Homework 1 Rubric – Name

[10 pts] Question 1:

[4 pts] Write a program to print 10, 10.0, “10.0”.

[] 1 pt if the program prints 10

[] 1 pt if the program prints 10.0

[] 1 pt if the program prints "10.0" (as a string)

[] 1 pt for correct syntax in each print statement

[6 pts] Find the type of each and explain the differences between 10, 10.0, and “10.0”.

[] 1 pt correctly identifying the type of 10 (integer)

[] 1 pt correctly identifying the type of 10.0 (float)

[] 1 pt correctly identifying the type of "10.0" (string)

[] 1 pt explanation that 10 is an integer, representing whole numbers

[] 1 pt explanation that 10.0 is a float, representing decimal or fractional numbers

[] 1 pt explanation that "10.0" is a string, which is just a sequence of characters (text) and not a number

[15 pts] Question 2:

[5 pts] Print the expression that adds two integers, 5 and 10.

[] 3 pts correct addition of 5 and 10, printing 15

[] 2 pts correct syntax

[5 pts] Concatenate two strings, '5' and '10', and print the result.

[] 3 pts correct concatenation of '5' and '10' to form "510"

[] 2 pts correct syntax

[5 pts] What is the difference between adding integers and concatenating strings?

[] 2 pts explanation that adding integers results in a numeric sum

[] 3 pts explanation that concatenating strings joins them into a single string without performing arithmetic

[20 pts] Question 3:

[7 pts] Verify if Python will accept the code below. Explain why or why not.

[] 3 pts explanation that Python does not accept adding a string and an integer directly

[] 2 pts conversion of the string '100' to an integer or conversion of 50 to a string

[] 2 pts correct result printed after conversion

[8 pts] Let x be ‘2’ and y be 100. Find the product of x and y and let the product have a variable name z. Concatenate it with ' is the same as 2x100.'.

[] 2 pts correctly assigning x as '2' and y as 100

[] 2 pts correct product calculation after converting x to an integer

[] 2 pts assigning the result to variable z

[] 2 pts concatenation with ' is the same as 2x100.'

[5 pts] What happens if you try to convert the string 'hello' to an integer?

[] 5 pts explanation that Python raises a ValueError because 'hello' is not a valid integer representation

[10 pts] Question 4:

[5 pts] Assign the value 25 to a variable called age. Print the value of age. Reassign age to a new value of 'twenty-five' (a string). Print the new value and check its type.

[] 1 pt value 25 assigned to the variable age

[] 1 pt value of age printed

[] 1 pt age reassigned to 'twenty-five'

[] 1 pt new value of age printed

[] 1 pt type of the new value correctly printed

[5 pts] How does Python handle changes in variable types?

[] 5 pts explanation that Python allows dynamic typing, where a variable can change type between assignments.

[15 pts] Question 5:

[5 pts] Assign values to three variables a, b, and c in a single line using tuple assignment. Print them in a single line, separated by commas.

[] 2 pts values correctly assigned using tuple assignment

[] 3 pts values printed in a single line, separated by commas

[5 pts] Print them with custom formatting using the format() method, where a is left-aligned in 10 spaces and b and c are right-aligned in 10 spaces.

[] 5 pts values printed with the correct formatting using .format()

[5 pts] How does tuple assignment work?

[] 5 pts correct explanation that tuple assignment allows multiple variables to be assigned in a single statement by unpacking a tuple

[15 pts] Question 6:

[5 pts] Prompt the user to enter their name. Use \n to print a greeting on two lines.

[] 2 pts correct user input prompt

[] 3 pts correct use of \n to print the greeting on two lines

[5 pts] Additionally, use \t to print their name indented with a tab.

[] 5 pts correct use of \t to print the name indented with a tab

[5 pts] How do control codes like \n and \t change the output formatting?

[] 5 pts correct explanation that \n creates a new line and \t creates a tab space to control the output layout

[15 pts] Question 7:

[5 pts] Write a program that asks the user to input two floating-point numbers.

[] 2 pts correct input prompts for two floating-point numbers

[] 3 pts inputs stored correctly

[5 pts] Apply addition, subtraction, multiplication, and division between two numbers.

[] 5 pts correct application and printing of addition, subtraction, multiplication, and division

[2 pts] Print the result of each operation after rounding it up to the thousandth place.

[] 2 pts Results rounded to three decimal places and printed correctly

[3 pts] What happens if the user enters integers instead of decimal numbers?

[] 3 pts correct explanation that Python automatically converts integers to floats for operations

---

[-10 pts] Submission Requirements (no points added, only deducted if not followed)

[-8 pts] Incorrect file(s) submission:

[] PDF file with screenshots submitted in the correct format and named yourname_cs110_hw1.pdf and corresponding .py file

[-2 pts] Missing format:

[] student's name, ID, and Honor Code statement correctly included

Total: /100