Contents

Tips, Tricks, and Pitfalls

Level 4

I: Format Function

1. The format function supports decimal precision formatting, similar to decimal precision when using format flags. The following examples demonstrate this:

```
principal = 100000
rate = .0894364
term = 160

# This will display rate with 4 decimal places precision.
# Note the .4f, which is the same syntax used for format flags.
string = 'The loan has principal of {0}, a rate of {1:.4f}, ' \
         'and a term of {2}.'.format(principal, rate, term)

print string

# Same thing, using keyword format specifiers.
string2 = 'The loan has principal of {prin}, a rate of {rate:.4f}, ' \
          'and a term of {t}.'.format(t=term, prin=principal, rate=rate)

print string2

# Can also make the precision formattable!...

decimalPecision = 3
string3 = 'The loan has principal of {0}, a rate of {1:.{2}f}, ' \
          'and a term of {3}.'.format(principal, rate, decimalPecision, term)

print string3

decimalPecision = 3
string4 = 'The loan has principal of {prin}, a rate of {rate:.{prec}f}, ' \
          'and a term of {t}.'.format(t=term, prin=principal, rate=rate, prec=decimalPecision)

print string4

# Slightly different...
precision = '.2f'
string3 = 'The loan has principal of {0}, a rate of {1:{2}}, ' \
          'and a term of {3}.'.format(principal, rate, precision, term)
```

II: File System

1. In the lecture, we saw how to create and rename files. However, we did not mention how to move files. Moving files is actually a variation of renaming. For example:

```
>>> import os
>>> # This will move MyFile.txt to MyDir (assuming MyDir exists):
>>> os.rename('C:\\MyFile.txt', 'C:\\MyDir\\MyFile.txt')
```

   Can also rename the file while moving:

```
>>> import os
>>> # This will move MyFile.txt to MyDir and change the filename (assuming MyDir exists):
>>> os.rename('C:\\MyFile.txt', 'C:\\MyDir\\RenamedMyFile.txt')
```

2. The **os** module does not contain any method to copy files. However, the **shutil** module does. For example:

```
>>> import shutil
>>> # This will copy MyFile to MyDir (assuming MyDir exists):
>>> shutil.copyfile('C:\\MyFile', 'C:\\MyDir\\CopiedMyFile.txt') # Can be the same or different filename
```

   Another approach (always keeps the same filename):

```
>>> import shutil
>>> # This will copy MyFile to MyDir (assuming MyDir exists):
>>> shutil.copy2('C:\\MyFile', 'C:\\MyDir')
```