Challenge Walkthrough

Author: Kevin Sprague

Summary

Showing the usage for the LaTeX walkthrough template.

Walkthrough

Step 1

For inline code (like `sed -i 's/.*//g' $1`), I usually just use `\verb` or `\texttt{code}`. For relatively short codelistings, the `codebox` environment is fine:

```
import sys
def main():
    greeting = sys.argv[1] if len(sys.argv) > 1 else 'hello'
    msg = f'{greeting}, world'
    print(msg) # print the message to stdout

if __name__ == '__main__':
    main()
```

To highlight things using a color, you can use: `(*@\highlight{color}{code}@*)` within a codebox, or you can use `\highlight{color}{text}` outside of a codebox.

For longer codelistings use `codefile`

Step 2

Here's a `codefile` use:

```
#include <stdio.h>

int main(void) {
    for(int i = 0; i < 10000; ++i)
        printf("%d\n", i); //wow
    return 0;
}
```

Step 3

Inserting code, not too hard. But now let's say that we want to insert an image. Again, you need to follow the same rules as with codefile or codebox with regard to line spacing, but you can use the `\image` command to insert an image. The first argument is optional, but the second is mandatory. LaTeX is backwards. The result of calling image without the first argument is:

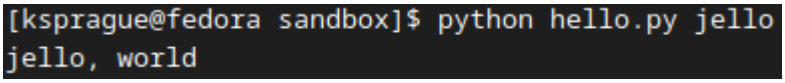

Figure 1

while the result of calling it with the first argument will be:

```
[ksprague@fedora sandbox]$ python hello.py
hello, world
```

Figure 2: You can caption an image by using the first argument to the image command.

There's probably a good argument for using `\footnote` for images, in addition to captions. That would work in the following way:

`\image[<caption goes here>\footnote{footnote text}]{filename}`

Step 4

Alright so let's say now we want to get even more spicy and include an image below a codelisting. We could do something like `\codefile\n\n\image`, but by doing so, we risk separating the code and its corresponding image. So we again use a minipage (behind the scenes) to make it so that the codebox and the image are inseparable.

```
1 #include <stdio.h>
2
3 int main(void) {
4     for(int i = 0; i < 10000; ++i)
5         printf("%d\n", i); //wow
6     return 0;
7 }
```

```
[ksprague@fedora sandbox]$ python hello.py bonjour
bonjour, world
```

Figure 3

Due to the way that LaTeX does custom commands, there can only be a single optional argument, so the codebox additional arguments (language mainly) are optional, while the caption is required, but that can be left blank as is done above.

Conclusions

Here you'd talk about all of the things that people learn by reading your walkthrough.

References

- Figure 1 was obtained by running `hello.py` with the command line argument `jello`
- `\figref{filename}` will do the reference to the figure number for that filename.
- Add other references.