7001 Python Foundations -

Assignment 1: Basic Python

The following exercises test basic Python that you will encounter in the MS program.

While Python allows any data type to be passed as a parameter (unlike Java and C++) Python 3.5 and above allow you to specify function parameter types. Use this specification format when you write your code.

The function below takes and returns a string and is annotated as follows:

```
def greeting(name: str) -> str:
    return 'Hello ' + name
```

In the function `greeting`, the argument `name` is expected to be of type `str` and the return type `str`.

However, note that Python compilers will **not** report type errors. The type annotations are for specification and documentation.

The types used here are: int, float, bool, List, Tuple, Dict, None, Any (expect any type) , Number (int or float)

If you were asked to take this course as articulation, as part of your admission, you should submit the results of your coding to the Assignments link.

Capture your code and output and submit as a PDF file. Use COURIER font, which is the standard for code examples. Courier font provides uniform spacing is makes it easy to read. Happy coding!

--- --- ---------- ---------- ----------

Coding Assignments

1. Write a Python function which accepts the radius of a circle and returns the area. Display area after function returns.

function: def get_area(radius : float)-> float:

Input: 1.1
Output: 3.8013271108436504

2. Write a Python function which accepts a list of strings and returns the last string in the list. Display the last string after function returns.

function: def get_last(mylist : list[str]) -> str:

IN: ['foo', 'bar', 'zork']
OUT: 'zork

3. Write a Python function that accepts an integer (n) and computes the value of n+nn+nnn. Display result after function returns.

function: def mult1(n : int) -> int:

IN: 5
OUT: 615

4. Write a Python function to calculate and return the sum of three given numbers. If the values are equal then return three times of their sum.

function: def sum1(n1: int, n2:int, n3: int) -> int:

IN: 2,2,2
OUT: 18

IN: 2,3,4
OUT: 9

5. Write a Python function to find whether a given number is even or odd. Return 1 if odd and 0 if even. Display result after function returns.

function: def even_odd(n1: int) -> int:

IN: 2
OUT: 0

IN: 67
OUT: 1

6. Write a Python function to test whether a passed letter is a vowel or not. Return True or False. Display result after function returns. Python does not have a char datatype so simply pass in a string of length 1.

function: def is_vowel (n1: char) -> bool:

IN: a
OUT: True

IN: z
OUT: False

7. Write a Python function to concatenate all elements in a list into a string and return it. Display result after function returns.

function: def make_strl (mylist : List[Any]) -> str:

IN: ['foo', 'bar']

OUT: 'foobar'

IN: ['foo', 23, 99.1, False]
OUT: 'foo2399.1False'

Note: You will need to convert non-string parameters to strings before concatenating.

8. Write a Python function to print out a **set** containing all the colors from color_list_1 which are **not** present in color_list_2. Display result after function returns.

function: def extract1(color_list_1 : List[str], color_list_2: List[str] -> set:

IN:

color_list_1 = ["White", "Black", "Red"]
color_list_2 = ["Red", "Green"]
OUT: {'Black', 'White'}'

9. Write a Python function that will return true if the two given integer values are equal or the absolute value of their sum or difference is 5.

function: def addem (n1 : int, n2: int]) -> bool:

IN: 4,4
OUT: True

IN: 3,2
OUT: True

IN: 23,27
OUT: True

IN: 29,27
OUT: True

IN: 23,67
OUT: False

10. Write a Python function to solve (x + y) * (x + y). Display result after function returns.

function: def solve1 (x : int, y: int]) -> int:

IN: 4,3 (for x and y)
OUT: 49

11. Write a Python function to solve $(x + y)^{(x + y)}$. Display result after function returns.

function: def solve1(x : int, y: int]) -> int:

IN: 4,3 (for x and y)
OUT: 823543

12. Write a Python function to compute the distance between the two points (x1, y1) and (x2, y2). Each point is provided as a tuple of two ints. Display result after function returns.

function: def solve1(p1: tuple[int], p2: tuple[int]) -> int:

IN: (2,5), (6,8)
OUT: 5

13. Write a Python that takes a list of integers and returns True if the ith integer occurs i times and returns False if not. Display result after function returns.

function: def is_ith (p1: list[int]) -> bool:

Test Cases:

Input: [1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6]
Output: True

Input: [1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6]
Output: False

Input: [1, 2, 2, 3, 3, 3]
Output: True

14. Write a Python function to return the longest string of a list of strings. Display result after function returns.

function: def longest_str (p1: list[str]) -> str:

Test Cases:

Input: ['cat', 'car', 'fear', 'center']
Output: center

Input: ['cat', 'dog', 'shatter', 'donut', 'at', 'todo', '']
Output: shatter

15. Write a Python function to split a given string (s) into a List of strings if there is a space in s, otherwise split on commas if there is a comma, otherwise return a list of the string characters in reverse order. Display result after function returns.

function: def split_str (s: String) -> List[str]:

Test Cases:

Input: 'abc de xyz'
Output: ['abc', 'de', 'xyz']
Input: 'xyz w k'
Output: ['abc', 'de', 'xyz']
Input: 'bogus'
Output: ['s', 'u', 'g', 'o', 'b']

16. Write a Python function to find the largest k numbers from a given list of ints. Display result after function returns.

function: def largest_k (mylist: List[int], knum: int) -> List[int]:

Test Cases:

Input: [1,2,3,4,5,6] , 3
Output: [6,5,4]

Input: [1,2,3,4,5,6] , 0
Output: [ ]

Input: [1,2,3,'z',5,6] , 2
Output: [6,5]

17: Write a Python function that takes a list of integers and returns True if all the integers are different from each other, False otherwise. Display result after function returns.

function: def all_different (p1: list[int]) -> bool:

Input: [1,2,3,4,5,6]
Output: True

Input: [1,2,4,4,5,6]
Output: False

18: Write a Python function to remove the duplicate elements of a given list of integers such that each element appear only once. Return the new list. Display result after function returns.

function: def remove_dups (p1: List[int) -> List[int]:

Input: [1,2,3,4,5,6]
Output: [1,2,3,4,5,6]

Input: [1,2,3,4,4,5,5, 6]
Output: [1,2,3,4,5,6]

19. Write a Python function to convert GPAs to letter grades according to the table below. Return a list of strings. Display result after function returns.

function: def grades_to_letter (p1: list[float]) -> List[str]:

Input: [4.0, 3.5, 3.8]
Output: ['A+', 'A-', 'A']

Input: [5.0, 4.7, 3.4, 3.0, 2.7, 2.4, 2.0, 1.7, 1.4, 0.0]
Output: ['A+', 'A+', 'A-', 'B+', 'B', 'B-', 'C+', 'C', 'C-', 'F']

GPAs	Grades
4.0:	A+
3.7:	A
3.4:	A-
3.0:	B+
2.7:	B
2.4:	B-
2.0:	C+
1.7:	C
1.4:	C-
below 1.4:	F

20. Write a Python function to capture the vowels from each string in a list of strings and return a list with only the vowels from each string. Display result after function returns.
Note: y counts as a vowel only when at the end of the word.

function: def grades_to_letter (p1: list[str]) -> List[str]:

Input: ['w3resource', 'Python', 'Java', 'C++']
Output: ['eoue', 'o', 'aa', '']

Input: ['ably', 'abruptly', 'abecedary', 'apparently', 'acknowledgedly']
Output: ['ay', 'auy', 'aeeay', 'aaey', 'aoeey']