LEARN AND HELP

Course	Python 101
Term	
Class #	
Date	
Chapter. Topic	2. Input, Processing, and Output

Python Basics
Variables, Data Types, Casting, inputs, outputs

Siva R Jasthi

Computer Science and Cybersecurity

Metropolitan State University

Outline

1. Data Types (Real Life Examples)
2. Variables
3. Comments
4. Python Data Data Types
5. Printing Variables
6. Strings, Single quotes, double quotes, triple quotes
7. Taking inputs from users
8. Casting the data types
9. Arithmetic Operators

10. Python Tutor - Visualizing the programs
11. Python Tutor - Saving the link to your program
12. Google colab

Let us chit chat

1. What is your name?
2. Who invented the formula 'E = MC*2'?
3. Where do you live? →
4. What is your favorite food? →

What is the pattern here?

Let us chit chat

1. How many countries did you visit? →
2. How many students are in your classroom? →
3. How many of them got A grade? →
4. How many doors on the car you have? →

What is the pattern? What is the data type?

Let us chit chat

1. What is the distance from where you are sitting to the kitchen? →
2. What is the amount of change (money) you found under the sofa?
3. How much money do you have in your savings account? →
4. What is the sales tax in your city?

What is the data type here?

Can you ask a question?

Let us chit chat

1. Is it raining there? →
2. Have you paid the registration fee for this class? →
3. Are you in middle school?
4. Do you like Indian movies?

The answers are _____

Let us chit chat (mix of data types)

1. Tell me about yourself!   ['siva Jasthi', 59, 100.56, False]
    1. My name is siva Jasthi (string)
    2. I am 58 years old (int)
    3. I have 100.56 in checking account (float)
    4. I don't own a motorcycle (Boolean)
2. Tell me more about the movie you last watched.
3. Tell me about the book you last read

Four data types:

1. String (str)
2. Integer (int)
3. Float (float)
4. Boolean (bool)

Python Speak: An example

Python Speak

Hello. My name is **"string"** John. People call me **"list"** [John, Johnny, Joy]. My family includes "**list**" (My dad, mom, sister, me). Some of my hobbies are **"list"** [biking, basketball, dance, acting, reading]. It is **"boolean"** true that I love to watch movies! One of my favorite movies is **"string"** Psycho. I am "**float**" 5.6 in in height. My sister, who is **"integer"** 15 years old, also loves that movie. I speak **"integer"** 2 languages, **"list"** [Spanish, English], and I'm learning **"string"** French.

What could be the data type?

integer
float
string
boolean

list

1. Distance from your chair to the kitchen? 10.2
2. What is your height? 6.0
3. How many fingers are on your hand? 5
4. What are the last three movies you watched? [sitaramam, RRR, Pushpa]
5. Did he graduate? yes
6. Is the light switch on? on
7. Is the answer correct? true
8. Where do you live? Minneapolis
9. How many doors are on a car? 5
10. How many pages are there in that book? 256
11. How many chocolates are in a jar? 116
12. What are the last three placed you visited? [Museum, Park, Mall]

Can you ask a question?

```
integer
float
string
list
boolean
```

Try posing a question!

The answer should be one of these data types.

Variables: 3 Questions you must ask

[1] What is the **name** of the variable?

[2] What is the **value** of the variable?

[3] What is the **data type** of the variable?

Assigning Values to Variables

Variable: Something that changes

In python, we assign values to variables using = operator.

```
a = 13    (take the value of 13 and assign it a)
b = 18
c = b + 2  (take of the value of b, add 2 to it, and assign it back to c)
b = b + 2  (take the value of b, add 2 to it, and assign it back to b)
```

Left Hand Side & Right Hand Side

You take Right and put it in Left

Right is Right

Quiz Time

door_count = 4

Question	Answer
[1] What is the **name** of the variable?	
[2] What is the **value** of the variable?	
[3] What is the **data type** of the variable?	

Quiz Time

gpa = 3.9

Question	Answer
[1] What is the **name** of the variable?	
[2] What is the **value** of the variable?	
[3] What is the **data type** of the variable?	

Quiz Time

course = "Python101"

Question	Answer
[1] What is the **name** of the variable?	
[2] What is the **value** of the variable?	
[3] What is the **data type** of the variable?	

Quiz Time

id = "101"

Question	Answer
[1] What is the **name** of the variable?	
[2] What is the **value** of the variable?	
[3] What is the **data type** of the variable?	

Assignment Operator =

In python, we assign values to variables using = operator.

```
a = 2     (take the value of 2 and assign it a)
b = 3     (take the value of 3 and assign it to b)
c = b + 2  (take of the value of b, add 2 to it, and assign it back to c)
b = b + 2  (take the value of b, add 2 to it, and assign it back to b)



x = a + b + c    (explain this statement in plain English)
```

Variable Naming: Rules

- A variable name must start with a letter or the underscore character.

- A variable name cannot start with a number. However, it can have a number elsewhere in its name

- A variable name can only contain alpha-numeric characters and underscores (A to z, 0 to 9, and _)

- **Variable names are case-sensitive** (age, Age, and AGE are three different variables)

- Python Keywords can not be used as variable names.

Python Variable Naming: Do's and Don'ts

Python Variable Names: Do's and Don'ts

Do...	Don't...
Use meaningful names • first_name is much preferred over x	• Use single letter names • Use python reserved words (and, if, integer, float)
Use underscores to represent spaces: • total_score • overtime_hours	Use spaces: • The variable name "total score" will generate an error
Use lowercase letters for variable names • weekly_pay	Use capital letters....ever • The variable WeeklyPay is not "Pythonic"
Only use letters for variable names • full_name	Use special characters or numbers in variable names: • first&lastname • hours_over_40

https://www.youtube.com/watch?v=0Me6hxgT9oc

Python Key words

Here is a list of the Python keywords.   Enter any keyword to get more help.

False	def	if	raise
None	del	import	return
True	elif	in	try
and	else	is	while
as	except	lambda	with
assert	finally	nonlocal	yield
break	for	not	
class	from	or	
continue	global	pass	

How to name the variables?

Example: You want to capture the email of a student in a variable?
Which one of these would you like and why?

Variable Name	Is	
x	Va	
se	Va	
email	Va	
student	Va	
studentemail	Va	
STUDENTEMAIL	Va	
students_email	Va	
studentEmail	Va	
email_student	Va	
email_of_student	Va	
emailOfStudent	Va	
Student's email	In	
Student email	Invalid	It has a space

OK or Not OK

Variable Name	Is this legal (valid) name?	Is it good name?
movie_name		
movie name		
movieName		
1moviename		
1_movie_name		
_movie_name		
movie_name_1		
movie-name		
_movie_name_		
movie#name		
movie_name_*		
m		
n		
movie_name_		

Variables and Data Types

Based on the value, the type of variable is determined.

```
a = 13
b = 18


country_name = "India"
cost_of_masala_dosa = 4.5


is_it_raining = True
is_dog_hungry = False
```
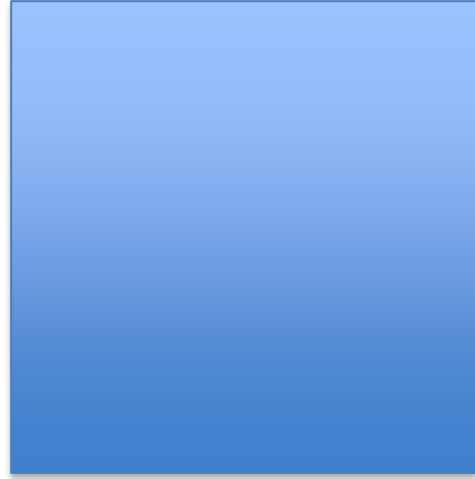
Can you guess the type of variable b? What about country_name?

Quiz Time

x = 1

x = 2

x = x + 1

x = x + 5

x = x * x

print(x)

What is the value of x after executing these statements?

Quiz Time

```
x = 25
y = 50
z = 75
x = y
y = z
z = x
```

Which of the variables have the value 50 after executing the code segment?

x only
y only
x and z only
x, y, and z

Let us assign those values to variables

506

True

56.78

"Joy"

'506'

Joy

Strings should always be enclosed in quotes!

Otherwise, python doesn't know that it is a string.

```
>>> a = 506
>>> b = 56.78
>>> c = True
>>> d = "506"
>>> e = "Joy"
.
```

```
>>> f = Joy
Traceback (most recent call last):
  File "<pyshell#93>", line 1, in <module>
    f = Joy
NameError: name 'Joy' is not defined
.
```

What is your **type( )**

You can use **type( )** function to know
the data type of a variable?

```
>>> a = 506
>>> b = 56.78
>>> c = True
>>> d = "506"
>>> e = "Joy"
.
```

```
>>> type(a)
<class 'int'>
>>> type(b)
<class 'float'>
>>> type(c)
<class 'bool'>
>>> type(d)
<class 'str'>
>>> type(e)
<class 'str'>
```

Given a value, can you tell its type?

Guess the data types for the following values

256	False
10.24	"cool"
"128"	Bestie

What could be the reason why Joy box is colored yellow?

Variables are chameleons

They not only change the value, but they can change types too.

```
...
>>> house_number = "2020"
>>> print(house_number)
2020
>>> print(type(house_number))
<class 'str'>
>>> house_number = 2020
>>> print(house_number)
2020
>>> print(type(house_number))
<class 'int'>
```

Variables are chameleons

The type of variable depends on the value it is assigned.

"Value" dictates the data type of a variable.

A given variable can change its datatype on the fly.

```
Python 3.6
(known limitations)

1   x = 10
2   print(type(x))
3
4   x = 5.6
5   print(type(x))
6
7   x = "python"
8   print(type(x))
9
10  x = True
11  print(type(x))
```

Print output (drag lower right corner to resize)

```
<class 'int'>
<class 'float'>
<class 'str'>
<class 'bool'>
```

Frames   Objects

Global frame

x True

print( ): It prints! Duh!

print( ) function is what we use to output variables.

```
# a is the age of Bob
a = 13

# b is the age of John
b = 18

#printing the value of a
print(a)

#printing the value of b
print(b)
```

Functions vs Variables

A "variable" has something

Examples of variables

name = "Jane"
marks = 90
is_used_car = True
balance = 67.9

A "function" does something. Functions need to be called (invoked). Functions usually take parameters. So, you should have ( ) when calling functions.

Examples of Functions

print() → for printing something
print("hello")
print(3+4)

type()   → for knowing the data type of a variable
type(a)
type("hi")

Comments

Once your program becomes longer, it becomes to difficult to read.

So, you need to provide the comments – indicating the purpose of the variables (and other blocks of code).

Revisiting the previous slide..

```
# a is the age of Anna
a = 13
# b is the age of Bindu
b = 18
```

White Space

We can also make liberal use of white space to make our programs readable.

For example, we can add an empty white line between the two variables

```
# a is the age of Anna
a = 13

# b is the age of Bindu
b = 18
```

Summary: What did we learn today?

Data Types: int, float, bool, str
Variables
Assignment Operator
Variable Naming Conventions: Do's and don'ts
Commenting and White Space
Functions vs Variables
type ( ) function
print() function
Comments and White Space

pythonTutor – Visualizing, Executing, Saving
colab

Self-paced learning

Visit www.w3schools.com/python

Python Tutorial

Python HOME

Python Intro

Python Get Started

Python Syntax

Python Comments

Python Variables

Python Data Types

Python Numbers

Python Casting

Python Strings

Python Booleans

Python Operators

Chapter 2 Summary

2. Input, Processing, Output

Basic Data Types
- int   float   str   bool

Advanced Data Types
- list   tuple   set   dict

Python Key Words (33)
- Value Keywords: True, False, None
- Operator Keywords: and, or, not, in, is
- Control Flow Keywords: if, elif, else
- Iteration Keywords: for, while, break, continue, else
- Structure Keywords: def, class, with, as, pass, lambda
- Returning Keywords: return, yield
- Import Keywords: import, from, as

Constants
- Values do not change
- Use all UPPER CASE (INTEREST_RATE)

Variables
- Values do change
- Use lowercase and pot hole case (student_name)
- Can't use Python keywords or spaces or symbols

Variables: Three things matter
- 1. name, 2. value, 3. data type   -
- type(var) is your friend to know the data type
- print(var) is your friend to know its value

Python goes from TOP to BOTTOM
- A = 10; B = 20
- C = A + B # OK
- X = A + B + D # Not OK

Python goes from RIGHT to LEFT during the assignment
- A = 2
- A = A + 1
- A = A * A

Assignment
- a = 10 (means "Assign the value of 10 to a"
- Short-hand assignment operator
  - a = 2
  - a += 2   # a = a + 2
  - a *= a   # a = a * a

Basic Functions
- input( ) to get inputs. Returns a string
- type( ) for knowing the type of a variable
- print( ) to display outputs to console
  - print(*args, sep = ' ', end = '\n')
  - Can print many arguments
  - Can print different data types
  - \n new line character (escape char)
  - \t tab character (escape char)

Data Conversion (Casting)
- From str to int - int("2")
- From int to str: - str(2)
- From str to float: float('2.3')
- From str to Boolean: bool('True')

Arithmetic Operators
- Addition (+)   Subtraction (-)
- Multiplication (*)   Modulus (%)
- Division
-     Float Division (/)
-     Floor Division (//) (integer; goes to small)
- Exponentiation (**)

Operator Precedence
- PEMDAS or GEMS
- If in doubt, throw in a parenthesis
- ** right binding; rest is left binding

Coding Conventions
- Commenting
- Empty lines to separate the blocks of code
- Consistency