

# CSC110 Leture 2: Representing Data

Hisbaan Noorani

September 14, 2020

## Contents

<b>1 Ex 1: Categorizing Data</b>	<b>1</b>
<b>2 Ex 2: Data Types in Python</b>	<b>1</b>
<b>3 Ex 3: Variables</b>	<b>2</b>
<b>4 Ex 4: comprehensions and range</b>	<b>3</b>

## 1 Ex 1: Categorizing Data

Imagine that you are a part of a deeply interconnected society of big cats owners in America. In particular, you are the proud owner of a private zoo set up in Oklahoma that houses many species of dangerous big cats such as tigers, lions, puma, ligers and tigons. You know of other owners of private zoos and their “collections” of large cats.

1. Think about different kinds of data you would have in this setting. For each data type below, write a description of a piece of data that would have that type. We have completed the first row for you.

Data Type	Example
Boolean	“Have I been shut down by the government today?”
Natural Number	“The number of big cats in my posession”
Real Number	“The ratio of big cats to owners where it is $\frac{\text{owners}}{\text{big cats}}$ ”
String	“The name of one of the big cats”
Set	“The set of names of the big cats”
List	“The set of names of the big cats, arranged by aquisition date”
Map	“The names of the big cats mapped to their ages”

## 2 Ex 2: Data Types in Python

1. In the table below, write down the python type for each python literal.  
The first row is done for you.

Data Type	Python Literal
int	12345
float	12345.0
bool	True
str	'You are doing great :).'
set	{'David', 'Mario'}
list	[1, 2, 3, 4, 5]
tupple	('wow', 'roud', 'brackets')
dictionary	{'cool'}

2. What is the limitation of the float datatype, and why does this limitation exist?

Floats are limited to a certain amount of decimal places because computers have a finite amount of ram.

3. Without using the python console, complete the table below.

Python Expression	Value	Type of Value
9 / 3	3.0	float
9 // 3	3	int
2 + 2.0	4.0	float
3 int {1, 2}	False	bool
'x' + 'y'	'xy'	str
not False	True	bool
\[1, 2] + [2, 3]	[1, 2, 2, 3]	list
\['hi', 'bye'][0]	'hi'	str

4. In the table below, add parentheses to indicate the order that operations are evaluated (the last one is not obvious)

Python Expression	Expression with ()	Type of result
3 + 5 * 2	3 + (5 * 2)	int
4 + 8 / 2 ** 2 / -2	4 + ((8 / (2 ** 2)) / 2)	int
-2 + 15 % 7	-2 + (15 % 7)	int

### 3 Ex 3: Variables

1. Consider this code

```
1 >>> x = 4
2 >>> y = x + 2
3 >>> z = {y: x}
```

After the above code is executed, to which values do  $x$ ,  $y$ , and  $z$  refer?

Variable	Value
$x$	4
$y$	6
$z$	{6: 4}

2. Determine whether the assignment statements below have been properly expressed. If it is, move onto the next statement. If it is not, explain why it is incorrect and correct the statement.

(a)  $5 = x$

Variable names cannot begin with a number should be:  $x = 5$

(b)  $x = 5.0$

Correct

(c)  $x = \text{hello!}$

If `hello!` is a string, it must be surrounded by `"` If `hello!` is a variable name, it cannot contain `!` should be either  $x = \text{'hello!'}$  or  $x = \text{hello}$

(d)  $x = \text{true}$

True should be capitalized if it is a boolean or surrounded by `"` if it is a string should be either  $x = \text{True}$  or  $x = \text{'true'}$

(e)  $x = \text{set}()$

Correct

(f)  $x = [1, \text{'hi'}, 3]$

Correct

(g)  $x = \{1: \text{'two'}, \text{'three'}\}$

dict incomplete should be  $x = \{1: \text{'two'}, 2: \text{'three'}\}$

## 4 Ex 4: comprehensions and range

1. What range of numbers do the following python range expressions represent?

(a) `range(0, 5)`

0, 1, 2, 3, 4

(b) `range(5, 10)`

5, 6, 7, 8, 9

(c) `range (5, 5)`

`[]`

2. What is one similarity between a set literal and a set comprehension? What is one difference?

Similarity - They both use curly braces `{}` to define them.

Difference - Set literals are fixed, set comprehension can vary based on variables.

3. For each of the following descriptions, write a comprehension that evaluates to the set.

(a) The set of integers between 30 and 50, inclusive.

```
1 {v for v in range(30, 51)}
```

(b) The set of squares of the natural numbers less than 2000.

```
1 {v ** 2 for v in range(0, 44)}
```

(c) The list of integers between -30 and 30 inclusive.

```
1 [v for v in range(-30, 31)]
```

4. You are given a variable `s` that refers to a (very very long string). Write an expression that evaluates to a list containing the first 20 characters in the string, in the order they appear.

```
1 [s[v] for v in range(20)]
```

5. For each of the following mappings, write a Python dictionary expression that evaluates to the mapping.

(a) A mapping from a number to its square for the first 50 natural numbers.

```
1 {v: v ** 2 for v in range(51)}
```

(b) A mapping from input to output of the function  $f(x) = x/(x-1)$  for integer inputs greater than 1 and less than 2000

```
1 {v: (v)/(v-1) for v in range(2,2000)}
```

(c) A mapping from a number to a list containing that many elements of the string 'Hello' for the first 40 natural numbers

```
1 {v: ['Hello' for x in range(v)] for v in range(51)}
```

(d) A mapping from an integer to the set of integers between 0 and that integer, for integers from 1 to 20, inclusive

```
1 {v: {x for x in range(v + 1)} for v in range(21)}
```