Learn to code

in Python 3

Setup

- Login to your REDACTED
  - REDACTED
  - REDACTED
  - REDACTED
  - REDACTED
- Launch IDLE
  -  -> All Programs -> Python 3.x -> click **IDLE (Python GUI)**
- Create Module
  - File | New File
- No IDLE? Go to http://www.ideone.com
  - Choose Python 3

'Hello world'

- Shell (>>>):

  ```
  print('Hello world')
  ```

  Enter

- Module (.py):

  ```
  print('Hello world')
  ```

  Run | Run Module | OK (to save) | 'test.py'

Data types

- **int** → 3 and **float** → 3.14
  operators: + - * / // % **
- **str** → 'Hi' ; "Hi" ; " 'Hi' " ; ' "Hi" ' ; "" 'Hi' ""
  operators: + * [ ] [:]
- **list** → ['Hello', 3, 3.14] and **tuple** → ('Hello', 3, 3.14)
  operators: + * [ ] [:]
- **bool** → True ; False
  operators: or and not is

Exercise 1 (shell)

1. extract the 'y' from 'Python'
2. turn 'echo' into 'echoechoecho'
3. 3.14 to the 4^{th} power
4. extract the first 3 letters from ['a','b','c','d','e']
5. merge [1, 2] & [3, 4] into [1, 2, 3, 4]

Boolean expressions

- Membership:

  'a' in 'abacus' ; 1 not in ['Hi', 2, 3.14] ; 1 in 31?

- Identity:

  3 is not 3.0 ; 'a' is not 'A' ; ['Hi', 1] is not [1, 'Hi']

- Equality:

  3 == 3.0 ; 'a' != 'A' ; ['Hi', 1] != [1, 'Hi']

- Comparison:

  4 > 3 ; 3 <= 3.0 ; 8 > 6 > 2 ; 'abacus' < 'zero'

Exercise 2 (shell)

1. The word 'Hello' contains the letter 'z'
2. The number 7 is equivalent to the number 7.0
3. The opposite of True is False
4. 'c' is closer to the end of the alphabet than 'a'
5. 'hi' is contained in 'Ohio' and 4 is greater than 0

Variables

✓ my_python_variable

× myPythonVariable

- first = 'Joe' (NOT first == 'Joe')
  last = 'Smith'
  full = first + ' ' + last
  record = (last, first, full)

- a = 3 b = a
  a = 4
  b == ?

Built-in functions

- Console input:

```
username = input('Enter your name: ')
```

- Console output:

```
print('Hello', username)
```

- Type conversion:

```
str(3)  ;  int('3')  ;  list('3')  ;  int([3])?
```

- Length:

```
len('Hello')  ;  len([1, 2, 'Hi', 3.14])  ;  len(31)?
```

Exercise 3 (module)

1. Prompt the user for a word, bind a variable to it, and print the variable. Run the module.

2. Prompt the user for their name and tell them how long their name is (including spaces). Run the module.

3. Prompt the user for two numbers. Print the result of adding the two numbers together. Run the module.

Conditional statements

-
```
if a == b:
    print('Correct')
```

-
```
if a == b:
    print('Correct')
else:
    print('Incorrect')
```

-
```
if a == b:
    print('Correct')
elif a > b:
    print('Too high')
else:
    print('Too low')
```

User-defined functions

- ```
  def print_name():
      print('Joe')
  ```

- ```
  def print_name(name: str):
      print(name)
  ```

- ```
  def get_name() -> str:
      name = input('Please enter your name: ')
      return name
  ```
  ```

Exercise 4 (module)

1. Prompt the user for a number. If the number is negative, print 'Error'. For any other value, print 'OK'. Run the module.

2. Define a function which prompts the user for two words and returns a list containing the two words. Call the function and print its return value. Run the module.

Loops

- ```
  for v in ['a', 'b', 'c', 'd']:
      print(v)
  ```

- ```
  for c in 'Hello world':
      print(c)
  ```

- ```
  for i in range(0,10):
      print(i)
  ```

- ```
  num = 50
  while num != 0:
      print(num)
      num = num // 2
  ```

- ```
  while True:
      n = input('Name: ')
      if n != "":
          print('Hi', n)
      else:
          break
  ```
  ```

Exercise 5 (module)

1. Using a loop, print the following text to the screen:

   0 Hello world
   1 Hello world
   2 Hello world
   3 Hello world

2. Using a loop, print the following text to the screen:

   2
   err
   is
   human

Methods

- str methods:

```
'This is a sentence'.split(' ') == ['This', 'is', 'a', 'sentence']
' + '.join(['John', 'Jill', 'Jane']) == 'John + Jill + Jane'
'LoWEr'.lower() == 'lower'
'Mississizppi'.find('z') == 8
```

- list methods:

```
[1, 2, 3, 4, 1, 5, 1, 4, 7].count(1) == 3
[1, 2, 3, 4, 1, 5, 1, 4, 8].sort() ---> [1, 1, 1, 2, 3, 4, 4, 5, 8]
['John', 'Jill'].append('Jack') ---> ['John', 'Jill', 'Jack']
['John', 'Jill', 'Jane'].remove('Jill') ---> ['John', 'Jane']
```

== is equivalent to (returns boolean)

---> is mutated to (no return value)

A simple utility

```
def count_vowels(string: str) -> int:
    num_vowels = 0
    vowels = ['a', 'e', 'i', 'o', 'u']
    for c in string:
        if c.lower() in vowels:
            num_vowels = num_vowels + 1
    return num_vowels

while True:
    string = input('Please enter a string: ')
    if len(string) > 0:
        print(string, '--->', count_vowels(string), 'vowels')
    else:
        break
```

Wrap up

- Disconnect from REDACTED
  - click REDACTED
  - click **OK**
- REDACTED
  - REDACTED
- Questions?