**FIRST CLASS: 12th November 2017**

Introduction of Python and walkthrough with commonly used features.

**SECOND CLASS: 18th November 2017**

**Variables and operator in python:**

1. types and variable
2. rules for defining a variable
3. using type()
4. complex numbers using complex()
5. Operators
   1. Arithmetic +,-,\*,/,%,^
   2. Comparison or Relational ==,!=,<=,>=,<,>,is, is not, in
   3. Logical or,and
   4. Bitwise |/or,&/and,^/xor,~/2’s complement
6. Math.ceil/floor
7. sys.argv[]?
8. raw\_input() V/s input()
9. if elif statement

**Exercise**:

1. Input the number from keyboard and check if number is positive or negative.

Modify the code and print “Number is Zero” if number is 0.

2. Decide a number and take guesses from the user and check if it is the number that we have.

3. Python Program to check if number is even or odd.

**THIRD CLASS: 19th November 2017**

**1. Strings intro and exercise**

1. Definition
2. Using + and \* for strings
3. Indexing
4. Split using str.split(“<separator>”)
5. Join using str.join(<sequence>)
6. Using partition
7. Capitalize()
8. Lower()
9. Upper()
10. Count()
11. Max()
12. Min()
13. Title()

**Homework:**

1. Write a program to check the data type and based on type print the number of items present

in the data and get the last element of the data

2. Write a program to store data of 5 students with following details Name, age and class) in the form of list and access their details by checking membership and print their Name, age and class details

**2. List intro and exercise**

* 1. Definition
  2. Indexing
  3. Slicing
  4. Repetition
  5. Concatenation
  6. membership
  7. Updating Lists using indexing

**FOURTH CLASS: 25th November 2017**

**1. Continue List**

* 1. append/insert/remove/del/pop/min/max
  2. List Comprehension **[ expression for item in list** *if conditional* **]**

**Note1:**

python is a dynamic language, it's not usually a good idea to give a variable and a function the same name.

**Example:**

id() is a function in python, so it's recommend not to use a variable named id. Bearing that in mind, that applies to all functions that you might use... a variable shouldn't have the same name as a function.

'id' is a built-in method in Python. ***Assigning a value to 'id' will overwrite the method.*** It is best to use either an identifier before as in "some\_id" or use it in a different capitalization method.

**Note2:**

The difference is that raw\_input() does not exist in Python 3.x, while input() does. Actually, the old raw\_input() has been renamed to input(), and the old input() is gone, but can easily be simulated by using eval(input()).

**Note3:**

Finding max from a list that contains mix values of int and str. Here is the solution:

print(max([i for i in myList if isinstance(i,int)]))

print(max([i for i in myList if isinstance(i,str)]))

separate the by checking type using “isinstance” of list elements and find the max accordingly.

**Exercise**:

1. Write a program to check the data type and based on type print the number of items present in the

data and get the last element of the data

2. Write a program to store data of 5 students with following details Name, age and class)

3. Write a program to find all occurrences of number “3” (using list comprehension) in the below

mention list and later find out the maximum index value of that list output

tmp = [1,2,**3**,5,6,7,**3**,8,4,9,5,0,4,5,2,**3**,7]

Ans: [2,6,15]

4. Multiply Each item of list1 with list2 elements (using list comprehension)

List1 = [1,2,3,4,5,6,7,8,9]

List2 = reverse of list1

- Store the result in new list.

- Filter only Even numbers

- Filter only Odd numbers

- Optimize above program to produce only even number and remove duplicates if any

**FIFTH CLASS: 26th November 2017**

**2. Tuple intro and exercise**

* 1. Definition
  2. Indexing
  3. Slicing
  4. Repetition
  5. Concatenation
  6. List to tuple and vice versa
  7. **append/deletion/pop**/min/max

**3. Dictionary intro and exercise**

* 1. Definition
  2. Keys()
  3. values()
  4. cmp()
  5. Clear()
  6. Del()
  7. copy()
  8. has\_key(key)
  9. update()
  10. pop() and popitem()
  11. Dict Comprehension **[ expression for item in list** *if conditional* **]**

**Exercise**:

1. Write a program to create a dictionary which holds students name as key and their values contain

another dict which contains their age and class.

Ask user to input the name of student and print their details by checking existence of the student

Name.

2. Write a program to create a dictionary (using dictionary comprehension) which holds key as index

and value as value of an element for a given list

tmp = [1,2,**3**,5,6,7,**3**,8,4,9,5,0,4,5,2,**3**,7]

3. write a program to create a dictionary that contains (i, i\*i) such that it is an integral number

between 1 and n (both included). and then the program should print the dictionary.

**SIXTH CLASS: 2nd December 2017**

1. Making Decisions - if Statement
   1. If
   2. If..else
   3. Nested if..else
   4. Multiple Condition using logical operators
2. Loop Control
   1. While
   2. For
   3. Nested loops
   4. All loops with “else” statement
   5. Break , continue and PASS statement in loops
3. Iterator
   1. List iterator
   2. Tuple iterator
   3. Dictionary iterator

**Class Assignment**:

Multiply Each item of list1 with list2 elements.

1. using for loop

2. using iter()

List1 = [1,2,3,4,5,6,7,8,9,]

List2 = [11,12,13,14,15,16,17,18,19]

* Store the result in new list.
* While performing this operation, filter even and odd numbers from result and store them in a separate list
* Make sure to remove duplicates from the resultant list
* Use iter() to concatenate two strings
* Use iter() to create dictionary using two lists. One list holds keys and other list holds values

**SEVENTH CLASS: 3rd December 2017**

1. **Functions & Scope of variables**
   1. Definition
   2. Syntax with example
   3. Calling function with reference and value
   4. Return statement
   5. Arguments:
      1. Required arguments
      2. Keyword arguments
      3. Default arguments
      4. Variable-length arguments
   6. The Anonymous lambda Function
   7. Use of filter,map and reduce with lambda function
   8. Scope of Variables local and global

**Exercise:**

* + - 1. Write a Python function that accepts a string and calculate the number of upper case letters and lower case letters. Don’t use inbuilt functions.

**Homework:**

Write a Python program that accepts a hyphen-separated sequence of words as input and prints the words in a hyphen-separated sequence after sorting them alphabetically.

Sample Items : green-red-yellow-black-white

Expected Result : black-green-red-white-yellow

Use reduce function and using “+” operator, add only 1st element of passed number. Check the result?

E.g:

tmp = [10,2,3]

reduce(lambda x,y:x+1,tmp) **==??**

also

reduce(lambda x,y:x,tmp) **==??**

Using reduce function, retrieve only the 1st and last element of the given list

**EIGHTH CLASS: 9th December 2017**

**Modules**

1. What is module
2. Import statement
   1. Definition and its syntax
   2. Import single and multiple functions
   3. Import functions with aliasing
   4. From …Import statement
   5. From …Import \* statement
   6. Dir() function
   7. globals() and locals() functions
   8. reload() function

**Exercise:**

1. Write a module (with one line of documentation for each defined function) to perform calculator operation for add, sub, mult and div
2. Call above written module using “**import”** in different file and load various available modules and perform the same operations (i.e. add, sub, mult and div)
3. Now create a separate module file for each written function and load them as single package in your test script.

**NINTH CLASS: 10th December 2017**

**File I/O**

1. What is File
2. How to open File for following operations
   1. Read
   2. Write
   3. Append
   4. Close
   5. File positions:
      1. tell()
      2. seek()
   6. next
   7. with statement
   8. file attributes (file.name/closed/mode)
   9. Some “os” related file operations
      1. rename
      2. remove
      3. mkdir
      4. chdir
      5. getcwd

**Exercise:**

1. Write a script which opens a text file and read the each line and print them along with line number.
2. Write a script which opens scripts itself in read only mode and print your program as output. Make sure that newline is not printed since each print statement will by default add one new line
3. Write a program to filter out those lines which is ending with even number in the statement. Also make sure that blank line is not available in final output.

Content of txt file:

This is line1

This is line2

This is line3

This is line4

This is line5

This is line6

This is line7

This is line8

This is line9

This is line10

**TENTH CLASS: 23rd December 2017**

**Logging**

1. What is Logging
2. How to log using “import logging” module
3. Various Levels of logging

**Debugging**

1. What is debugging
2. How to debug your program using “import pdb” module
3. Frequently used and supported commands for pdb

**Exception**

1. What is Exception
2. How to catch exception using “Try … Except”
3. Catching cause of the exception using variable
4. Try … Except….Finally…. clause
5. Raise an exception using “raise …”
6. Assert an exception

**Homework:**

* + 1. Write a program for the above text (experiment:3) and filtered output should be stored in “output.txt” which does not exist. Handle this exception gracefully and store the output.
    2. Debug above program using pdb module by using all the existing keywords

**ELEVENTH CLASS: 24th December 2017**

**Regular Expression**

1. What is Regular expression
2. Match V/s Search
3. Search and Replace
4. Meta and literal characters
5. Regular Expression Modifiers: Option Flags
6. Patterns & descriptions
7. Greedy & non-greedy option
8. BACK REFERENCING

**Exercise:**

1. Write a program to replace all the digits from given text and print the result
2. Write a program to filter out those lines which is
   1. Ending with even number in the statement, print them in output
   2. Print the number of blank lines in final output.
3. Write a script which opens a text file that contains some valid and invalid email IDs.

Read each line of file and print only valid email IDs.

Following is the content of email.txt file:

email1: test123@learnbay.com  
email2: testAbcNY@learnbay.com  
email3: testAb@cNY@learnbay.org  
email4: testAbcNY@learnbay.in123  
email5: testAbcNY@learnbay.in@gmail.com

**Homework:**

1. Continue class exercise to make search pattern more optimized
2. Write above program again which takes input from user to enter email ID and return the message if email ID is valid or not
3. Write a script which takes input to enter IP address and return a message if IP address is valid or not