MITx 6.00.1x
Introduction to Computer Science and Programming Using Python

Week 1 - Python Basics / 2. Core Elements of Programs

Exercises 1

For each of the following expressions, indicate the value returned, or if the evaluation would lead to an error, write the word 'error' (note this is a word, not a string, no quotes). While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

`"a" + "bc"`
Answer: ‘abc’

`3 * "bc"`
Answer: ‘bcbcbc’

`"3" * "bc"`
Answer: error

`"abcd"[2]`
Answer: ‘c’

`"abcd"[0:2]`
Answer: ‘ab’

`"abcd"[:2]`
Answer: ‘ab’

`"abcd"[2:]`
Answer: ‘cd’

Exercises 2 Part 1

For each of the expressions below, specify its type and value. If it generates an error, select type 'NoneType' and write the word 'error' (note this is a word, not a string, no quotes) in the box for the value. While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understand of basic Python expressions.

Assume we’ve made the following assignments:

```
> str1 = 'hello'
> str2 = ','
> str3 = 'world'
```

1. `str1`
Answer: string
Answer: ‘hello’

2. `str1[0]`
Answer: string
Answer: ‘h’

3. `str1[1]`
Answer: string
Answer: ‘e’

4. `str1[-1]`
Answer: string
Answer: ‘o’

5. `len(str1)`
Answer: int
Answer: 5

Exercises 2 Part 2

For each of the expressions below, specify its type and value. If it generates an error, select type 'NoneType' and write the word 'error' (note this is a word, not a string, no quotes) in the box for the value. While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understand of basic Python expressions.

Assume we’ve made the following assignments:

```
> str1 = 'hello'
> str2 = ','
> str3 = 'world'
```

1. `str1[len(str1)]`
Answer: NoneType
Answer: error

2. `str1 + str2 + str3`
Answer: string
Answer: ‘hello,world’

3. `str1 + str2 + ' ' + str3`
Answer: string
Answer: ‘hello, world’

4. `str3 * 3`
Answer: string
Answer: ‘worldworldworld’

5. `'hello' == str1`
Answer: boolean
Answer: True

Exercises 2 Part 3

For each of the expressions below, specify its type and value. If it generates an error, select type 'NoneType' and write the word 'error' (note this is a word, not a string, no quotes) in the box for the value. While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understand of basic Python expressions.

Assume we’ve made the following assignments:

```
> str1 = 'hello'
> str2 = ','
> str3 = 'world'
```

1. `'HELLO' == str1`
Answer: boolean
Answer: False

2. `'a' in str3`
Answer: boolean
Answer: False

3.

```
str4 = str1 + str3
'low' in str4
```

Answer: boolean
Answer: True

4. `str3[1:3]`
Answer: string
Answer: ‘or’

5. `str3[:3]`
Answer: string
Answer: ‘wor’

Exercises 2 Part 4

For each of the expressions below, specify its type and value. If it generates an error, select type 'NoneType' and write the word 'error' (note this is a word, not a string, no quotes) in the box for the value. While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understand of basic Python expressions.

Assume we’ve made the following assignments:

```
> str1 = 'hello'
> str2 = ','
> str3 = 'world'
> str4 = str1 + str3
```

1. `str3[:-1]`
Answer: string
Answer: ‘worl’

2. `str1[1:]`
Answer: string
Answer: ‘ello’

3. `str4[1:9]`
Answer: string
Answer: ‘elloworl’

4. `str4[1:9:2]`
Answer: string
Answer: ‘elwr’

5. `str4[::-1]`
Answer: string
Answer: ‘dlrowolleh’

Exercises 3

For each of the following expressions, indicate the value that prints out when the expression is evaluated. If the evaluation would lead to an error, write the word 'error'; if nothing would print out, write the word 'blank'.

While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

If the temperatures seem weird to you, like most of the world, you probably use the Celsius system. We Americans still use the crazy Fahrenheit system...

1.

```
if 6 > 7:
   print("Yep")
```

Answer: blank

2.

```
if 6 > 7:
   print("Yep")
else:
   print("Nope")
```

Answer: ‘Nope’

3.

```
var = 'Panda'
if var == "panda":
   print("Cute!")
elif var == "Panda":
   print("Regal!")
else:
   print("Ugly...")
```

Answer: ‘Regal!’

4.

```
temp = 120
if temp > 85:
   print("Hot")
elif temp > 100:
   print("REALLY HOT!")
elif temp > 60:
   print("Comfortable")
else:
   print("Cold")
```

Answer: ‘Hot’

5.

```
temp = 50
if temp > 85:
   print("Hot")
elif temp > 100:
   print("REALLY HOT!")
elif temp > 60:
   print("Comfortable")
else:
   print("Cold")
```

Answer: ‘Cold’

Exercises 4

Below are some short Python programs. For each program, answer the associated question.

Try to answer the questions without running the code. Check your answers, then run the code for the ones you get wrong.

This question is going to ask you what some simple loops print out. If you're asked what code like this prints:

```
num = 5
if num > 2:
    print(num)
    num -= 1
print(num)
```

Write what it prints out, separating what appears on a new line by a comma and a space. So the answer for the above code would be:

```
5, 4
```

If a given loop will not terminate, write the phrase 'infinite loop' (no quotes) in the box. Recall that you can stop an infinite loop in your program by typing CTRL+c in the console.

Note: What does `+=`, `-=`, `*=`, `/=` stand for?

`a += b` is equivalent to `a = a + b`

`a -= b` is equivalent to `a = a - b`

`a *= b` is equivalent to `a = a * b`

`a /= b` is equivalent to `a = a / b`

MITx 6.00.1x
Introduction to Computer Science and Programming Using Python

1.

```
num = 0
while num <= 5:
    print(num)
    num += 1

print("Outside of loop")
print(num)
```

Answer: 0, 1, 2, 3, 4, 5, 'Outside of loop', 6

2.

```
numberOfLoops = 0
numberOfApples = 2
while numberOfLoops < 10:
    numberOfApples *= 2
    numberOfApples += numberOfLoops
    numberOfLoops -= 1
print("Number of apples: " + str(numberOfApples))
```

Answer: infinite loop

3.

```
num = 10
while num > 3:
    num -= 1
    print(num)
```

Answer: 9, 8, 7, 6, 5, 4, 3

4.

```
num = 10
while True:
    if num < 7:
        print('Breaking out of loop')
        break
    print(num)
    num -= 1
print('Outside of loop')
```

Answer: 10, 9, 8, 7, 'Breaking out of loop', 'Outside of loop'

5.

```
num = 100
while not False:
    if num < 0:
        break
print('num is: ' + str(num))
```

Answer: Infine Loop

Exercises 5

Below are some short Python programs. For each program, answer the associated question.

Try to answer the questions without running the code. Check your answers, then run the code for the ones you get wrong.
This question is going to ask you what some simple loops print out. If you're asked what code like this prints:

```
num = 5
if num > 2:
    print(num)
    num -= 1
print(num)
```

write what it prints out, separating what appears on a new line by a comma and a space. So the answer for the above code would be:

```
5, 4
```

If a given loop will not terminate, write the phrase 'infinite loop' in the box.

```
num = 10
for num in range(5):
    print(num)
print(num)
```

Answer: 0, 1, 2, 3, 4, 4

```
divisor = 2
for num in range(0, 10, 2):
    print(num/divisor)
```

Answer: 0.0, 1.0, 2.0, 3.0, 4.0

```
for variable in range(20):
    if variable % 4 == 0:
        print(variable)
    if variable % 16 == 0:
        print('Foo!')
```

Answer: 0, 'Foo!', 4, 8, 12, 16, 'Foo!'

```
for letter in 'hola':
    print(letter)
```

Answer: 'h', 'o', 'l', 'a'

```
count = 0
```

```
for letter in 'Snow!':
    print('Letter # ' + str(count) + ' is ' + str(letter))
    count += 1
    break
print(count)
```

Answer: 'Letter # 0 is S', 1

Exercises 6

Below are some short Python programs. For each program, answer the associated questions.

Try to answer the questions without running the code. Check your answers, then run the code for the ones you get wrong. You'll learn the most this way, by figuring things out, instead of just running the code and reading off the answers.

```
myStr = '6.00x'

for char in myStr:
    print(char)

print('done')
```

1. How many times does 6 print out?
Answer: 1

2. How many times does . print out?
Answer: 1

3. How many times does 0 print out?
Answer: 2

4. How many times does x print out?
Answer: 1

5. How many times does done print out?
Answer: 1

```
greeting = 'Hello!'
count = 0

for letter in greeting:
    count += 1
    if count % 2 == 0:
        print(letter)
    print(letter)

print('done')
```

1. How many times does H print out?
Answer: 1

2. How many times does e print out? Disregard the letters in the word done.
Answer: 2

3. How many times does l print out?
Answer: 3

4. How many times does o print out?
Answer: 1

5. How many times does ! print out?
Answer: 2

6. How many times does done print out?
Answer: 1

```
school = 'Massachusetts Institute of Technology'
numVowels = 0
numCons = 0

for char in school:
    if char == 'a' or char == 'e' or char == 'i' \
       or char == 'o' or char == 'u':
        numVowels += 1
    elif char == 'o' or char == 'M':
        print(char)
    else:
        numCons -= 1

print('numVowels is: ' + str(numVowels))
print('numCons is: ' + str(numCons))
```

1. How many times does o print out? Disregard the o’s in the last two print statements.
Answer: 0

2. How many times does M print out?
Answer: 1

3. How many times does numVowels print out?
Answer: 11

4. How many times does numCons print out?
Answer: -25

Exercises 7

Code Sample

```
iteration = 0
count = 0
while iteration < 5:
    for letter in "hello, world":
        count += 1
    print("Iteration " + str(iteration) + "; count is: " + 
str(count))
    iteration += 1
```

We wish to re-write the above code, but instead of nesting a `for` loop inside a `while` loop, we want to nest a `while` loop inside a `for` loop. Which of the following loops gives the same output as the Code Sample?

Try to answer the following questions by just reading the code. Reading code is a very good skill to have (and will help you both in your programming career and on your exams!). It is okay to check your answers that you obtain from just reading the code, then in your interpreter run the code for the ones you got wrong on your first attempt.

Test 1

```
for iteration in range(5):
    count = 0
    while True:
        for letter in "hello, world":
            count += 1
        print("Iteration " + str(iteration) + "; count is: " + 
str(count))
```

Answer: No, Test 1 does not give the same output as the Code Sample

Test 2

```
for iteration in range(5):
    count = 0
    while True:
        for letter in "hello, world":
            count += 1
        print("Iteration " + str(iteration) + "; count is: " + 
str(count))
        break
```

Answer: No, Test 2 does not give the same output as the Code Sample

Test 3

```
count = 0
phrase = "hello, world"
for iteration in range(5):
    index = 0
    while index < len(phrase):
        count += 1
        index += 1
    print("Iteration " + str(iteration) + "; count is: " + 
str(count))
```

Answer: Yes, Test 3 gives the same output as the Code Sample

Test 4

```
count = 0
phrase = "hello, world"
for iteration in range(5):
    while True:
        count += len(phrase)
        break
    print("Iteration " + str(iteration) + "; count is: " +
str(count))
```

Answer: Yes, Test 4 gives the same output as the Code Sample

Test 5

```
count = 0
phrase = "hello, world"
for iteration in range(5):
    count += len(phrase)
    print("Iteration " + str(iteration) + "; count is: " +
str(count))
```

Answer: Yes, Test 5 gives the same output as the Code Sample