



Basic Program Structure

Data Type, Variable - PYTHON



At the end of this lesson, you should be able to:

- Describe the following:
 - Variables in Python
 - Assignment operator in Python
 - Arithmetic operators in Python
 - Basic numeric data types in Python
- Use variables, assignment operator, arithmetic operators, and basic numeric data types in coding using Python

Topic Outline



Variables in Python



Assignment Operator in Python



Arithmetic Operators in Python



Basic Numeric Data Types in Python

Variables in Python

Names are used to make the program more readable, so that the “**something**” is easily understood.

e.g., **radiusFloat**

```
# 1. prompt user for the radius
# 2. apply circumference and area formulae
# 3. print the results

import math
radiusString = input("Enter the radius of your circle:")
radiusFloat = float(radiusString)
circumference = 2 * math.pi * radiusFloat
area = math.pi * radiusFloat * radiusFloat

print() # print a line break
print("The circumference of your circle is:", circumference, "\", and the area is:", area)
```

 *More on import, read input, and type conversion*

Identifier in Python

Identifier: a name given to an entity in Python

- Helps in differentiating one entity from another
- Name of the entity must be unique to be identified during the execution of the program



Rules for Writing Identifiers

What can be used?

- Uppercase and lowercase letters A through Z ($26 * 2 = 52$)
- The underscore, '_' (**1**)
- The digits 0 through 9, except for the first character (**10**)

$$52 + 1 + 10 = 63$$



Syntax Rules in Python

- Must begin with a letter or _
 - 'Ab123' and '_b123' are ok
 - '123ABC' is not allowed
- May contain letters, digits, and underscores

this_is_an_identifier_123
- Should **not** use keywords

- Upper case and lower case letters are different

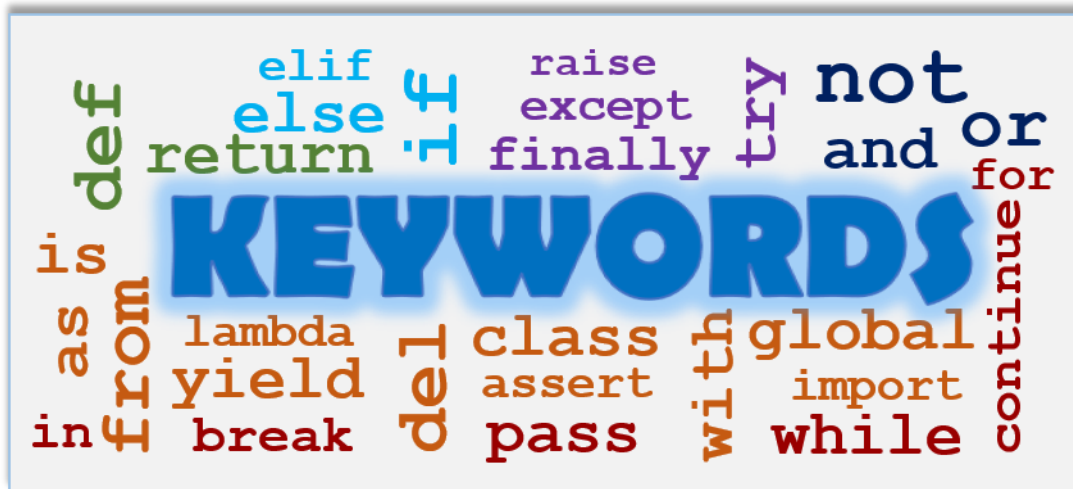
'LengthOfRope' is **not** 'lengthofrope'

 *Python is **case sensitive***

- Can be of any length
- Names starting with _ have special meaning

Keywords

- Special words reserved in Python
- Programmers should **not** use keywords to *name* things



Note: Old Python keyword '**exec**' was removed in Python 3



Let's examine the following variable names, which do you think are invalid?

int	return	For
Us\$	2person	userName
HALF_WINWIDTH	__name__	Phone#



Let's examine the following variable names, which do you think are invalid?

int	return	For
Us\$	2person	userName
HALF_WINWIDTH	__name__	Phone#

Allowed Characters: Uppercase and lowercase letters A through Z, the underscore, '_' and the digits 0 through 9 (except for the first character)

Should not use keyword

- (Us\$, Phone#): \$ and # are not allowed;
- (2person): a digit is not allowed as a first character
- (return): 'return' is a keyword

A Common Pitfall in Python

```
john_math_score = 90
peter_math_score = 70
mary_math_score = 80
john_eng_score = 60
peter_eng_score = 60
mary_eng_score = 60

total = john_math_score + peter_math_score + mary_math_score
average_math = total/3.0
print("average Math score =", average_math)
Total = john_eng_score + peter_eng_score + mary_eng_score
average_eng = total/3.0
print("average English score =", average_eng)
```



} All English scores are 60



Message 1

Be careful! Python is case sensitive!



Message 2

A program, that can run doesn't mean that it is correct.

Logic error



Can we interpret and run this program?

Yes



Is the result correct?



Hint: Typo

Python Naming Conventions

```
import math
radiusString = input("Enter the radius of your circle:")
radiusFloat = float (radiusString)
circumference = 2 * math.pi * radiusFloat
area = math.pi * radiusFloat * radiusFloat
```



VS.

```
import math
a = input("Enter the radius of your circle:")
b = float (a)
c = 2 * math.pi * b
d = math.pi * b * b
```

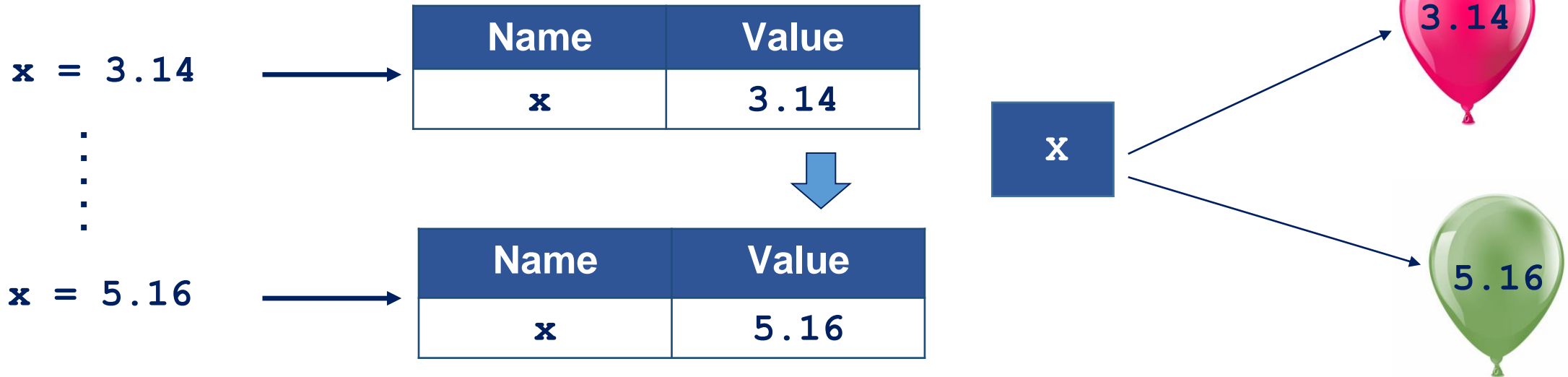


What is c? It is not immediately clear.

- Both programs work
 - They are different when **readability counts**
-
- variable names should be in lowercase, with words separated by underscores as necessary to improve readability
e.g. **radius_float**
 - mixedCase is allowed
e.g. **radiusFloat**

Operations

- Once a variable is created, we can **store**, **retrieve**, or **modify** the value associated with the variable name.
- Subsequent assignments can update the associated value.





What do you think is the output of the following Python code?

```
x = 9
print (x)
x = 7.8
print (x)
x = "welcome"
print (x)
```


Fun Guessing: Answer



What do you think is the output of the following Python code?

```
x = 9
print (x)
x = 7.8
print (x)
x = "welcome"
print (x)
```

Answer

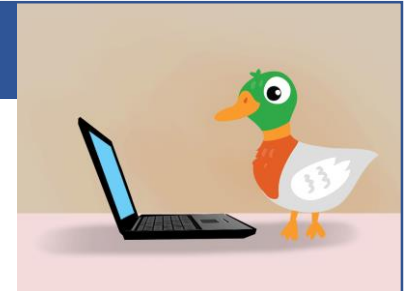
9
7.8
welcome



Compared to C and Java, how does Python know the data types?

Python uses *Duck-Typing*

“When I see a bird that walks like a duck and swims like a duck and quacks like a duck, I call that bird a duck.” – James Whitcomb Riley



Examples

```
>>> a = 99
>>> b = 99.9
>>> c = '100'
>>> d = True
```



Four variables!




What are their data types?

Data Types (Cont'd)

Type Function

In Python, the **type()** function allows you to know the type of a **variable** or **literal**.



```
>>> x = 9
>>> type (x)
<class 'int'>
>>> x = 7.8
>>> type(x)
<class 'float'>
>>> x = "Welcome"
>>> type (x)
<class 'str'>
>>> x = 'Python'
>>> type (x)
<class 'str'>
>>> type (8.9)
<class 'float'>
```

- Python does not have variable declaration, like Java or C, to announce or create a variable.
- A variable is created **by just assigning a value to it** and the type of the value defines the type of the variable.
- If another value is re-assigned to the variable, **its type can change**.

Data Types (Cont'd)

String - designated as **'str'**

- It is basically a sequence, typically a sequence of characters delimited by single quote ('...') or double quotes ("...")
- First *collection type* that was discussed
- Collection type contains multiple objects organized as a single object

 *More on this later..*



Examples

```
>>> a = "Length"
>>> b = "1003 welcome"
>>> c = "ewwew sdcd &8 $5##"
>>> d = 'ewwew sdcd &8 $5##'
```



What do you think is the output of the following Python code?

```
total = 4 + 3
sum = total * 2
Total = total + sum
print (total)
print ('Total')
```



Quick Check: Answer



What do you think is the output of the following Python code?

```
total = 4 + 3
sum = total * 2
Total = total + sum
print (total)
print ('Total')
```

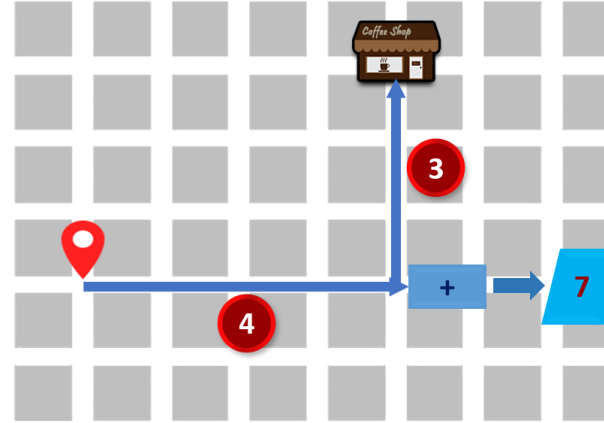
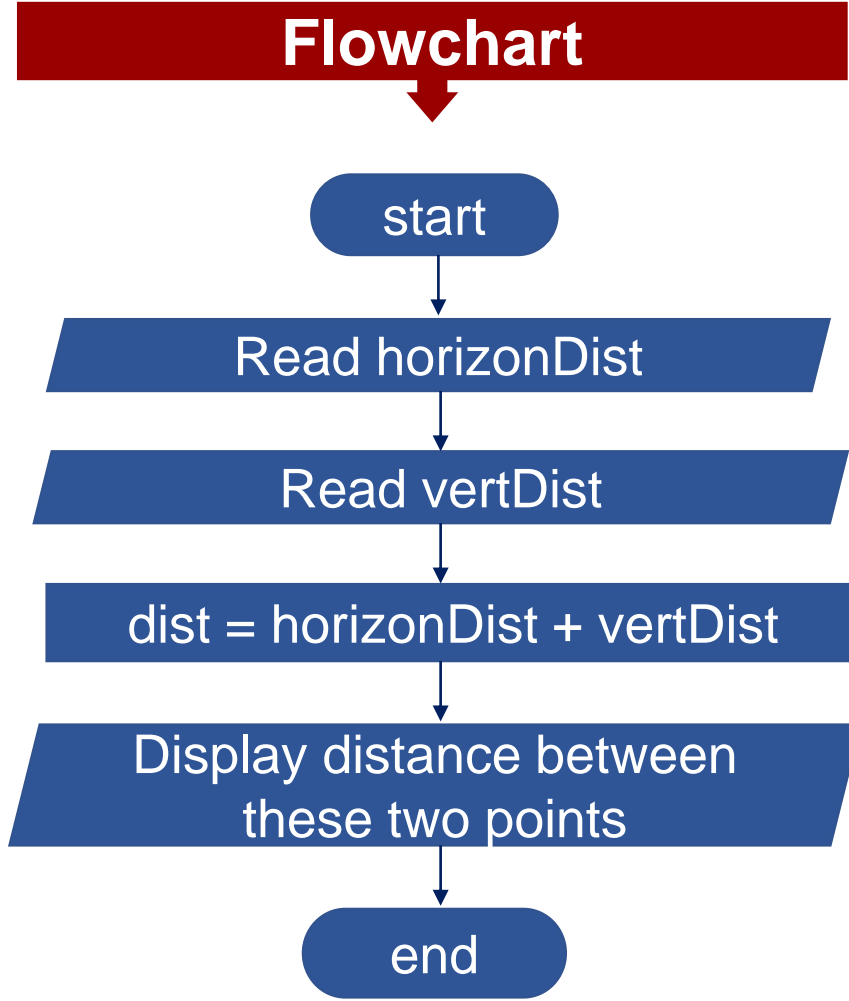


Answer

7
Total

Scenario 3: Find the Distance Traveled - Recall

Flowchart

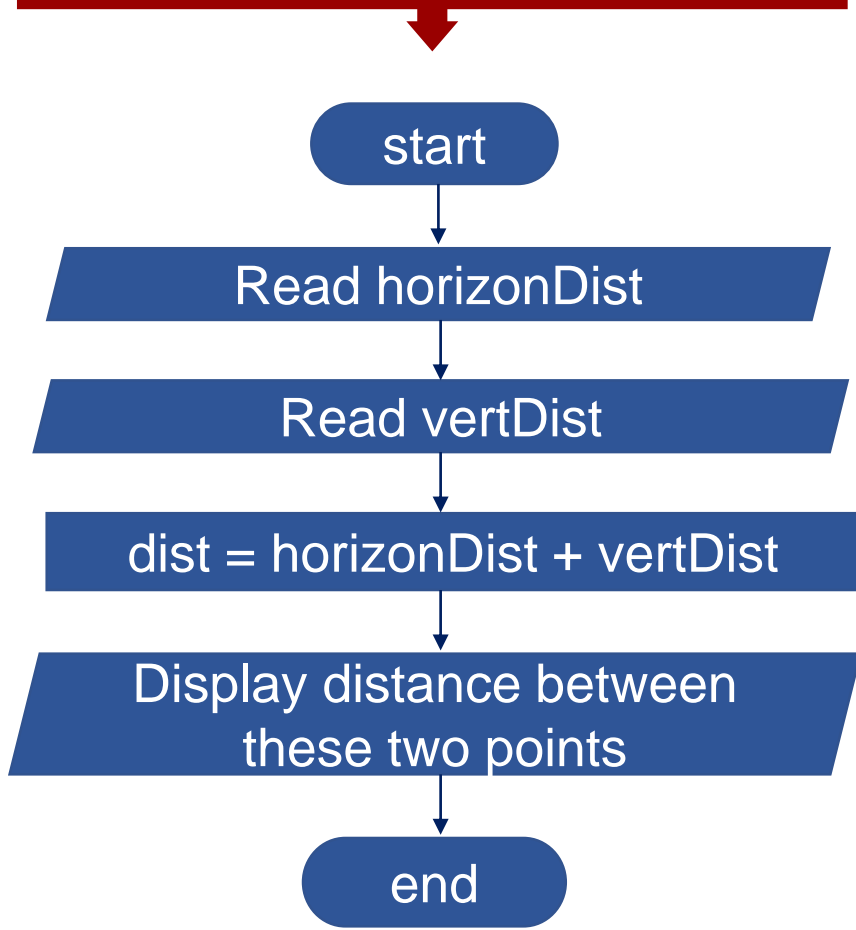


Preparatory Questions

- How many variables should you define? **(3)**
- What is the data type of each variable? **(integer)**
- Do you need assignment operator in your program? **(Yes)**
- Do you need arithmetic operators in your program? **(Yes)**

Scenario 3 - Python Codes

Flowchart



Python Code Version 1

```
horizon_dist = (int)(input("Read horizonDist"))
vertical_dist = (int)(input("Read vertDist"))
travel_dist = horizon_dist + vertical_dist
print("distance from A to B is ", travel_dist)
```

Output

Read horizonDist 4
Read vertiDist 3
distance from A to B is 7

print
(for displaying data)

input
(for reading data)

Scenario 3 - Python Codes: Comparison

Version 1

```
horizon_dist = 4
vertical_dist = 3
travel_dist = horizon_dist + vertical_dist
print(travel_dist)
```

Output: 7

Version 2

```
horizon_dist = 4
vertical_dist = 3
travel_dist = horizon_dist + vertical_dist
print("distance from A to B is ", travel_dist)
```

Output: distance from A to B is 7

Version 3

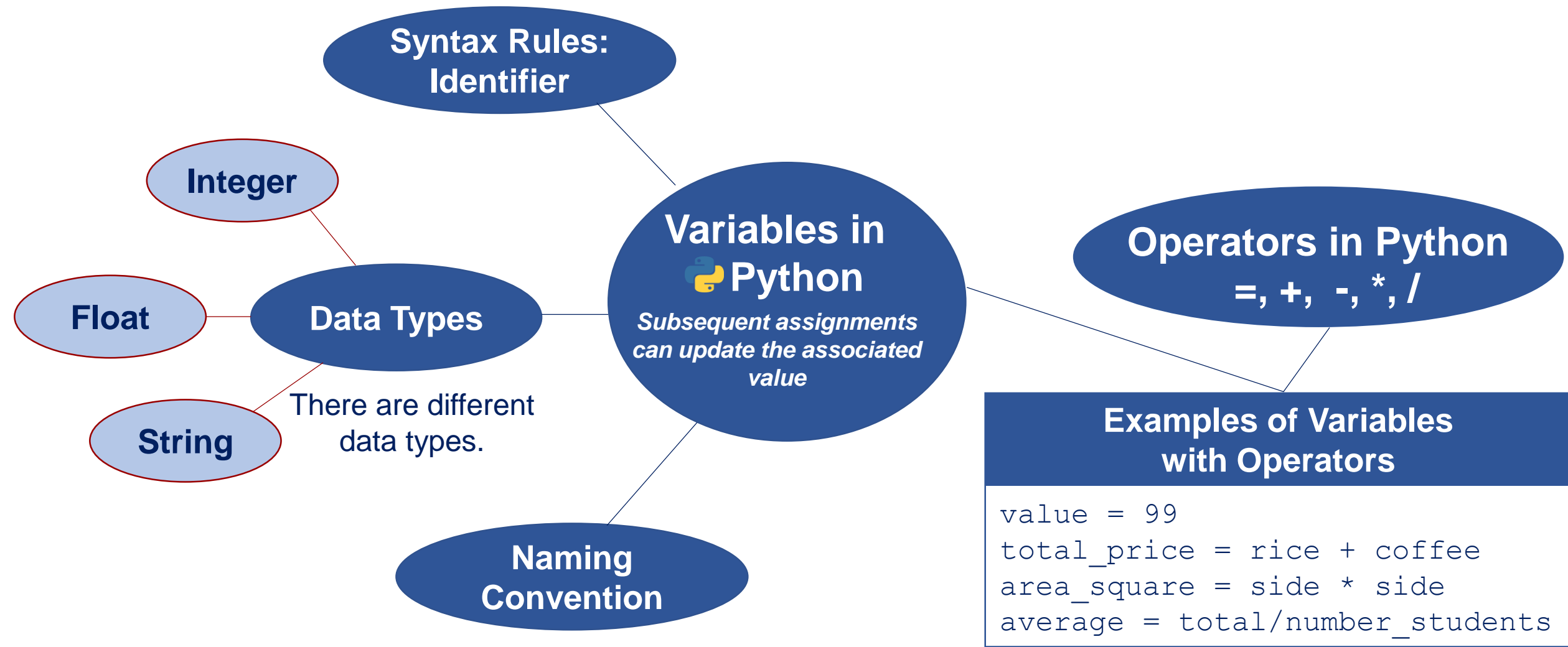
```
horizon_dist = (int)(input("Read horizonDist"))
vertical_dist = (int)(input("Read vertDist"))
travel_dist = horizon_dist + vertical_dist
print("distance from A to B is ", travel_dist)
```

Output:






Read horizonDist 4

Read vertDist 7




distance from A to B is 7



References for Images

No.	Slide No.	Image	Reference
1	All pages with Python codes		Python Logo [Online Image]. Retrieved April 18, 2018 from https://pixabay.com/en/language-logo-python-2024210/ .
2	5, 18		By User:Bobarino - Made by following Information.png, CC BY-SA 3.0, retrieved April 18, 2014 from https://en.wikipedia.org/w/index.php?curid=9180601 .
3	6		By Unknown - Io Interactive, Public Domain, retrieved April 18, 2018 from https://commons.wikimedia.org/w/index.php?curid=22908895 .
4	7		Warning [Online Image]. Retrieved April 18, 2018 from https://pixabay.com/en/warning-attention-road-sign-146916/ .
5	11, 16		Question problem [Online Image]. Retrieved April 18, 2018 from https://pixabay.com/en/question-problem-think-thinking-622164/ .

References for Images (Cont'd)

No.	Slide No.	Image	Reference
6	11		Light Bulb Idea [Online Image]. Retrieved April 18, 2018 from https://pixabay.com/en/light-bulb-idea-enlightenment-plan-1926533/ .
7	5		Balloon [Online Image]. Retrieved April 18, 2018 from https://pixabay.com/en/balloon-red-birthday-party-885715/ .
8	6, 18		Survey icon [Online Image]. Retrieved April 18, 2018 from https://pixabay.com/en/survey-icon-survey-icon-2316468/ .