Variables and Data Structures

Variables

Allowed characters:
letters
digits
underscore

Exceptions:
no starting digit
no keywords

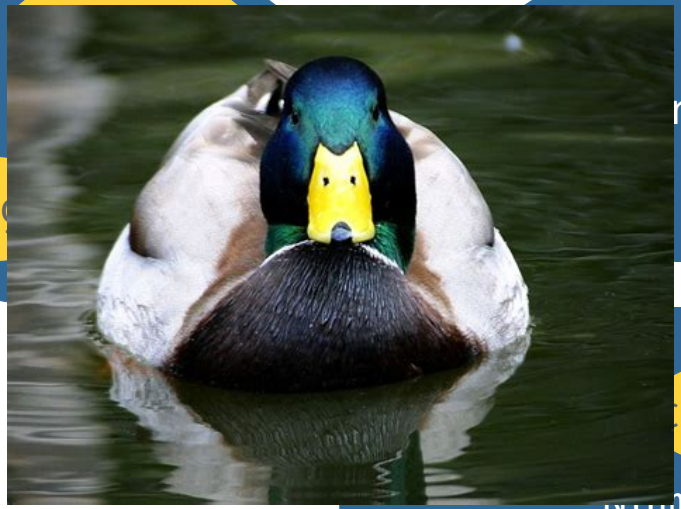
```
Python 3.8.5 (default, Sep  4 2020, 07:30:14)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> variable = 5
>>> print(variable)
5
>>> CONSTANT = 10
>>> variable + CONSTANT
15
>>> variable = variable + CONSTANT
>>> variable
15
>>> CONSTANT = 1
>>> CONSTANT
1
>>>
```

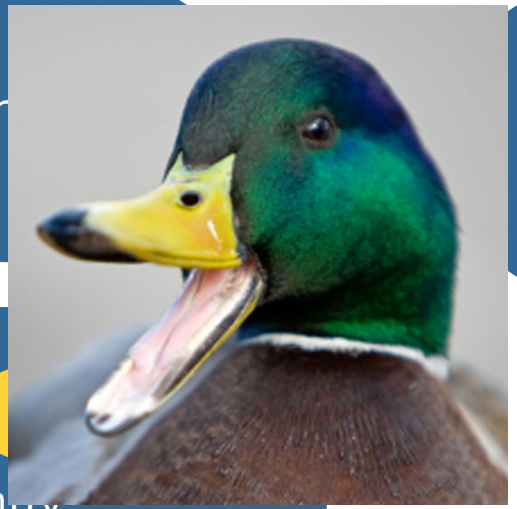
Keywords

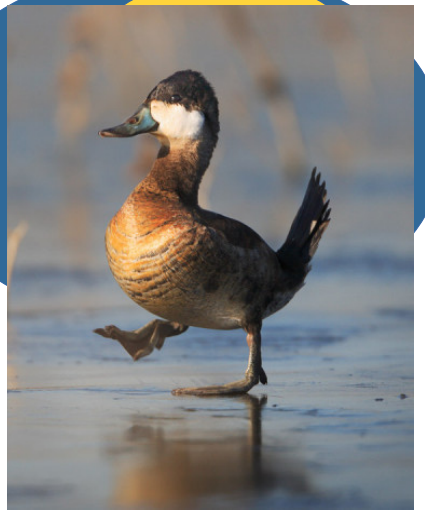
False, None, True, and, as, assert,
async, await, break, class,
conti
except,
if,
nonlocal, not, or, pass, raise,
return, try, while, with, yield

```
>>> True = 234
  File "<stdin>", line 1
SyntaxError: cannot assign to True
```

Data Types







float64

...

Data Structures

```
>>> dictionary = {"key": "value"}
>>> dictionary = {"Something": 1, "Another Thing": [1, 2]}
>>> dictionary = {"First": {"Second": {"Third": {"Fourth": 4}}}}
>>>
```

```
>>> array = [1, 2.1, "adc"]
>>> array = [1, 2, 2, 1]
>>> array = [[3, 7], [1, 3]]
>>>
```

```
>>> set = {"a", 1, 3.2}
>>> set = {1, 4, 5, 5}
```

```
>>> tupel = (1,2,3)
>>> tupel = ("a", 1)
>>> tupel = ("c",)
```

```
>>> import numpy
>>> np_array = numpy.array([1,2,3,4])
>>> np_array = numpy.array([[1, 2], [3,4]])
>>> np_array = numpy.array(["a", 1, 5.3])
>>>
```

Dictionaries

```
>>> your_data = {"Time": 12.30, "Sample": "Sample 12", "Measurement": [13, 2, 44, 56]}
>>> your_data
{'Time': 12.3, 'Sample': 'Sample 12', 'Measurement': [13, 2, 44, 56]}
>>>
```

your_data	
“Time”	12.30
“Sample”	“Sample 12”
“Measurement”	[13, 2, 44, 56]

Key

Value

```
>>> your_data["Time"]
12.3
>>> your_data["Sample"]
'Sample 12'
>>> your_data["Measurement"]
[13, 2, 44, 56]
>>> your_data["Time"] = 14.00
>>> your_data["Time"]
14.0
>>> your_data["Day"] = "Wednesday"
>>> your_data
{'Time': 14.0, 'Sample': 'Sample 12', 'Measurement': [13, 2, 44, 56], 'Day': 'Wednesday'}
>>>
```

Arrays

```
>>>
>>> Measurement = [13, 2, 44, 56]
>>> Measurement
[13, 2, 44, 56]
```

Measurement	
0	13
1	2
2	44
3	56

Index

Value

```
>>>
>>> Measurement[0]
13
>>> Measurement[1]
2
>>> Measurement[1] = 100
>>> Measurement[1]
100
>>> Measurement.append(53)
>>> Measurement
[13, 100, 44, 56, 53]
>>>
```

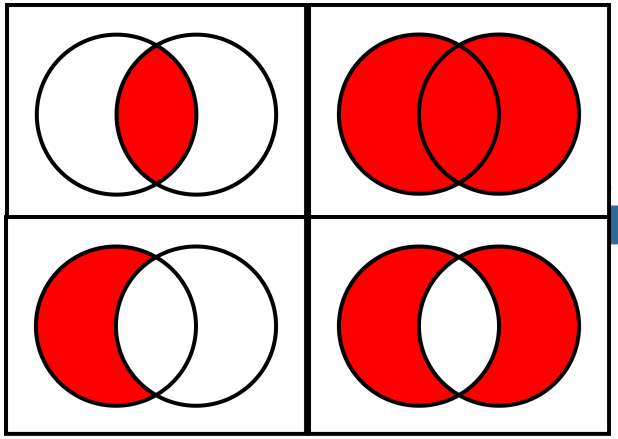
Sets

```
>>> your_set = {"a", 24, "a", 3.151492}
>>> your_set
{24, 'a', 3.151492}
```

your_set
“a”
24
“a”
3.151492

your_set
“a”
24
3.151492

Value



Use them like mathematical sets.

No Index
Elements unchangable

Tupels

```
>>>
>>> tupel = (1, "aa", 1, 2.75)
>>> tupel
(1, 'aa', 1, 2.75)
>>>
```

tupel	
0	1
1	"aa"
2	12.75
3	56

Index Value

```
>>> tupel[0]
1
>>> tupel[1]
'aa'
>>> tupel[1] = 1
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'tuple' object does not support item assignment
>>>
```

Numpy Arrays

numpy_array	
0	0
1	1
2	2
...	...

Index

Value

```
>>> import numpy
>>> numpy_array = numpy.array([0,1,2,3,4,5])
>>> numpy_array
array([0, 1, 2, 3, 4, 5])
```

```
>>> numpy_array[0]
0
>>> numpy_array[1]
1
>>> numpy_array = numpy.append(numpy_array, 6)
>>> numpy_array
array([0, 1, 2, 3, 4, 5, 6])
```

```
>>> matrix = numpy.array([[1, 2], [3, 4], [5, 6]])
>>> matrix
array([[1, 2],
       [3, 4],
       [5, 6]])
>>> matrix.shape
(3, 2)
>>> matrix.reshape(2,3)
array([[1, 2, 3],
       [4, 5, 6]])
```

```
>>> numpy.array(["22", 4])
array(['22', '4'], dtype='<U2')
>>> numpy.array([True, 23])
array([ 1, 23])
```

```
>>> vector = numpy.array([1,2,3,4])
>>> vector
array([1, 2, 3, 4])
>>> vector.shape
(4,)
>>> vector.reshape(1,4)
array([[1, 2, 3, 4]])
>>> vector.reshape(-1,1)
array([[1],
       [2],
       [3],
       [4]])
>>> vector.reshape(2,2)
array([[1, 2],
       [3, 4]])
```

Strings

Data Type no Data Structure, but is similar to arrays

```
>>> string = "Hello!"
>>> string
'Hello!'
>>>
```

string	
0	“H”
1	“e”
2	“l”
...	...

Index | Character

```
>>> string[0]
'H'
>>> string[1]
'e'
>>> string + " Hello?"
'Hello! Hello?'
>>> " ".join([string, "Hello?"])
'Hello! Hello?'
>>> f"{string} Hello?"
'Hello! Hello?'
>>>
```

Slicing

Slicing works for: Strings, Arrays, Numpy Arrays, Tupels

array	
0	...
1	...
2	...
3	...
4	...
5	...
6	...
7	...
8	...
9	...

```
>>>
>>> array = [0,1,2,3,4,5,6,7,8,9]
>>> array[0]
0
>>> array[-1]
9
>>> array[0:2]
[0, 1]
>>> array[2:6]
[2, 3, 4, 5]
>>> array[2:6:2]
[2, 4]
>>> array[::-1]
[9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
>>> array[:]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

array[start:end:step]

array_copy	
start	...
start + step	...
start + 2*step	...
...	...
end-1	...

Simple In- & Output

input() reads a string from the console
print() prints a string to the console

```
>>>
>>> user_input = input("Would you answer this question?")
Would you answer this question?Yes.
>>> print(user_input)
Yes.
>>> type(user_input)
<class 'str'>
```

```
>>> number = input("What's your favourite number? ")
What's your favourite number? 8
>>> type(number)
<class 'str'>
>>> number = int(number)
>>> type(number)
<class 'int'>
```

Exercise 2: Hello, You!

Ask the users name and generate a customised greeting.

Ask the users age and store it as integer.

Print only the middle letters of the name and spell it backwards.