Your Name:	TJ Sipin
Umail:	tjsipin@umail.ucsb.edu
Perm #:	343050-1

This homework and its due date are posted on the course website: https://ucsb-cs8.github.io/w20/. Submit a PDF file of this assignment following the guidelines posted on the website.

h01: Perkovic Ch 2.1 (Expressions Variables and Assignments) + 2.2 (Strings)

0. (10 points) Use the provided homework template (without any blank pages), filling out all requested fields, and correctly submitting a legible electronic document on Gradescope before the due date. By submitting this document you agree that this is your own work and that you have read the policies regarding Academic Integrity: https://studentconduct.sa.ucsb.edu/academic-integrity.

1. (2 pts each) Section 2.1 describes how several operators and built-in functions in Python work. What would be the result of entering the following expressions at the Python IDLE shell prompt?

(*Note: You are encouraged to check your answers at the Python prompt before turning in your work, but try this on paper first, just by reading the text and trying to predict what will happen. Then try typing in the results at the Python prompt. Change your answers if they were originally incorrect, but even more important, try to figure out why you were incorrect.*)

Be very precise. Note that `True` is not the same in Python as `true`; *capitalization*, i.e., upper vs. lower case, matters. You will not get full credit for answers that are not precisely correct.

Expression	**Result**	**Expression**	**Result**
`11 % 7`	4	`8 / 5`	1.6
`2**3**4`	2417851639229258349412352	`8 // 5`	1
`(2**3)**4`	4096	`6 + 2 == 8`	True
`True or False`	True	`8 != 16/2`	False
`True and False`	False	`8 == True`	False

2. (5 pts) As described in Section 2.1, a Python assignment statement contains the *assignment operator* (i.e, =), an *expression* and a *variable*, but not in that order. What is the correct order for these three parts, reading from left to right? Make sure you understand what each part represents.

variable	assignment operator	expression

3. (1 pts each) Section 2.1 contains a list of thirty-three *reserved words* in Python that may not be used as the name of a variable. You don't need to memorize this list, but you do need to know where to find it, either in the book, or online. So, to be sure you can find it, place a checkmark next to *only* the words that are true Python keywords.

☑ `and`	☐ `end`	☑ `elif`	☐ `include`
☑ `break`	☐ `define`	☑ `else`	☑ `None`
☑ `continue`	☐ `delete`	☐ `false`	☑ `return`

4. (2 pts each) Section 2.2 describes strings in Python, including the concepts of "dictionary order" (also called "lexicographic order"), concatenation of strings, multiplication of strings by an integer, the `in` operator, the `not in` operation, the `len()` function, and string indexing. Review that material.

Assuming the following assignment statements have been entered at the Python prompt in IDLE to create the three variables below, indicate the value of each of the expressions on the right:
 	
`school = "UCSB"`
`course = "CS8"`
`qtr = "W20"`
	Expression	**Result**

	`course * 2`	CS8CS8
	`(school + '~')*2`	UCSB~UCSB~
	`qtr[1:3]`	20
	`qtr > course`	True
	`"CS" in school`	True

5. (4 pts each) **Using the three variables above**, string indexing, and operators `+` and `*`, write Python *expressions* that evaluate to:

Expression	**Result**
school + " " + course + " " + qtr	`UCSB CS8 W20`
(school + " " + course + " ") * 2	`UCSB CS8 UCSB CS8`
`school[` 1:2 `] == course[` 0:1 `]`	`True`