|  |  |  |  |
| --- | --- | --- | --- |
| Blue text on a black background  Description automatically generated with medium confidence | A blue text on a black background  Description automatically generated with medium confidence |  | Wilson Woon 2024 ©  Updated on 17 Feb 2024 |

**Problem Set 1: Python Basics  
  
Total Marks: 25 points  
% of Grade: 5%  
Deadline: Monday, 4 March (23:55)  
Penalties for Late Submission: Please refer to the Course Outline**  
Instructions: Answer all questions. Submit all the required files to Canvas.  
  
  
**Question 1: Playback Speed [3 points]**In a file called *playback.py*, implement a program in Python that prompts the user for input and then outputs that same input, replacing each space with ... (i.e., three periods).

Hints:

1. *input* function returns a *str*, as per <docs.python.org/3/library/functions.html#input>
2. a *str* comes with quite a few methods, per <docs.python.org/3/library/stdtypes.html#string-methods>

Expected output:



**Question 2: Einstein [3 points]**Even if you haven’t studied physics (recently or ever!), you might have heard that , wherein *E* represents energy (measured in Joules), *m* represents mass (measured in kilograms), and *c* represents the speed of light (measured approximately as 300000000 meters per second), per [Albert Einstein et al](https://en.wikipedia.org/wiki/Albert_Einstein). Essentially, the formula means that mass and energy are equivalent.  
  
In a file called *einstein.py*, implement a program in Python that prompts the user for mass as an integer (in kilograms) and then outputs the equivalent number of Joules as an integer. Assume that the user will input an integer.  
  
Hints:

1. *input* function returns a *str*, as per <docs.python.org/3/library/functions.html#input>
2. *int* can convert a *str* to an *int*, per <https://docs.python.org/3/library/functions.html#int>
3. Python comes with several built-in functions as indicated in <https://docs.python.org/3/library/functions.html>

Expected output:

A close-up of a white background

Description automatically generated  
  
**Question 3: Tip Calculator [5 points]**In the United States, it’s customary to leave a tip for your server after dining in a restaurant, typically an amount equal to 15% or more of your meal’s cost. Not to worry, though, we’ve written a tip calculator for you, below!

A computer screen shot of a program code

Description automatically generated

Well, we’ve written most of a tip calculator for you. Unfortunately, we didn’t have time to implement two functions:

* *dollars\_to\_float*, which should accept a str as input (formatted as $##.##, wherein each # is a decimal digit), remove the leading $, and return the amount as a float. For instance, given $50.00 as input, it should return 50.0.
* *percent\_to\_float*, which should accept a str as input (formatted as ##%, wherein each # is a decimal digit), remove the trailing %, and return the percentage as a float. For instance, given 15% as input, it should return 0.15.

Assume that the user will input values in the expected formats. Save your program in a file called *tip.py*.  
  
Hints:

1. *input* function returns a *str*, as per <docs.python.org/3/library/functions.html#input>
2. *float* can convert a *str* to a *float*, per <https://docs.python.org/3/library/functions.html#float>
3. a *str* comes with quite a few methods, per <docs.python.org/3/library/stdtypes.html#string-methods>

Expected Output:

**Sample 1**

A close-up of a white background

Description automatically generated

**Sample 2**

A close up of black text

Description automatically generated  
  
**Question 4: Home Federal Savings Bank [4 points]**

In a file called *bank.py*, implement a program that prompts the user for a greeting. If the greeting starts with “hello”, output $0. If the greeting starts with an “h” (but not “hello”), output $20. Otherwise, output $100. Ignore any leading whitespace in the user’s greeting, and treat the user’s greeting case-insensitively.  
  
Hints:

1. a str comes with quite a few methods, per <docs.python.org/3/library/stdtypes.html#string-methods>
2. Be sure to give $0 not only for “hello” but also “hello there”, “hello, Newman”, and the like.

Expected Output:

**Sample 1**



**Sample 2**



**Sample 3**



**Sample 4**



**Sample 5**



**Sample 6**



**Sample 7**



**Question 5: Math Interpreter [5 points]**

Python already supports math, whereby you can write code to add, subtract, multiply, or divide values and even variables. But let’s write a program that enables users to do math, even without knowing Python.  
  
In a file called *interpreter.py*, implement a program that prompts the user for an arithmetic expression and then calculates and outputs the result as a floating-point value formatted to one decimal place. Assume that the user’s input will be formatted as x y z, with one space between x and y and one space between y and z, wherein:

* x is an integer
* y is +, -, \*, or /
* z is an integer

For instance, if the user inputs 1 + 1, your program should output 2.0.

Hints:

* Recall that a *str* comes with quite a few methods, per <docs.python.org/3/library/stdtypes.html#string-methods>, including split, which separates a *str* into a sequence of values, all of which can be assigned to variables at once. For instance, if the expression is a *str* like 1 + 1, then

x, y, z = expression.split(" ")

will assign 1 to x, + to y, and 1 to z.

* Your program will need to handle the case where y is / and z is a 0 or divide by zero. If this case is detected, you should print out “Cannot divide by zero”.

Expected Output:

**Sample 1**



**Sample 2**



**Sample 3**



**Sample 4**



**Sample 5**

  
  
**Question 6: Coca Cola Machine [5 points]**

Suppose a machine sells bottles of Coca-Cola (Coke) for 50 cents and only accepts coins in these denominations: 25 cents, 10 cents, and 5 cents.  
  
In a file called *coke.py*, implement a program that prompts the user to insert a coin, one at a time, each time informing the user of the amount due. Once the user has inputted at least 50 cents, output how many cents in change the user is owed. Assume that the user will only input integers and ignore any integer that isn’t an accepted denomination.  
  
Expected output:  
  
**Sample 1**

A white background with black text

Description automatically generated

**Sample 2**

A close-up of black text

Description automatically generated

**Sample 3**

A white text with black text

Description automatically generated

**Sample 4**

A screenshot of a computer

Description automatically generated

**Reference**

All the materials were adapted with modifications from Harvard’s CS50P.