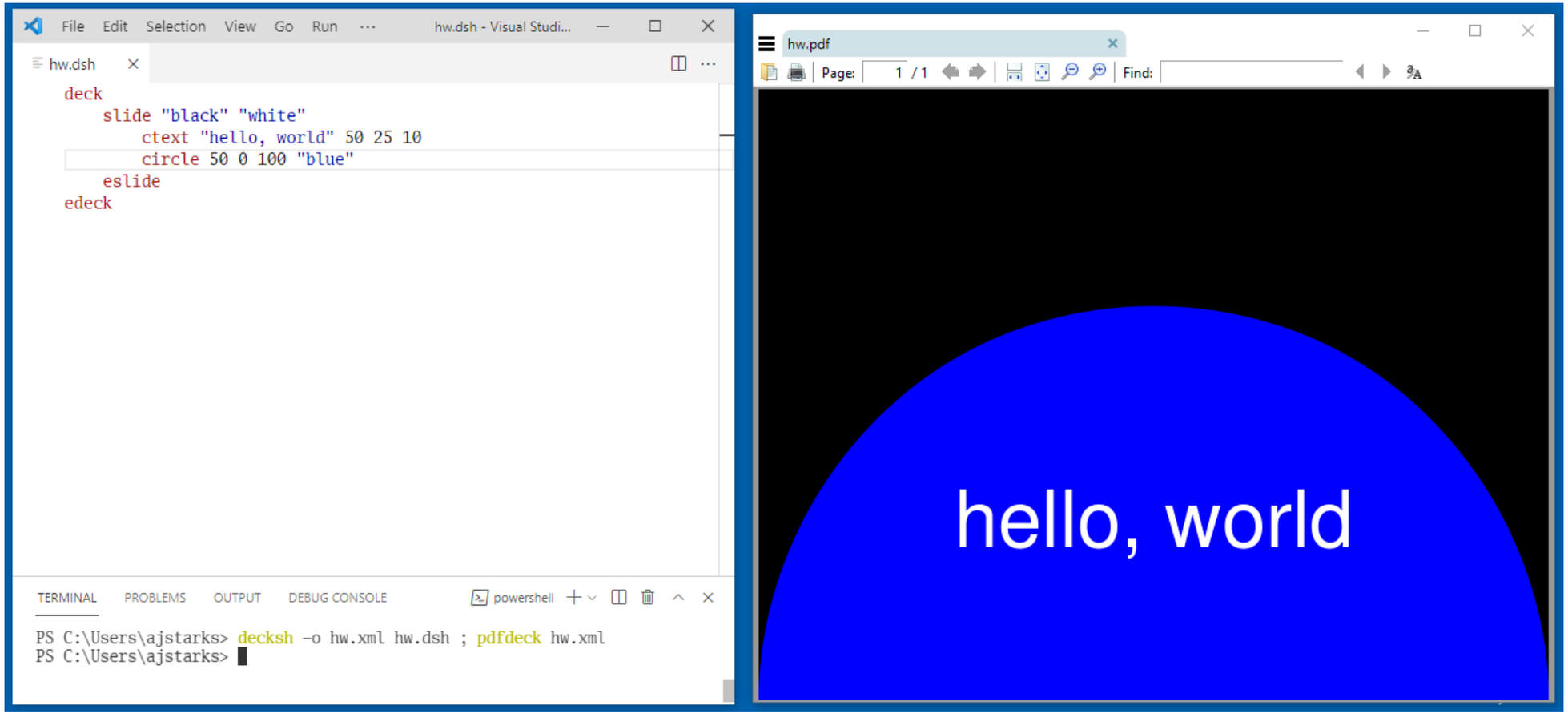


terminal: VSCode

Windows

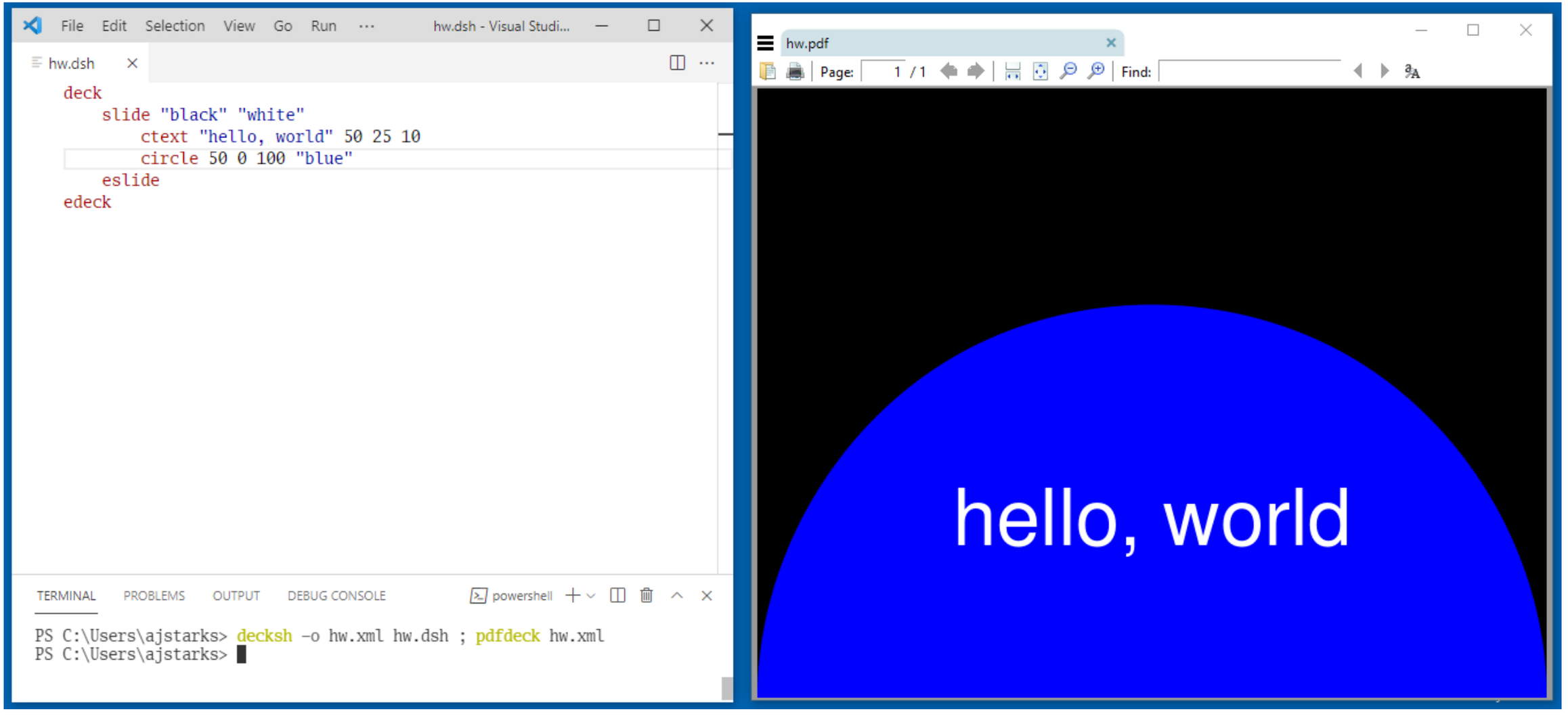
editor: VSCode

viewer: Sumatra PDF

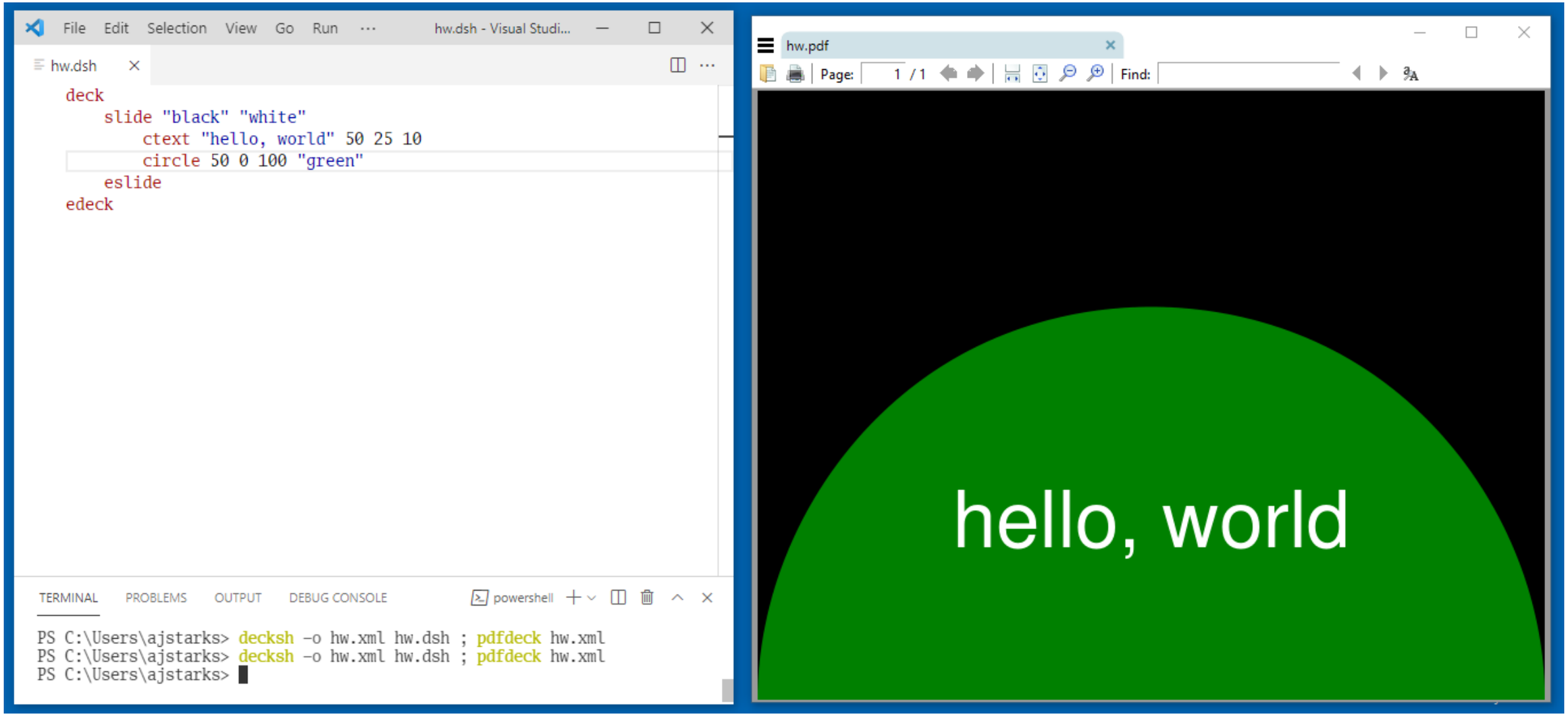


terminal: VSCode

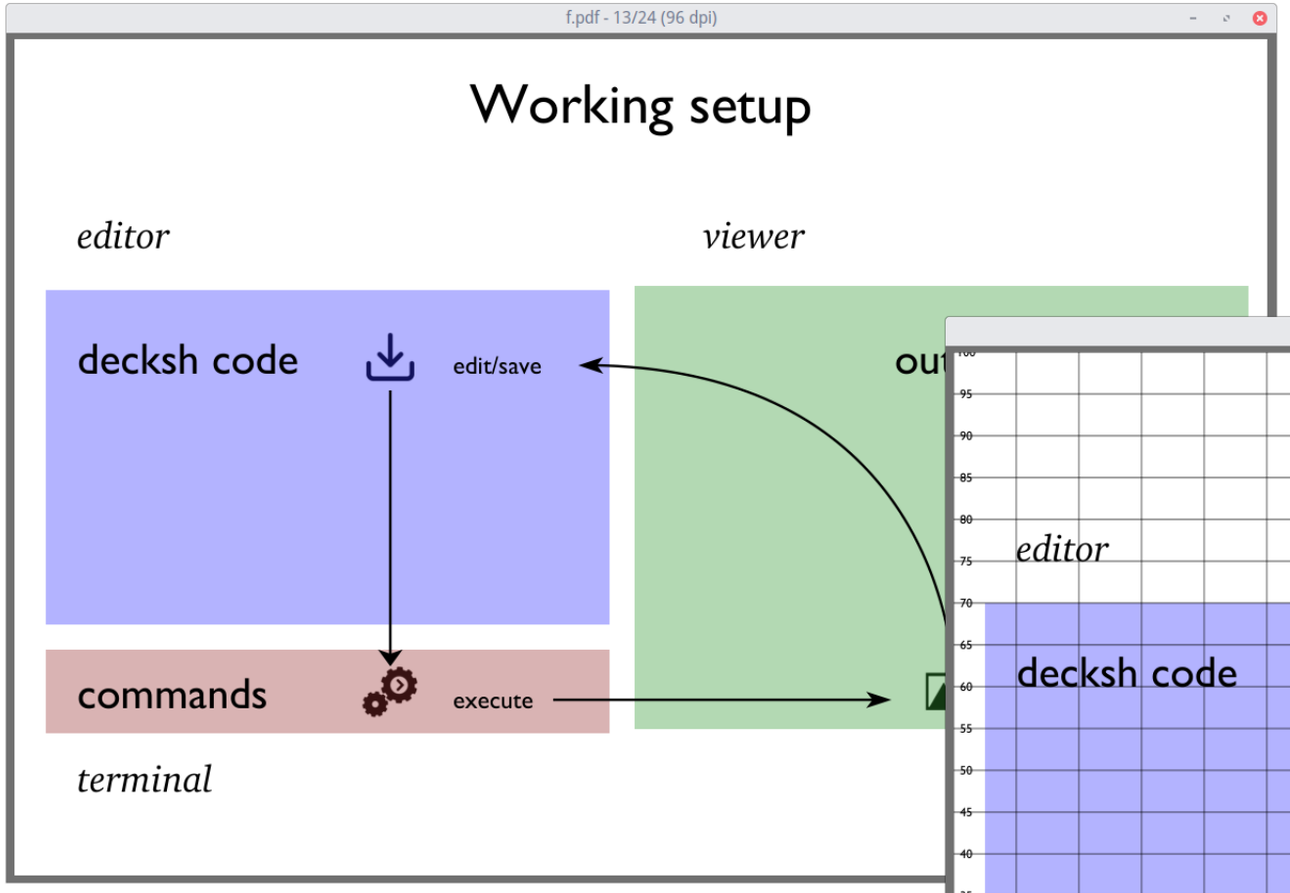
Render


File Edit Selection View Go Run ··· hw.dsh - Visual Studi...
hw.dsh
deck
slide "black" "white"
ctext "hello, world" 50 25 10
circle 50 0 100 "blue"
eslide
edeck
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE powershell
PS C:\Users\ajstarks> decksh -o hw.xml hw.dsh ; pdfdeck hw.xml
PS C:\Users\ajstarks>
hw.pdf
Page: 1 / 1
Find:
hello, world

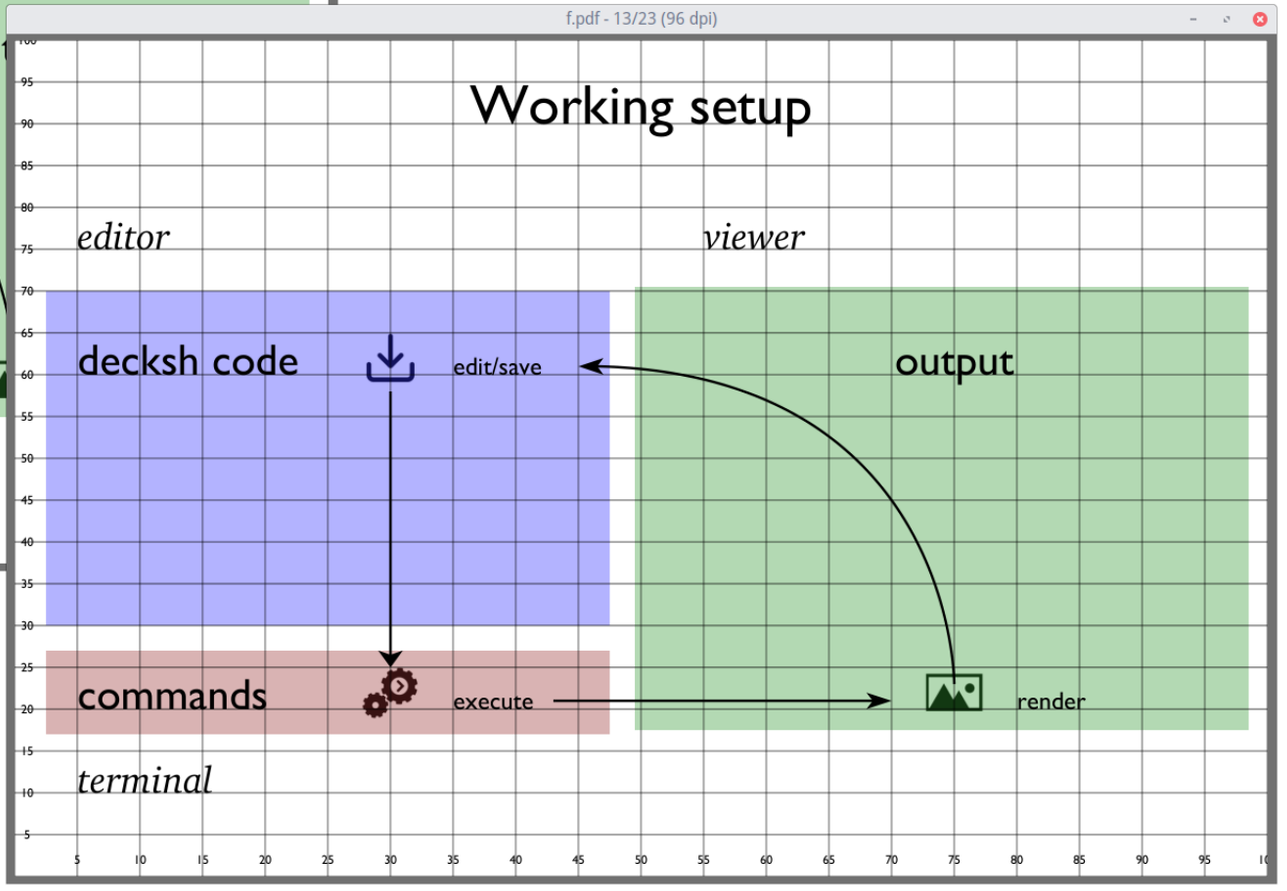
Update


File Edit Selection View Go Run ··· hw.dsh - Visual Studi...
hw.dsh
deck
slide "black" "white"
ctext "hello, world" 50 25 10
circle 50 0 100 "green"
eslide
edeck
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE powershell
PS C:\Users\ajstarks> decksh -o hw.xml hw.dsh ; pdfdeck hw.xml
PS C:\Users\ajstarks> decksh -o hw.xml hw.dsh ; pdfdeck hw.xml
PS C:\Users\ajstarks>
hw.pdf
Page: 1 / 1
Find:
hello, world

Using the -grid option



no grid



-grid 5

The command line

decksh command usage

`decksh`	read from stdin, write to stdout
`decksh in.dsh`	read from file, write to stdout
`decksh -o out.xml`	read from stdin, write to file
`decksh -o out.xml in.dsh`	read from file, write to file

```
decksh example.dsh | pdfdeck ...
```

pdfdeck [options] inputfile

Option	Default	Description
`-sans`	helvetica	Sans Serif font
`-serif`	times	Serif font
`-mono`	courier	Monospace font
`-symbol`	zapfdingbats	Symbol font
`-pages`	1-1000000	Pages to output (first-last)
`-pagesize`	Letter	Page size (w,h or Legal, Tabloid, A[3-5], ArchA, 4R, Index)
`-grid`	false	Draw a percent grid
`-fontdir`	$HOME/deckfonts	Font directory
`-outdir`	Current directory	Output directory
`-stdout`	false	Output to standard output
`-author`	""	Document author
`-title`	""	Document title

command examples

```
decksh -o file.xml file.dsh; pdfdeck file.xml
```
process file.dsh to file.xml to file.pdf

```
decksh file.dsh | pdfdeck -stdout - > output.pdf
```
Pipe the output from decksh, making output.pdf

```
pdfdeck -pagesize 1920,1080 -pages 10-20 -grid 5 file.xml
```
render pages 10-20 to file.pdf, page size of 1920 (width) x 1080 (height) pixels, on a 5% grid

```
pdfdeck -sans FiraSans-Regular -serif Charter-Regular -mono Inconsolata-Bold file.xml
```
use FiraSans-Regular.ttf, Charter-Regular.ttf, and Inconsolata-Bold.ttf from the deckfonts directory

```
echo file.dsh|entr -s 'decksh file.dsh|pdfdeck -stdout - > f.pdf; pkill -HUP mupdf'
```
When file.dsh changes, make f.pdf, and refresh the viewer