Introduction to Python

Fondren Library
Research Data Services

100 SECONDS OF

Terminology

Library: A library is a collection of modules that includes pre-written code (code blocks) to help with common tasks.

Modules: Established and External Tools that serves a specific function, such as file handling, data analysis, web development.

Import: Allows one to bring in modules to execute a specific module or library (package) in your coding environment (env).

Function: Is a reusable block of code that performs a specific task. Functions input, process, and return a specific output.

Variable: A variable is a name that refers to a value that is stored overtime in a computer's memory. It is used as a label for a piece of data, that allows you to manipulate the data in your code.

Classes (Python): Define what objects are used for, and also, the types of behaviors and processes that can be executed by an object.

Objects: An instance of a class (modules, variables, etc), that has an attached set of attributes and behaviors, with specific types of conditions that can be executed from that block of code.

DATA TYPES

Python knows various types of data. Common ones are:

- Strings – “a”, “hi”
- Integer numbers – 2, 4, 6,
- Floating point numbers – 3.14, 2.0, 2.12
- Boolean – True/False

Strings - Used to represent text in our coding blocks.
Ex. print(“Hello World”)

Integers (int) - Whole Numbers that don’t have fractional components (ex. No decimals or fractions).
Ex. sum = 42 + 76
print(sum)

Floating Point Numbers (floats) - are used to represent numbers with decimal points (also show fractional numbers such as 1/4, 5/2, 8/10).
Ex. pi = 3.14
print(pi)

Boolean Values (bool/bools) - True or False Values | Computer represents as true or absent of true values (false)
Ex. is_student = False
print(is_student)

CHECK AND CHANGE TYPES
BUILT-IN FUNCTION type

In [50]:
```
1 print(type(int("3")))
2 print(str(3))
3 print(float(3))
```

```
<class 'int'>
3
3.0
```

Operands -
Characters, symbols, strings, objects that are being operated on

Arithmetic Operators

Operator	Meaning	Example
+	Addition	4 + 7 ⟶ 11
-	Subtraction	12 - 5 ⟶ 7
*	Multiplication	6 * 6 ⟶ 36
/	Division	30 / 5 ⟶ 6
%	Modulus	10 % 4 ⟶ 2
//	Quotient	18 // 5 ⟶ 3
**	Exponent	3 ** 5 ⟶ 243

USE AS A CALCULATOR
MATHEMATIC OPERATORS

+, -, /, *, %

```
In [1]:  1 3+4
         2
Out[1]: 7
```

```
In [2]:  1 ((5+6-1)*2/5)**2
         2
Out[2]: 16.0
```

Exercise 1:
Radius=5, π=3.14, calculate the area of the circle

```
radius = 5
pi = 3.14159265359
area = pi * radius * 2
print("The area of
the circle is:", area)
```

OPERATORS WORK DIFFERENTLY BASE ON DATA TYPE

```
In [ ]: 1 print("Hello"+"World")
```

```
In [ ]: 1 print("Hello"*3)
```

```
In [46]: 1 print("Hello"+"World")
HelloWorld
```

```
In [2]: 1 print("Hello"*3)
HelloHelloHello
```

Working with Strings

len(“word”)

Word.capitalize()

Word.upper()

Word.lower()

Word.title()

LOGIC OPERATORS <, >, ==, !=, <=, >= AND STATEMENTS and, or, not RETURN BOOLEAN DATA TYPE

```
In [10]: 1 3>7

Out[10]: False

In [11]: 1 True and False

Out[11]: False

In [12]: 1 True or False

Out[12]: True

In [13]: 1 not True

Out[13]: False
```

Logical Operators/Statements (Relationships between operands or values)

ASSIGNING VARIABLES

In [28]:

```
a = 3
b = 4

c = a + b
d = a*b + c
e = a**b/c

print (c)
print (d)
print (e)

```

```
7
19
11.571428571428571
```

Built-In Function: INPUT

```
How old are you?
```

```
How old are you? 6
```

```
How old are you?6
Your age is  6
```

```
In [20]:
1 Age = input("How old are you?")
2 print ("Your age is ",Age)
```

Exercise 2:
Create a variable affiliation, prompt a question, "Are you a student or a staff member?"

print "You are a " + input

USER-DEFINED FUNCTION

In [26]:
```
def C_F(C):
    F = 1.8*C+32
    return F

temp = C_F(20)
print(temp)

```

```
68.0
```

Exercise 3: Create a BMI function and calculate BMI for person1 and person2. BMI = weight/height²

person1: height:1.65m, weight:60kg
person2: height:1.75m, weight:75kg

```
def BMI(H,W):
    bmi=

person1 =
person2 =
print(person1)
print(person2)
```

BASIC DATA STRUCTURES IN PYTHON

- Lists [1,2,3] ordered and changeable
- Tuples (1,2,3) ordered and unchangeable
- Dictionary {‘a’: 1, ‘b’:2, ‘c’:3} changeable, key-value pairs

LIST

Create a list:

```
1 mylist = ['apple', 'orange', 'banana']
2 print (mylist)
```

```
['apple', 'orange', 'banana']
```

Access item:

```
1 mylist = ['apple', 'orange', 'banana']
2 print (mylist[1])
```

```
orange
```

Change Item Value:

```
1 mylist = ['apple', 'orange', 'banana']
2 mylist[1] = 'cherry'
3 print (mylist)
```

```
['apple', 'cherry', 'banana']
```

Add Items:

```
1 mylist = ['apple', 'orange', 'banana']
2 mylist.append('pear')
3 print(mylist)
```

```
['apple', 'orange', 'banana', 'pear']
```

Remove Items:

```
1 mylist = ['apple', 'orange', 'banana']
2 mylist.remove('apple')
3 print(mylist)
```

```
['orange', 'banana']
```

Exercise 4:

1) Create a list of your favorite songs, print the list
2) Print the 3rd item in the list
3) Change the 3rd item into another song
4) Add one more song
5) Remove one song

CONTROL FLOW – IF/ELSE

In [31]:

```
1 GPA = 4.0
2 if GPA >3.8 and GPA <= 4.0:
3     print ("Welcome to Rice!")
4 elif GPA <= 3.8:
5     print ("Sorry")
6 else:
7     print ("Ooops, type a GPA in range")
```

```
Welcome to Rice!
```

Exercise 5: Create a variable called "behavior", assign a value "good" to it

if "good" print "candy"
elif "bad" print "no candy"
else print "ask your mom"

```
def bark():
```

CONTROL FLOW – FOR LOOP

```
1 for x in range(1,6):
2     print (x)
```

```
1
2
3
4
5
```

Exercise 6:
Create a list called “animals" and put "cat","dog","pig"...in it
Use for loop to print each one out

```
5 animals=['cat','dog','pig']
6 for x in animals:
7     print (x)
```

```
cat
dog
pig
```

Additional Resources

- datacamp.com
- software-carpentry.org
- stackoverflow.com

Office Hours - 8PM to 9PM