3407001022021 Basic Information Technologies

İsmail Hakkı Parlak

ismail.parlak@ibu.edu.tr

Room: 335

https://github.com/ihpar/bit_chem

Variables

- Variables are containers for storing data values.
- Python has no command for declaring a variable.
- A variable is created the moment you first assign a value to it.

$x^2 = 2x$ $x = 2$ 3 $== 2+1$ ✓

- x = 4
  - Creates a variable named x. Stores 4 in x.
  - = operator should not be confused with == operator.
  - = is the assignment operator. Whereas == is the equality check operator.

Variables

- type(x) returns the type of the variable x.
- Variables can store any data type and be updated as many times as needed.
- x = 5
- x = 7
- $x = \boxed{x + 2}$
- y = "Hello world!"
- z = y

Variables

x = 2

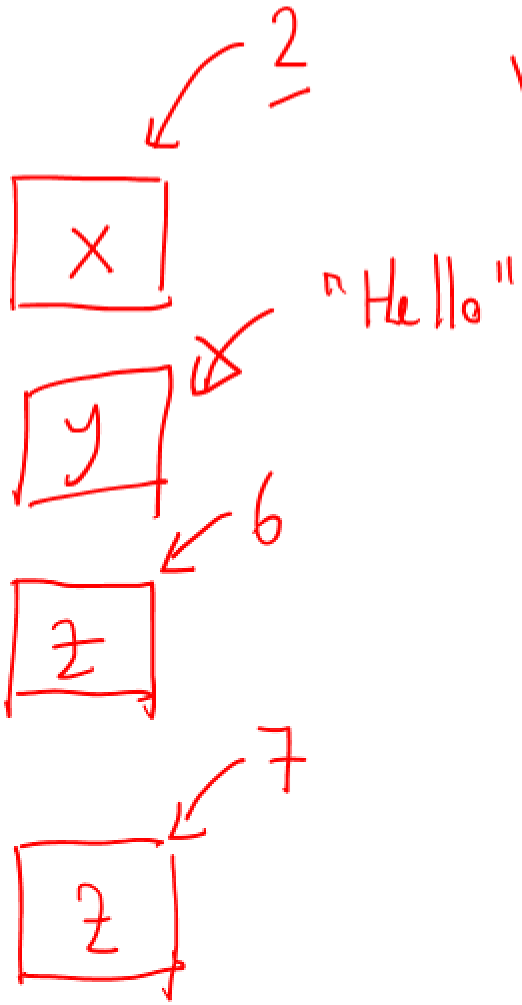
y = "Hello"

$z = \underbrace{x+4}_{2+4}$

6

$z = \underbrace{z+1}_{RHS}$

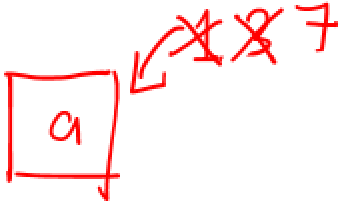
6+1



a = 1 ✓

a = 3 ✓

a = 7 ✓



x = 2

type(x) → int

a = "Hi"

type(a) → str

c = False

type(c) → bool

age = 10
print(Age)

x=5 ✓

y= "Hi" ✓

dateofbirth=2000

a = "ismail" ✓ date_of_birth=2000

name = "ismail" date of birth ✗

Variables

- A variable can have a short name (like x and y) or a more descriptive name (age, carname, total_volume).

age = 22

a = 22

- Rules for Python variables:
  - A variable name must start with a letter or the underscore character a.._z A.._Z _ _x , _name ✓
  - A variable name cannot start with a number 4a=5 ✗
  - A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _) ✓
  - Variable names are case-sensitive (age, Age and AGE are three different variables)
  - A variable name cannot be any of the Python keywords.

https://www.w3schools.com/python/python_variables_names.asp

Variables

- Variable names with more than one word can be difficult to read.
- There are several techniques you can use to make them more readable:
  - **Camel Case**
    - Each word, except the first, starts with a capital letter:
    - myVariableName = "John" ✓
  - **Pascal Case**
    - Each word starts with a capital letter:
    - MyVariableName = "John" ✓
  - **Snake Case**
    - Each word is separated by an underscore character:
    - my_variable_name = "John" ✓✓

https://www.w3schools.com/python/python_variables_names.asp

Variables

- The Python print() function is often used to output variables.

-
  ```
  x = "Python is awesome"
  print(x)
  ```
- In the print() function, you output multiple variables, separated by a comma:

-
  ```
  x = "Python"
  y = "is"
  z = "awesome"
  ✓ print(x, y, z) → "Python is awesome
  ```

https://www.w3schools.com/python/python_variables_output.asp

Strings

- Strings in python are surrounded by either single quotation marks, or double quotation marks.
- 'hello' is the same as "hello".
- You can display a string literal with the print() function:
- print("Hello")
print('Hello')

name = "ismail" ✓ " .... "

name = 'ismail' ✓ ' .... '

" .... "

"ismail' ✗

https://www.w3schools.com/python/python_strings.asp

Strings

'It's good'

→

"It's good"

''

- You can use quotes inside a string, as long as they don't match the quotes surrounding the string:
- print("It's alright") ✓
  print("He is called 'Johnny'") ✓
  print('He is called "Johnny"') ✓
- You can assign a multiline string to a variable by using three quotes:
- a = """Lorem ipsum dolor sit amet,
  consectetur adipiscing elit,
  sed do eiusmod tempor incididunt."""
  print(a)

https://www.w3schools.com/python/python_strings.asp

Strings

length

len(a) → 13

- To get the length of a string, use the len() function.
- a = "Hello, World!" (1)
  print(len(a))

print(len(a))

print(13) → 13

- To check if a certain phrase or character is present in a string, we can use the keyword in.
- txt = "The best things in life are free!"
  print("free" in txt) → True
- To check if a certain phrase or character is NOT present in a string, we can use the keyword not in.
- txt = "The best things in life are free!"
  print("expensive" not in txt) → True

https://www.w3schools.com/python/python_strings.asp