# INF 502 – SOFTWARE DEVELOPMENT METHODOLOGIES

Week 2

Part II – Python basics

NAU NORTHERN ARIZONA UNIVERSITY

# First things first…

- Connect to iClicker for in-class participation and attendance

https://join.iclicker.com/7YDD3

# Welcome to Python

To install Python:
https://www.python.org/downloads/

You can write your code on any text editor you may have installed such as vim, vi, Notepad++, Sublime Text, etc

But…

To install PyCharm:
https://www.jetbrains.com/pycharm/download

# Kicking Off

Python: Interpreted language

Interactive editor

INF502 — Python — 61×13

```
cmpMXQFTQCMYV:INF502 acs549$ python3
Python 3.10.6 (v3.10.6:9c7b4bd164, Aug  1 2022, 17:13:48) [Cl
ang 13.0.0 (clang-1300.0.29.30)] on darwin
Type "help", "copyright", "credits" or "license" for more inf
ormation.
>>> print("hello world!")
hello world!
>>>
```

# Kicking Off

Python: Interpreted language

Running from a file

HelloWorld.py UNREGISTERED

```
1 print("hello world!")
```

Line 1, Column 20 Tab Size: 4 Python

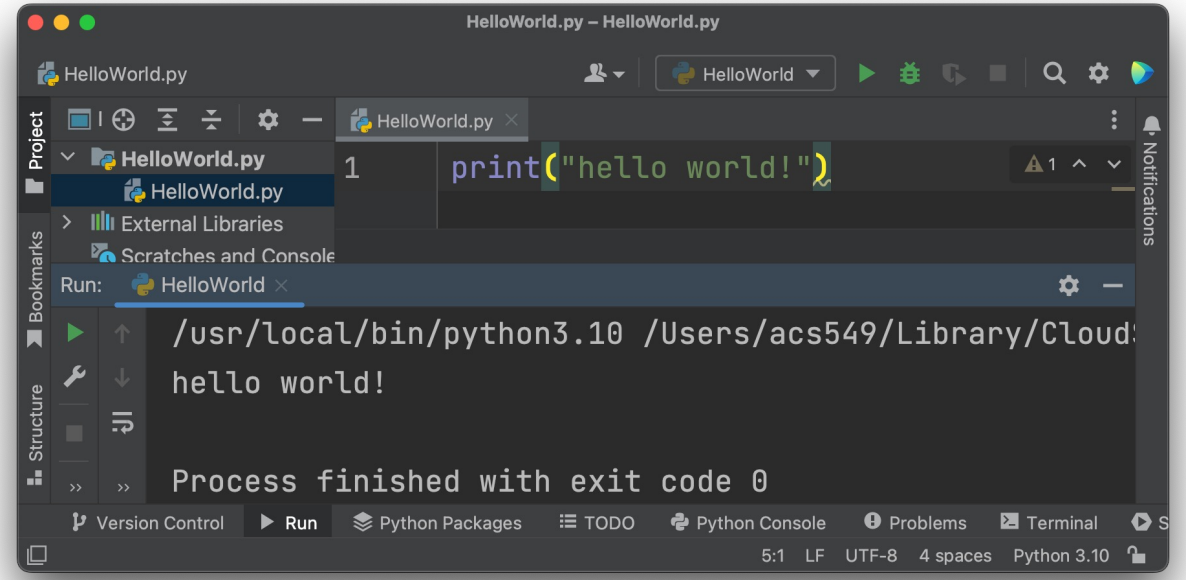
INF502 — -bash — 52×5

```
cmpMXQFTQCMYV:INF502 acs549$ python3 HelloWorld.py
hello world!
cmpMXQFTQCMYV:INF502 acs549$
```

# Kicking Off

Python: Interpreted language

Using an IDE such as Anaconda or PyCharm



# Formatting

Python uses **indentation** to delimit blocks of code

Comments start with #

Colons “:" are used to start a new block for different constructs

```
# function that answers if a
# number is even or not
def isEven (number):
   #is the remainder when
   #dividing number by 2 equals to 0
   if (number % 2 == 0):
      #yes, the number is even
      return True
   return False
```

# Variables

```
>>> x = 2        #x is an integer
>>> y = 'Igor' #y is a String
>>> y = 2.5      #y is now a floating point
>>> z = [1,2,3] #z is a list
>>> #each position in z is an integer
>>> type (z)
<class 'list'>
>>> x = y = z = 4 #chained assignment
```

- Variables are created when they are assigned a value
- Type-binding is associated 'on-the-fly'

# Arithmetic

- Mathematics apply

```
>>> a = 2+3      #a is 5
>>> b = 7-4      #b is 3
>>> c = 2*2.5    #c is 5.0 (float!)
>>> d = 2**4     #d is 16
>>> e = 5%2      #e is 1
>>> f = 7/2      #f is 3.5
>>> g = 7//2     #g is ...
```

# Strings

- Strings are delimited by single or double quotation marks

```
>>> single_quote = 'python'
>>> double_quote = “python"
>>> I_want_the_quote = 'It\'s python'
>>> yes_the_quote = “It's python"
>>> multi_line = 'this is a multi line \
sentence. It is possible to break it in \
multiple lines.'
```

# Strings

- Operations using Strings

```
#operations using strings
>>> salutation = "Hello"
>>> name = "John"
>>> complete_salut = salutation + ', ' + name + '!'
>>> print (complete_salut)
Hello, John!
```

# Getting user inputs is usually important

```
#input() function waits for an input from the keyboard
>>> salutation = "Hello"
>>> name = input("Tell me your name: ")
Tell me your name: <INPUT + ENTER>
>>> complete_salut = salutation + ', ' + name + '!'
>>> print (complete_salut)
Hello, <NAME ENTERED>!

>>> weight = input("Enter your weight in lb: ")
Enter your weight in lb: 200
>>> weight = int(weight) #what am I doing here???
>>> weight_kg = weight/2.205
>>> print (weight_kg)
90.70294784580499
```

# 4-minute madness

Write a program that prompts the user for the length of the two different sides of a rectangle (a and b). Print the area and the perimeter of the rectangle.

Write a program that prompts the user for a distance in kilometers and print out the converted distance in miles.

# iClicker time

Predict the output of following python program:

A. 5

B. 4

C. 9

D. 13

E. None of the above

iclicker1.py UNREGISTERED

HelloWorld.py | iclicker1.py

```python
x = 5
y = 4
z = y
x = x + z
print(z)
```

Line 6, Column 1 Tab Size: 4 Python

# Conditional

- if – else (and more…)

```
if (x > y):
   print (“x is greater than y")
elif (y < x):
   print (“y is greater than x")
else:
   print (“they are equal!")
```

```
parity = "even" if x % 2 == 0 else "odd"
```

# Comparison Operations

| Operation | Meaning |
|---|---|
| < | strictly less than |
| <= | less than or equal |
| > | strictly greater than |
| >= | greater than or equal |
| == | equal |
| != | not equal |
| is | object identity |
| is not | negated object identity |

# Less Suffering

Let's use files instead of Interactive Prompt

Create a file with the extension .py

- To run:
- python3 <filename>.py

# 3-minute madness

Create a Python file called angles.py. In this file, write a code that prompts the user an angle and prints out:

- "The angle is acute" → if the informed angle is less than 90°
- "The angle is right" → if the informed angle is equal to 90°
- "The angle is obtuse" → if the informed angle is greater than 90°

Run your file to test the outcomes!

# Be ready for what's next…

Watch for the HW1 due date

Python essentials (part 2)