|  |
| --- |
| Worksheet IX  Data Structures (more on List, Tuple and Set) in Python |

Find the purpose of the following built-in functions (for List in Python)

|  |  |
| --- | --- |
| list**.append**(x) |  |
| list.**insert**(i, x) |  |
| list.**remove**(x) |  |
| list.**pop**([i]) |  |
| list.**clear**() |  |
| list.**count**(x) |  |
| list.**index**(x[, start[, end]]) |  |
| list.**copy**() |  |

Create a list, namely fruits, with the following items in the list.

fruits = ['tomato', 'apple', 'mango', 'banana', 'kiwi', 'apple', 'banana', 'coconut', 'coconut']

1) Try the followings:

1.fruits.append('durian') and then print(fruits)

2. fruits.insert(0, 'mangosteen' ) and then print(fruits)

3. fruits.remove('kiwi') and then print(fruits)

4. fruits.pop( ) and then print(fruits)

5. fruits.pop(0) and then print(fruits)

5. fruits.pop(3) and then print(fruits)

7. fruits.clear( ) and then print(fruits)

fruits = ['tomato', 'apple', 'mango', 'banana', 'kiwi', 'apple', 'banana', 'coconut', 'coconut']

2) Try

1. print(fruits.count('coconut'))

2. x = fruits.count('coconut') and then print(x)

3. y = fruits.count('pamelo') and then print(y)

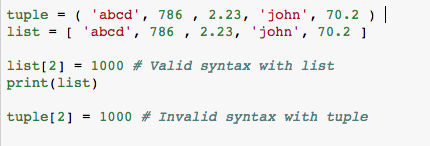
4. myfruits = fruits.copy() and then print(myfruits)

Understanding **List and Tuple**

A tuple is another sequence data type that is similar to the list. A tuple consists of a number of values separated by commas. Unlike lists, however, tuples are enclosed within parenthesis.

The main difference between lists and tuples is- Lists are enclosed in brackets ( [ ] ) and their elements and size can be changed, while tuples are enclosed in parentheses ( ( ) ) and cannot be updated. **Tuples can be thought of as read-only lists**.

3) For example- (try)



tuple[2] = 1000 < ---- you should get error message for this statement as you cannot modify a value in tuple.

**Sets** in Python: a set is an unordered collection **with no duplicate elements**. Basic uses include membership testing and eliminating duplicate entries.

4) To create a set in Python, there are two ways.

car = {'toyota', 'honda', 'honda', 'benz', 'nissan', 'toyota', 'bmw'}

car2 = set(['toyota', 'honda', 'honda', 'benz', 'nissan', 'toyota', 'bmw'])

print(car)

print(car2)

Try (what does these two statements return?)

1. ‘subaru’ in car

2. ‘toyota’ in car2

5) With the following List, write a Python code to print brands of car (with no duplication) in ascending and descending orders.

car = ['toyota', 'honda', 'honda', 'benz', 'nissan', 'toyota', 'bmw']

6) Write a Python function to calculate your saving when you retire, namely myretirement(). The function takes 3 inputs; your current age, your retirement age and your saving per month. The function should print your saving per year starting from a current year to a retirement year. Then write a Python program to take current age, your retirement age and saving per month from a user. Then call myretirement().

7) Write a Python function, namely tempconversion(), that takes temperature and its notation (C or F) and then convert temperature from Celsius to Fahrenheit (and vice version) depending on the notation entered. The tempconversion() should return the converted temperature. Then write a Python program to take temperature with notation (e.g., 101F or 45C) and call tempconversion() and then print the output.

8) Write a Python program to take an integer from a user and print out one of the following four cases:

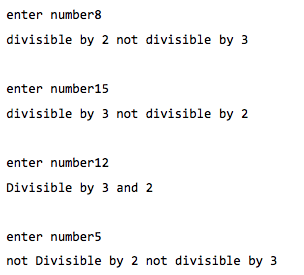
- Divisible by 2 not divisible by 3

- Divisible by 3 not divisible by 2

- Divisible by 3 and 2

- Not divisible by 3 and not divisible by 2

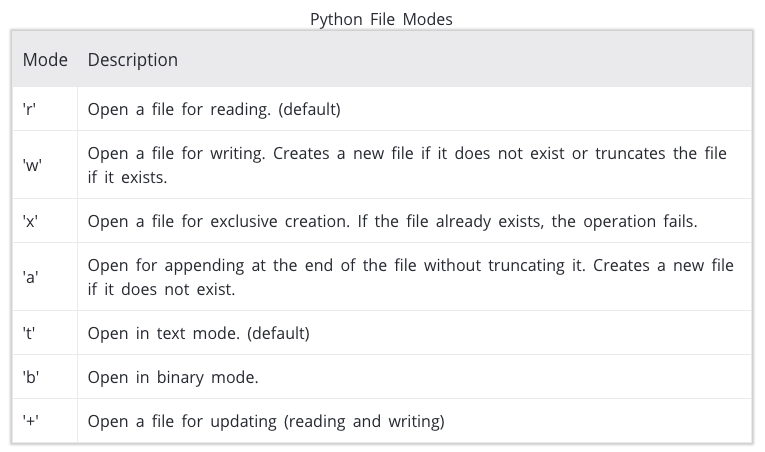
As examples:



9) Write a Python program to find the area of a triangle reading all three sides from a user. Note: use Heron’s formula. If you do not know what Heron’s formula is, search Google.

10) Write a Python function to find the binary equivalent of a number recursively (using recursive function). Then take an integer from a user and pass it to the function, then print the binary number.

**Introduction to file operations in Python:**



11) Try

fh = open("hello.txt", "w") # w option is to write the data to the file

# fh is just a variable name. You can change it to myfile or others

# If you decide to change it to, for example, myfile = open("hello.txt", "w")

# the following statement should be changed to myfile.write("Hello World")

fh.write("Hello World # 1")

fh.close()

* Look for hello.txt in your Jupyter notebook’s home directory and double click to open the file the see what is inside.

Continued from the previous 3 statements, try

fh = open("hello.txt", "a") # a is to append data to the file

fh.write("Hello World # 2")

fh.close()

* Look for hello.txt in your Jupyter notebook’s home directory and double click to open the file the see what is inside.

Continued from the previous 6 statements, try

# Method 1: to read data from a file

fh = open("hello.txt", "r") # r is to read data to the file

x = fh.readline()

y = fh.readline()

print(x)

print(y)

fh.close()

Also try,

# Method 2: to read data from a file

fh = open("hello.txt", "r")

for line in fh

print(line, end = ‘ ’)

fh.close()

**Between the above two methods for reading data from a file, which one is more efficient and fast if you have to read**