Problem 1: string comparison

Hint: the following is True: "" < "0" < "9" < "A" < "Z" < "a" < "z"

Circle the expressions that are True:

"a" < "z"	"ax" < "ay"	"abc" < "abCd"
"a" < "Z"	"x2" < "x1"	"zero" < "999"
"x" < "x"	"abcX" < "abcY"	"10" < "999"
"0" < "x"	"abcX" < "aBcY"	"1000" < "999"
"1" < "0"	"abc" < "abcd"	"88888888888" < "9"

Problem 2: string functions

Functions: upper, lower, strip, rstrip, lstrip, format, startswith, endswith, find.

Expression:	**Value (put in quotes):**
"dog".upper()	
"Dog".lower()	
" paint ".strip()	
" paint ".rstrip()	
"val: {}%".format(99)	
"{} {}".format("X", "Y")	

Expression:	**Value**
"abcd".startswith("ab")	
"abcd".endswith("bc")	
"abcd".find("a")	
"abcd".find("c")	
"abcd".find("B")	
"Python".find("th")	

Problem 3: sequence indexing

Assume **msg** is "Hello" and **x** is "num= 13". Some expressions cause an error.

Expression	**Result**
"abc"[0]	
"abc"[2]	

Expression	**Result**
msg[4]	
msg[5]	

Expression	**Result**
x[len(x) - 1]	
x[3]	

"abc"[-1]	

msg[len(msg)]	

x[1] + x[2]	

Problem 4: sequence slicing

Assume **msg** and **x** are as before, and **p** is "= ".

Expression	Result
"abcde"[0:2]	
"abcde"[2:6]	
"abcde"[2:9]	

Expression	Result
msg[:2]	
msg[2:]	
msg[-2:]	

Expression	Result
x[:x.find('=')]	
x[x.find(' ')+1:]	
x[x.find(p)+len(p):]	

Problem 5: for loop over sequence

What does the following code print?

```
msg = "301"
A = ""
B = ""

for character in msg:
    print(msg)
    A = A + character + "."
    B = character + B
```

What is in A afterwards?

What is in B afterwards?

Problem 6: for loop over range

What does this code print?

```
s = "PYTHON"
for i in range(len(s)):
    print(s[:i+1])
```