



Nelsie

Framework for Creating Slides

Ada Böhm
ada@kreatrix.org

Hello world example

```
from nelsie import SlideDeck

deck = SlideDeck()

@deck.slide()
def hello_world(slide):
    slide.text("Hello world!")

deck.render("slides.pdf")
```

Nelsie supports ...

Nelsie supports ...
... fragment ...

Nelsie supports ...
... fragment ...
... revealing.

Simple mechanism for complex slide changes



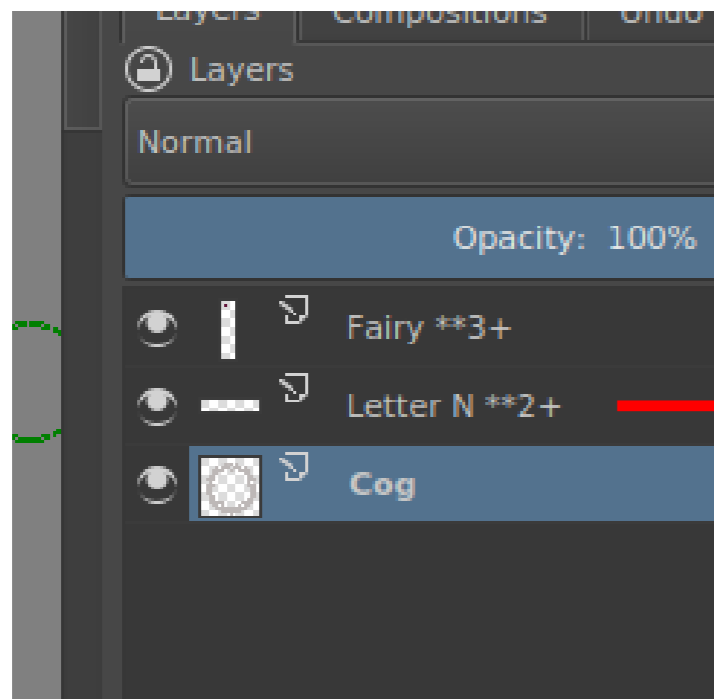
Simple mechanism for complex slide changes



Simple mechanism for complex slide changes

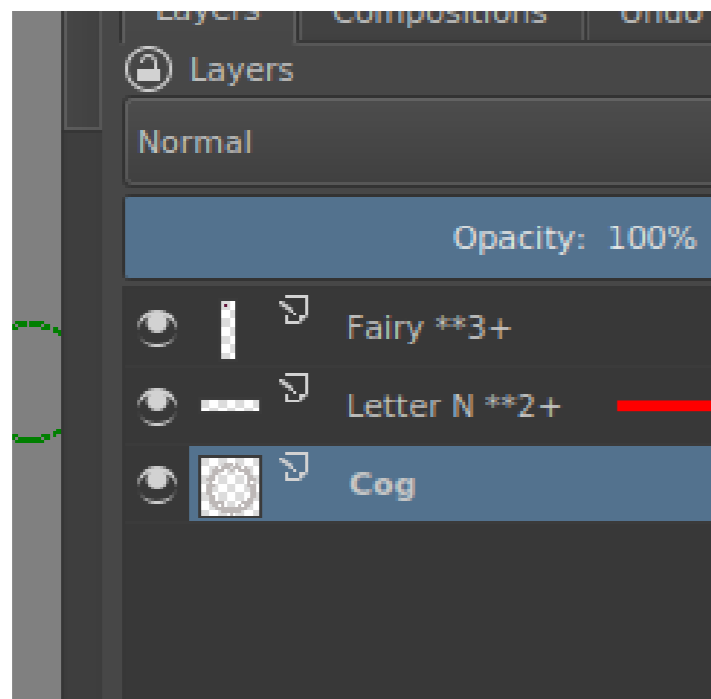


Name-driven revealing layers in SVG and ORA images



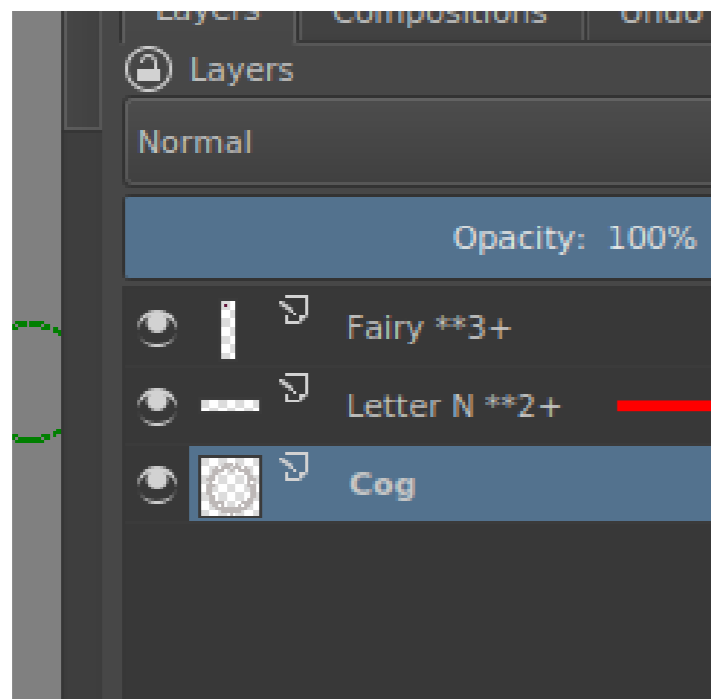
Layer is shown from the second step

Name-driven revealing layers in SVG and ORA images



Layer is shown from the second step

Name-driven revealing layers in SVG and ORA images



Layer is shown from the second step

Headers & Titles

Syntax highlighting

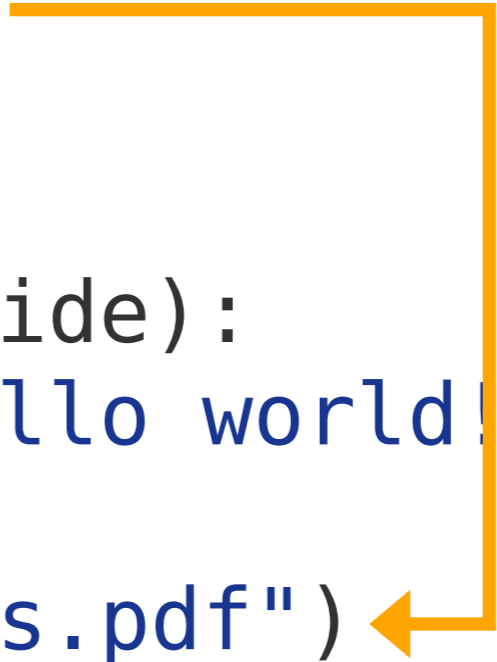
```
# This is comment  
from nelsie import SlideDeck  
  
deck = SlideDeck()  
  
@deck.slide()  
def hello_world(slide):  
    slide.text("Hello world!")  
  
deck.render("slides.pdf")
```

Line highlighting

```
# This is comment  
from nelsie import SlideDeck  
  
deck = SlideDeck()  
  
@deck.slide()  
def hello_world(slide):  
    slide.text("Hello world!")  
  
deck.render("slides.pdf")
```


Pointing into text

```
# This is comment  
from nelsie import SlideDeck  
  
deck = SlideDeck()  
  
@deck.slide()  
def hello_world(slide):  
    slide.text("Hello world!")  
  
deck.render("slides.pdf")
```

An orange arrow originates from the SlideDeck() call in the deck = SlideDeck() line. It extends horizontally to the right, then vertically downwards, and finally horizontally to the left, ending with an arrowhead pointing to the render("slides.pdf") call.

Pointing into text

```
# This is comment  
from nelsie import SlideDeck  
  
deck = SlideDeck()  
  
@deck.slide()  
def hello_world(slide):  
    slide.text("Hello world!")  
  
deck.render("slides.pdf")
```



Comment

Pointing into text

```
# This is comment  
from nelsie import SlideDeck  
  
deck = SlideDeck()  
  
@deck.slide()  
def hello_world(slide):  
    slide.text("Hello world!")  
deck.render("slides.pdf")
```



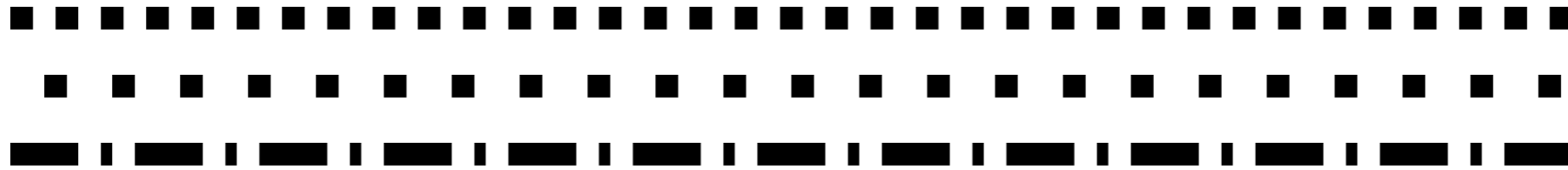
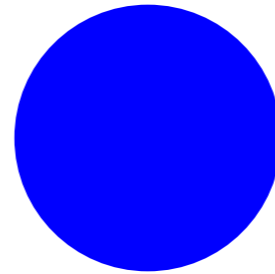
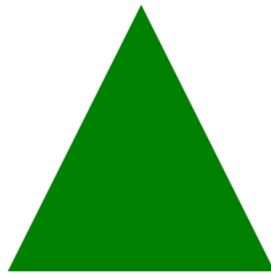
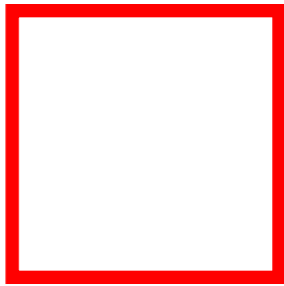
Own styles in syntax highlight

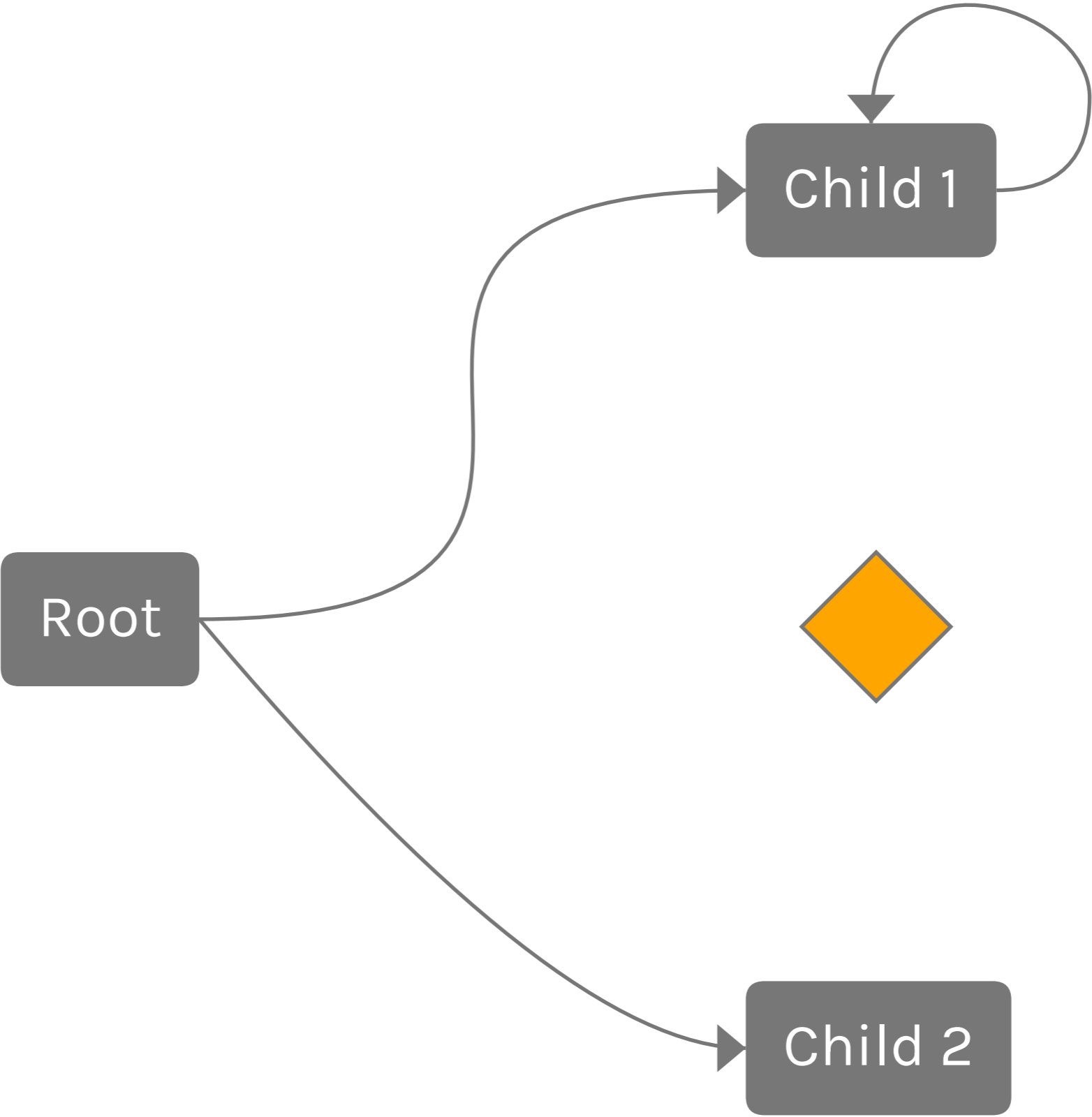
```
# This is comment  
from nelsie import SlideDeck  
  
deck = SlideDeck()  
  
@deck.slide()  
def hello_world(slide):  
    slide.text("Hello world!")  
  
deck.render("slides.pdf")
```

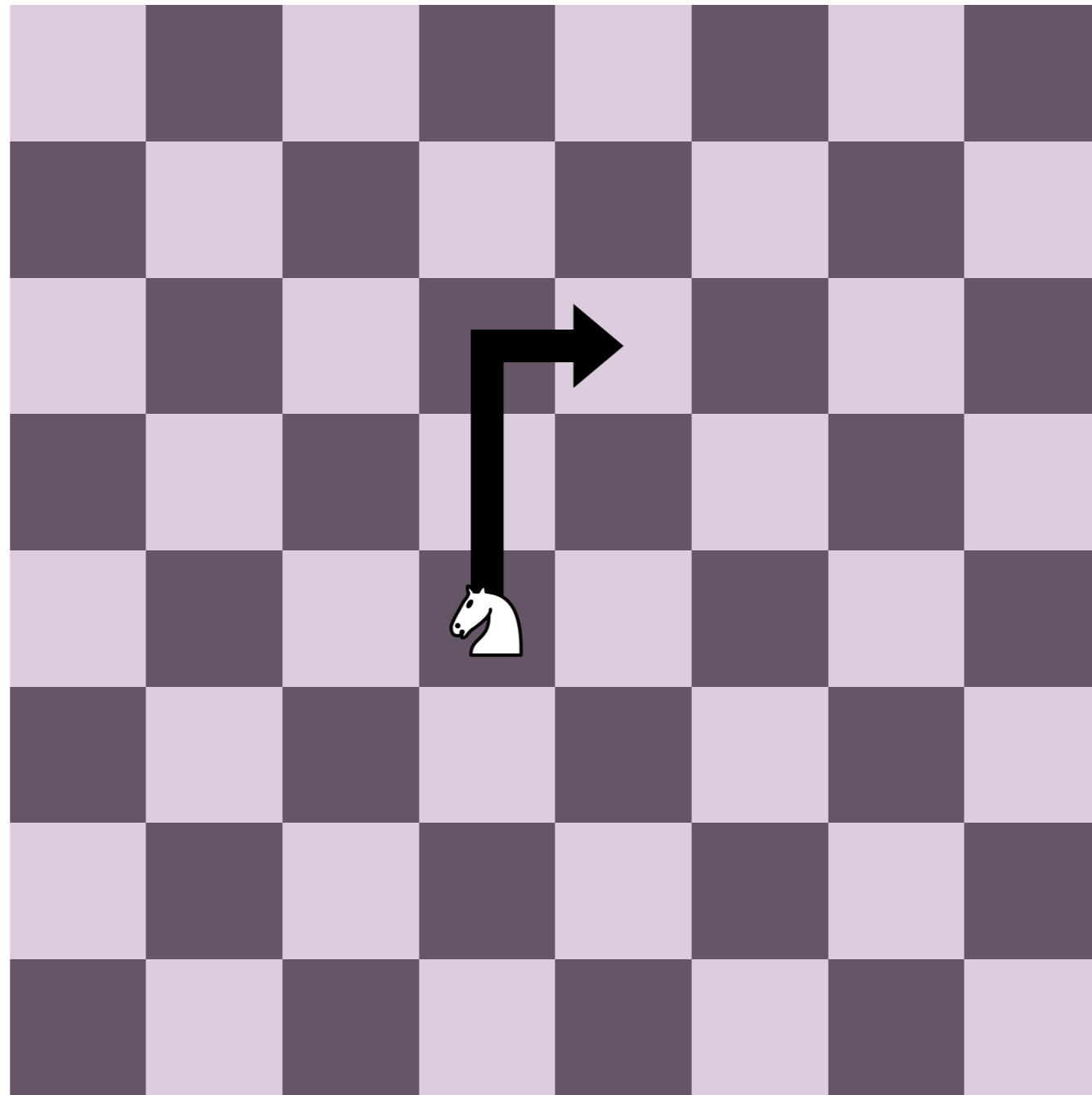
Console demo

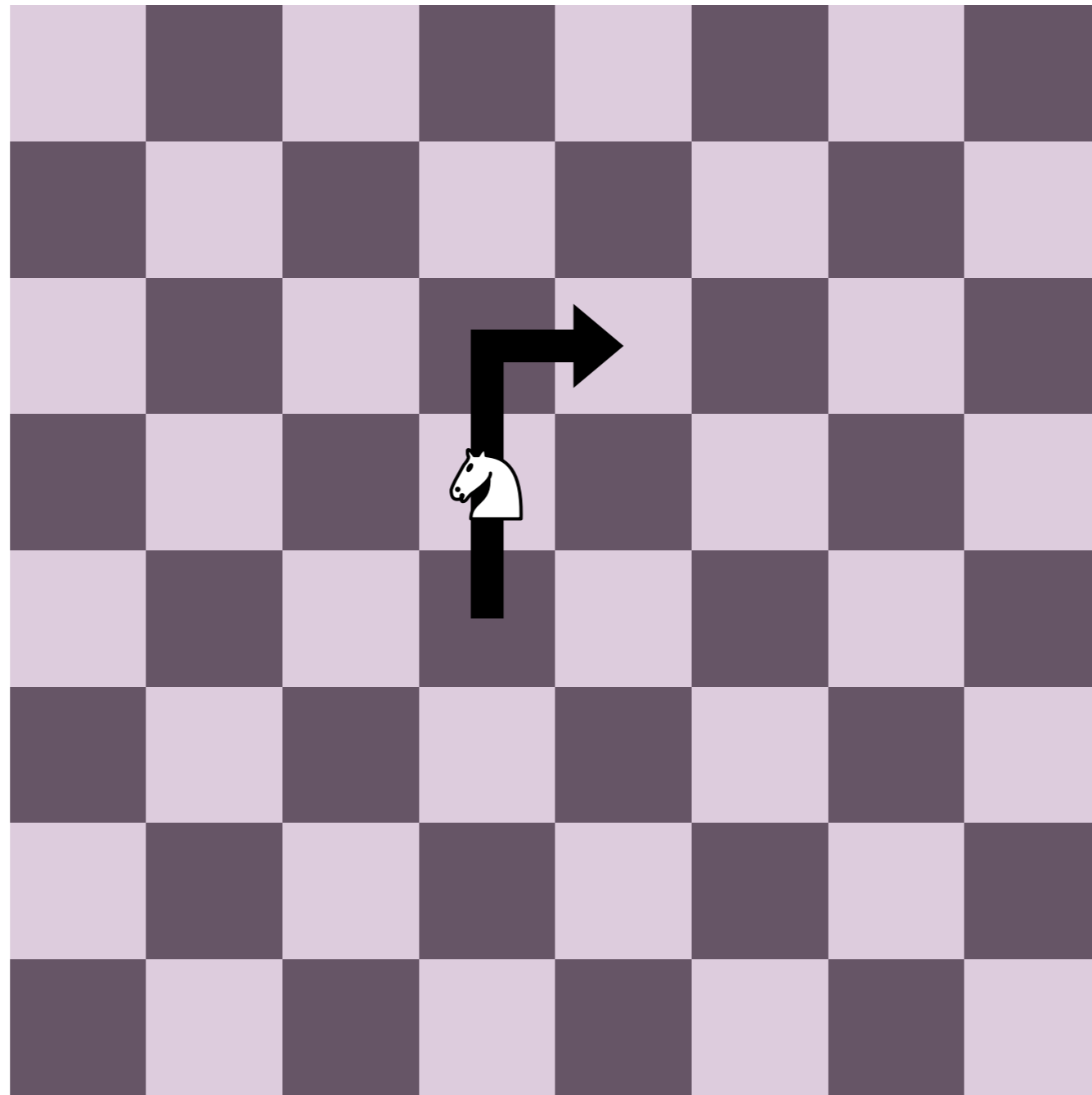
```
~/nelsie/example/bigdemo$ ls  
bigdemo.py  imgs  karla_font  
  
~/nelsie/example/bigdemo$ python3 bigdemo.py
```

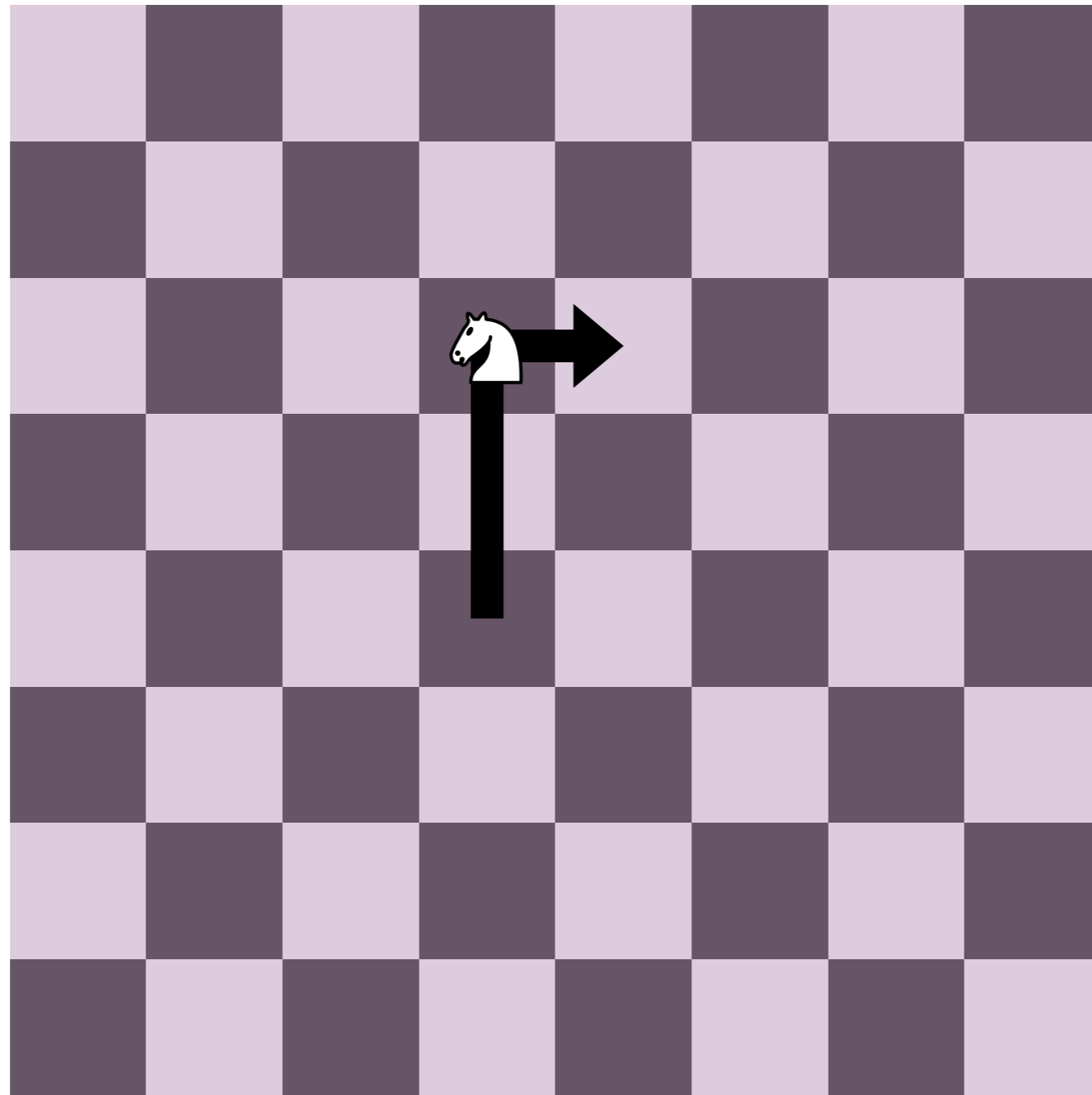
Shapes

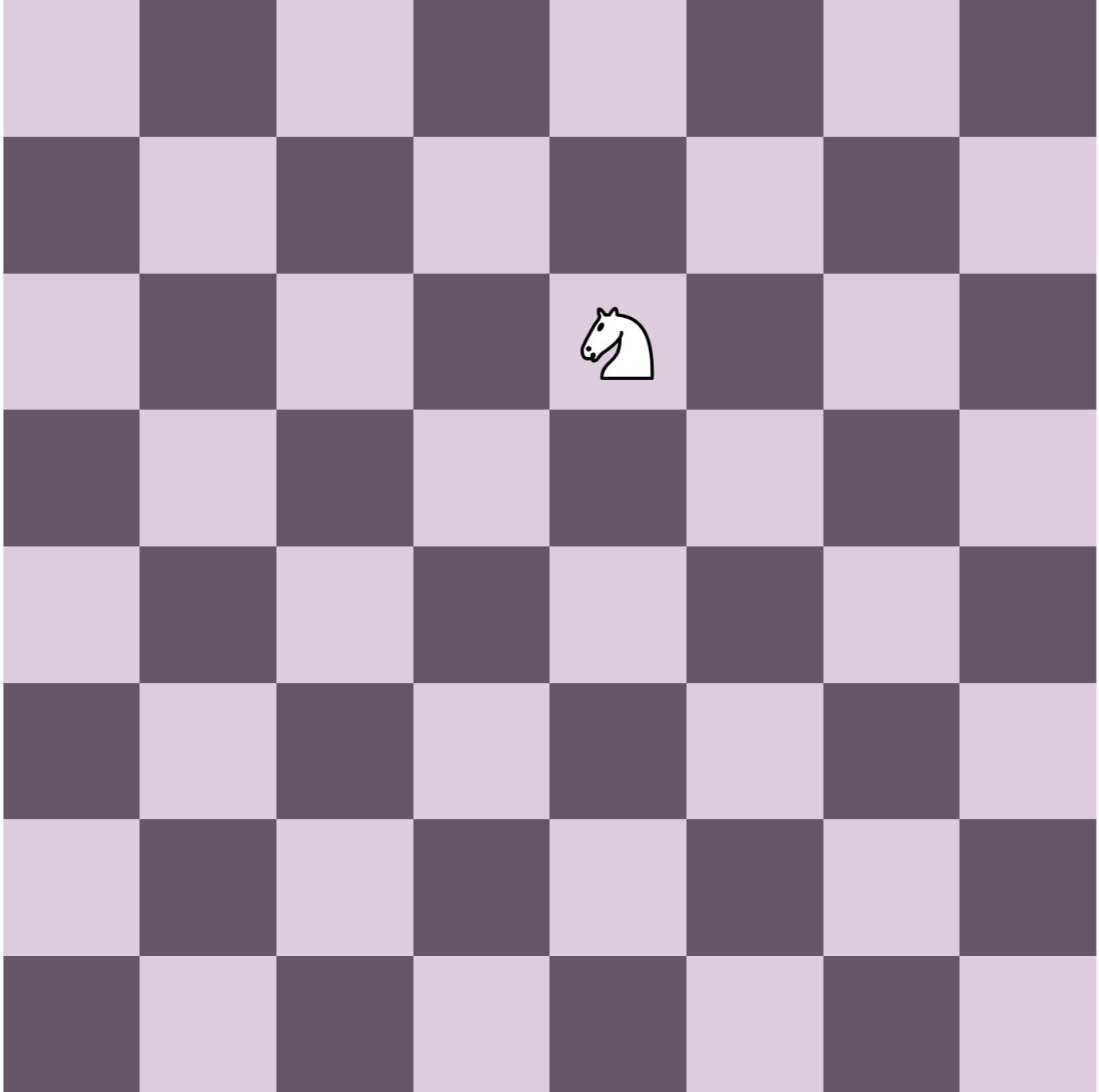














text [317.85602x140.8]

Debugging frames

title box [1024x173]

text [180.128x64]

Nelsie

text [604.494x44]

Framework for Creating Slides

text [173.39063x53.61]

Ada Böhm
ada@kreatrix.org