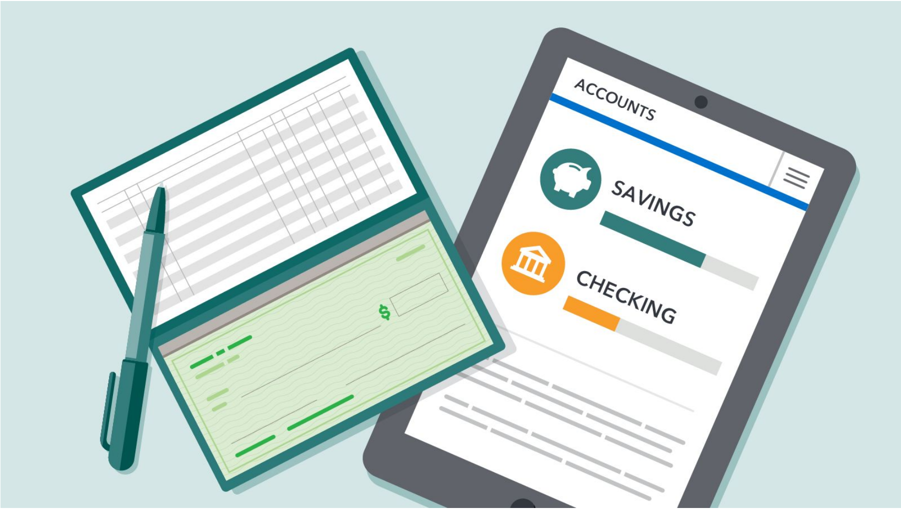
Chapter 6: Files and Functions

Feb 6, 2020



Today’s Outline

- Review:
  - Files
- Functions
- Example Problems

Quiz 4 Mistake

Which statement is false?

I fixed a mistake that I made when posing the original question. We did the question in class, so that everyone can get a free mark on the quiz.

Select one:

a. The add() method can be used to add an item to the end of a list.

b. In python lists are mutable and strings are not.

c. Unicode is a variable length encoding system.

d. Lists can contain any type of object

Tutorial 3

For Question 3 (video game background), if your student number doesn’t work as an rgb colour, that is fine. You can use any colour you like.

Files in Python

A file is a sequence of memory that is stored in secondary memory (ex. hard drive).

Conceptually a **text file** is a very long string that is stored in secondary memory.

File Processing

In Python, the open () function is used to create file objects.

```
infile = open (infileName, "r" )
outfile = open (outfileName, "w" )
appendfile = open(fileName,'a')
```

Reading lines from a text file

1: infile.read() - reads the entire file into a string

2: infile.readline() - reads the next line of a file

3: infile.readlines() - reads all of the remaining lines of a file into a list

4: for line in infile: - use to loop through each line in a file

Writing to a text file

```
print ([item to print] , file=outputfile)
```

Closing files

Once we have finished with the file, we can close it using the close() method.

Closing the file ensures that all of the changes made to the file object are saved in the file on the disk/hard drive.

Poem Example

Print the first 5 lines of the poem into a new text file named **poem5.txt**

Poem Example

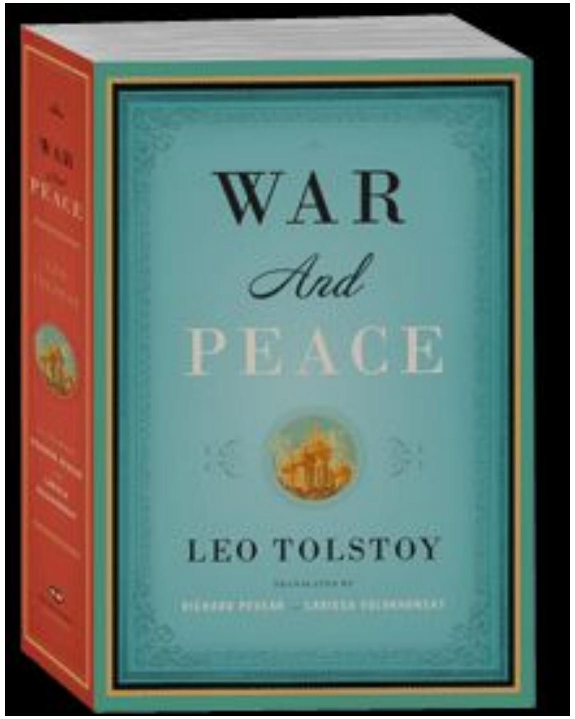
Print the first 5 lines of the poem into a new text file named **poem5.txt**

Append the last 5 lines of the poem to **poem5.txt**

War and Peace Example

War and Peace is one of the longest books ever written.

How many words are in War and Peace?



Pig Latin Bonus

Napoleon the pig is very clever and wants to read War and Peace.

Make a program that translates War and Peace into Pig Latin and saves it into a .txt file so that he can read the book.

Submit it to him by Feb 14.

Chapter 6: Functions - new concepts

1. Setting up a program with multiple functions (ex. Tutorial 2)

2. Function scope and parameters

3. Using sequences with functions

Concept 1: Function Review

Recall: A function is a “subprogram” that is outside of the main program.

When a function is “called”, the main program will be suspended, and the code inside the function will be executed.

Concept 2: Scope

Scope refers to the places in a program where a given variable may be referenced.

The variables inside a function are **local** to that function.

Scope Example 1

```
import math
def circle_area (r_miles):
 r_km = r_miles*1.60934
 A = math.pi*r_km**2
 return round(A,2)
```

*** The r_km variable is **local** to the circle_area function. We cannot directly access it in our main program.

Multiple Parameters

- Parameters are assigned based on their position in the function:

```
def richter_scale(A1,A0):
 M_l = math.log10(A1)-math.log10(A0)
 return round(M_l,1)


richter_scale(400,0.1)
```

Multiple Parameters

- You can also assign input parameters directly

```
def richter_scale(A1,A0):
 M_l = math.log10(A1)-math.log10(A0)
 return round(M_l,1)


richter_scale(A0=0.1, A1=400)
```

Print function parameters

Print() parameters:

print(value, . . . , sep, end, file)

** In this case, we need to use the optional keywords if we want to indicate the separation between values, the ending of the output, and the file to print to.

Is this ok?

```
def convert_time(time24):
 time12 = (time24 + 11)%12 + 1
 return time12


def main():
  time24 = 14
  time12 = convert_time(time24=time24)
  print(time12)

main()
```

Concept 3: Sequences with Functions

Sequences with Functions

Write an efficient program that sings the song “Happy Birthday” to each of the Dionne quintuplets: Yvonne, Annette, Cécile, Émilie, and Marie.

Happy birthday to you!

Happy birthday to you!

Happy birthday dear [name]

Happy birthday to you!

Birthday Functions

- Function to sing “Happy Birthday”
  - Function to sing “Happy birthday to you!”

Happy birthday to you!

Happy birthday to you!

Happy birthday dear [name]

Happy birthday to you!

Quintuplet Names

```
quintuplets = ["Yvonne", "Annette", "Cécile", "Émilie", "Marie"]
```

Sing “happy birthday” to each quintuplet using a for loop.

What will the output be?

```
def addInterest(balance, rate):
  newBalance = balance * (1+rate)
  balance = newBalance


def main():
  amount = 1000
  rate = 0.05
  addInterest(amount,rate)
  print(amount)
main()
```

What will the output be?

```
def addFruits(fruit_list):
  fruit_list.append("grape")
  fruit_list.append("pear")
  return fruit_list

def main():
  fruits = ["strawberry", "banana", "orange", "apple"]
  addFruits(fruits)
  print(fruits)
main()
```

What will the output be?

```
def addFruits(fruit_list):
  fruit_list.append("grape")
  fruit_list.append("pear")
  return fruit_list

def main():
  fruits = ["strawberry", "banana", "orange", "apple"]
  addFruits(fruits)
  print(fruits)
main()
```

**fruits and fruit_list are referring to the same list in memory

**in this specific case, updating fruit_list is the same as updating fruits

Similar to this Example

```
fruits = ["strawberry", "banana", "orange", "apple"]

fruit_list = fruits

fruit_list.append("grape")

fruit_list.append("pear")

print(fruits)
```

Graphics Object Example

```
import graphics
def moveEye (eye_object,x,y):
    eye_object.move(x,y)

win = graphics.GraphWin()
eye = graphics.Circle(graphics.Point(0,0),30)
eye.setFill("blue")
eye.setOutline("black")
eye.draw(win)
moveEye(eye,100,100)
```

Summary

Variables within a function are local to the function.

Exceptions occur if an object that is passed into a function is **mutable**.

Since lists and graphics objects are mutable, the function will be able to change the state of these objects in the main program.

Bank Accounts

Make a function that adds interest to a list of bank accounts.

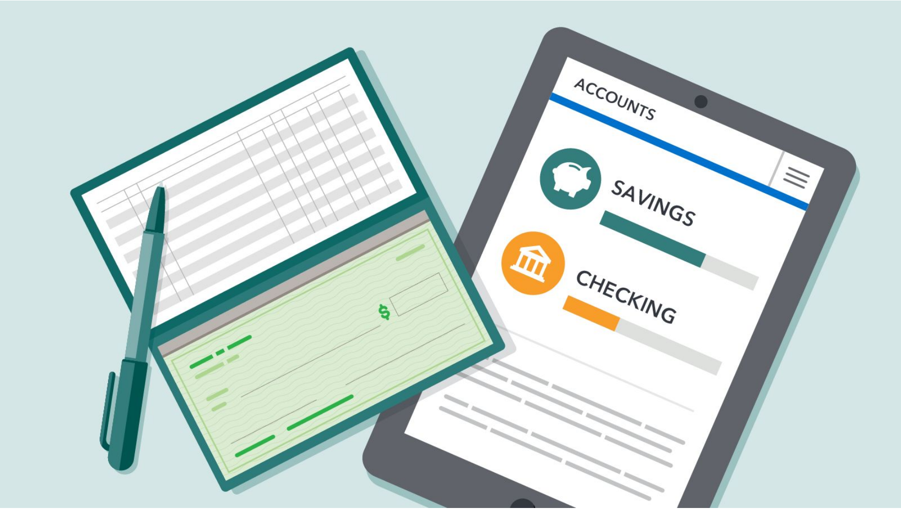
Input Example:

accounts = [1000, 0, 50000, 1000000]

rate = 0.05

Output Example:

accounts = [1050, 0, 52500, 1050000]



Frequency Problem

Estimate the relative frequency of the 26 letters in English.

Frequency Problem

Estimate the relative frequency of the 26 letters in English.

Procedure:

1) Find a large text file of English writing

1) Count the number of each type of letter in the file.

**Hint: the string method s.count('character') might be useful for this.