**DEMO CLASS \_\_\_\_\_\_\_**

1. Introduction
2. Python Intro
3. Interactive Mode Programming
4. Python Identifiers (used to identify a variable , function, class, module or other object
   1. Reserved Words
   2. Lines and Indentation
   3. Multi-Line Statements
   4. Quotation in Python
   5. Comments in Python
5. raw\_input
6. Python Variable Types
   1. Int
   2. Float
   3. String
   4. Multiple assignment a = b = c = 1 Or a,b,c = 1,2,"john"
7. Standard Data Types
   1. Numbers
   2. String
   3. List
   4. Tuple *(The main differences between lists and tuples are: 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.)*
   5. Dictionary (AKA: *associative arrays*)
8. Type casting(conversion):
   1. Ord():Converts a single character to its integer value.
   2. Hex():Converts an integer to a hexadecimal string.
   3. Oct():Converts an integer to an octal string.
   4. bin():Converts an integer to an binary equivalent.
   5. int('<str>', base) Converts an string to an equivalent base format
9. Loops
   1. For
   2. While

**FIRST CLASS: 31st March 2018**

1. String intro and exercise
   1. Definition
   2. Indexing
   3. Slicing
   4. Repetition
   5. Concatenation
   6. min/max
2. List intro and exercise
   1. Definition
   2. Indexing
   3. Slicing
   4. Repetition
   5. Concatenation
   6. append/deletion/pop/min/max
3. Tuple intro and exercise
   1. Definition
   2. Indexing
   3. Slicing
   4. Repetition
   5. Concatenation
   6. append/deletion/pop/min/max

**SECOND CLASS: 1st April 2018**

1. Dictionary intro and exercise
   1. Definition
   2. Indexing
   3. Slicing
   4. append/deletion/iterate over Items

**Class Assignment**:

1. Create a list which contains a number from 1 to 10

2. Using slicing filter even and odd number

3. Create a list of 10 numbers using list comprehension

4. from 3rd question, create dictionary using dict comprehension having key as index and element as value

5 Using list comprehension, create a list of even and odd numbers between 1 to 100

**Loops and control**

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.

* 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
* Using iter function print only even and odd numbers of a list between 1 to 100

**All the above can be found here:**

**List & Loop:**

<https://docs.google.com/presentation/d/1x14FVuvcO7SNNW7VP3DbXXTmc6dIvnlMAtA8kQ-aR00/edit#slide=id.g1c1e2302a2_0_95>

<https://docs.google.com/presentation/d/12Pdc-AHCV3caTp2de9Nson4CI7AnxoEFB_VmZBNDrZU/edit#slide=id.g1c3cb20054_0_16>

**Tupple & Dictionary:**

<https://docs.google.com/presentation/d/1d8rCKdH5vqfrBx7vIcOz8kTMHYVuVx8sI8iEVUMGLXM/edit#slide=id.p>

**THIRD CLASS: 07th April 2018**

1. **Functions & Scope of variables**
   1. Definition
   2. Syntax with example
   3. Return statement
   4. Arguments:
      1. Required arguments
      2. Keyword arguments
      3. Default arguments
      4. Variable-length arguments
   5. The Anonymous lambda Function
   6. Scope of Variables local and global

**Class Exercise:**

* + - 1. Write a Python function that accepts a string and calculate and return the number of upper case letters and lower case letters count as dictionary.

**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

2. Write a Python function to find the Max of three numbers

3. Write a Python function to calculate the factorial of a number (a non-negative

integer). The function accepts the number as an argument

4. Write a Python function to detect the number of local variables declared in a function (to be passed as parameter name)

Note: Try to optimize all the above program using **lambda** function if possible

**FORTH CLASS: 08th April 2018**

**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

**Exercise:**

1. Write a module (with one line of documentation) to generate a list of reverse, even and odd numbers of a given list of integers
2. Call above written module using “**import”** in different file and load various available modules and perform the same operations (i.e. generate list of reverse, even and odd numbers)

**Homework:**

Continuation of class exercise:

**File I/O**

1. What is File
2. How to open File for following operations
   1. Read (r,r+,rb,rb+)
   2. Write (w,w+,wb,wb+)
   3. Append (a,a+,ab,ab+)
   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:** Create a text file with random number of spaces between each line

**FIFTH CLASS: 14th April 2018**

**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 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: Create a text file with random number of spaces between each line

2. Write a program for the above experiment:1 and filtered output should be written in in same file which was opened for read only mode. Handle the exception gracefully and write the output.

**SIXTH CLASS: 15th April 2018**

**Continuation of Exception**

1. Try … Except….Finally…. clause
2. Raise an exception using “raise …”
3. Assert an exception

**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 script which opens a text file that contains some text ending with line number 1 to 10. E.g

“This is line1”

“This is line2” …“This is line10”

Filter those lines which end with even number and print them.

**Homework:**

1. Write a Python program to search 3’letter strings in a string.

Sample text : 'The quick brown fox jumps over the lazy dog.'

Searched words : 'fox', 'dog'

1. Continue class exercise to make search pattern more optimized
2. Write a program 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

**SEVENTH CLASS: 21st April 2018**

**Numpy**

• Introduction to Numpy Array

• Creation, Printing Arrays

• Basic Operations- Indexing, Slicing and Iterating

• Shape Manipulation – Changing shape and spliting of array

• Array stacking hstack and vstack

**EIGTH CLASS: 21st April 2018**

**Pandas & Matplotlib**

• Introduction to Pandas

• Importing data into Python

• Pandas Data Frames, Indexing DataFrames, Basic Operations With Data frame,

Renaming Columns, and filtering a data frame.

• Using pandas.where,pandas.group\_by with get\_group, groups, and filter using

lambda

• Matplotlib – Introduction, plot(),Controlling Line Properties,

Working with Multiple Figures, Histograms

**Homework:**

1. Continue grouping different feature in titanic data sets and create various plots

to understand the data properly. Derive the conclusion based on various graphs.