Python Programming Lab

SPRING SEMESTER 2022

**Lab Instructor : Mrs. Lifna C S**

**Email :** [**lifna.cs@ves.ac.in**](mailto:lifna.cs@ves.ac.in) horizontal line

1. **Text Books & References :**
   1. Dr. R. Nageswara Rao, “Core Python Programming”, Dreamtech Press
   2. [Beginning Python: Using Python 2.6 and Python 3.1. James Payne, Wrox Publication](https://drive.google.com/file/d/15BgP4u04T1uldcKBouih_7spjMuZVCtJ/view?usp=sharing)
   3. Anurag Gupta, G. P. Biswas, “Python Programming”, McGraw-Hill
   4. [E. Balagurusamy, “Introduction to computing and problem-solving using python”, McGraw Hill Education](https://drive.google.com/file/d/1iKiQ0cGAJwdeCuoy5ENnAHjTs6NZDnns/view?usp=sharing)
   5. [Learn Python the Hard Way, 3rd Edition, Zed Shaw's Hard Way Series](https://drive.google.com/file/d/1ehkgyRejhsRaEIVTKolFijIouFQaFEde/view?usp=sharing)
   6. [Laura Cassell, Alan Gauld, “Python Projects”, Wrox Publication](https://drive.google.com/file/d/1RNwvnYR8k_W1DLELnwmO6RiyQ7h0cvKa/view?usp=sharing)
2. **Free Python Online Courses**
   1. [Learn Python- Full Course for Beginners](https://www.youtube.com/watch?v=rfscVS0vtbw)
   2. [Learn Python Course](https://www.learnpython.org/)
   3. [DataCamp Python Programming Course](https://www.datacamp.com/tracks/python-programming)
   4. [Google Python Class](https://developers.google.com/edu/python/)
   5. [Official Docs Tutorial](https://docs.python.org/3/tutorial/index.html)
   6. [NumPy Tutorial](https://www.pythonprogramming.in/numpy-tutorial-with-examples-and-solutions.html)
3. **Online Python Editors :** [Tutorial Point](https://www.tutorialspoint.com/execute_python_online.php), [Programiz](https://www.programiz.com/python-programming/online-compiler/), [Onlinegdb](https://www.onlinegdb.com/online_python_compiler)

**Lab Outcome (LO’s) :**

**On successful completion of the course, the learner will be able to:**

| **LO 1** | To understand basic concepts in python. |
| --- | --- |
| **LO 2** | To explore contents of files, directories and text processing with python |
| **LO 3** | To develop a program for data structure using built in functions in python. |
| **LO 4** | To explore the django web framework for developing python-based web applications. |
| **LO 5** | To understand Multithreading concepts using python. |
| **LO6** | To explore the popular Python Packages - NumPy & Pandas and use them. |

| **No.** | **List of Practical Experiments** | **LO** |
| --- | --- | --- |
| 1 | [Exploring basics of python like data types and control statements.](#jom3rakzqlzy) | LO1 |
| 2 | [Creating functions, classes and objects using python.](#2402x9if3lk) | LO1 |
| 3 | [Demonstrate exception handling and inheritance.](#p0samna35ehk) | LO1 |
| 4 | [Exploring Files and directories, Modules & Packages and perform Text processing using Regular Expressions](#kix.fn2fnay3tit9) | LO2 |
| 5 | [Exploring Data Structures in Python](#kix.3cqyljywm1gq) | LO3 |
| 6 | [Creating GUI with python containing widgets.](#kix.tv43dbyu8wlr) | LO4 |
| 7 | [Exploring Database Programming in Python](#kix.3xzk08jwodng). | LO4 |
| 8 | [Exploring Network Programming in Python](#kix.fr8op3div9v7) | LO4 |
| 9 | [Exploring Django Web Development Framework in Python](#kix.5cccj52zhjtz). | LO4 |
| 10 | [Programs on Threading using Python.](#kix.5dyhux6gj30t) | LO5 |
| 11 | [Exploring basics of NumPy Methods & demonstrating use of NumPy: Array objects.](#kix.f81x0vqi06bd) | LO6 |
| 12 | [Program to demonstrate Data Series and Data Frames using Pandas.](#kix.i9k8c44d2ih7) | LO6 |
| 13 | [Program to send email and read content of URL](#kix.to0dfayce78r). | LO6 |
| 14 | [Mini Project](#kix.t22z0b7b1ufy) | LO1,LO2,LO3.LO4,LO5,LO6 |

**General Instructions**: For all experiments,

1. **Write Aim, Theory & Conclusion by hand** and upload it as the first pdf.
2. **Link your Google Colab Notebook** as the second attachment.

**Lab - 0 : Experiment No : 14 Dated : 13th Jan 2022**

***Title :***  Project Title

***Description :*** Write 2-3 Lines to describe the objective addressed in the Project title

***Program Flow :***

* Explain the working of the Project (at most one paragraph) using a block diagram. And If the project has multiple files, draw a diagram to show the connectivity between them.
* Enlist the packages used and their purpose

***Program :*** Attach the code with proper file names

***Output :*** Screenshot of the working model with proper labels

***Conclusion :***

* Findings
* Applications of the Project
* Future Scope (if applicable

==========================================================================

**Lab - 1 : Experiment No : 1 Dated : 13th Jan 2022**

***Aim :*** Exploring basics of Python - Data types and Control Statements.

***Theory :*** (Note write the Theory by hand)

* **Basic Data Types in Python** (enlist 5 functions for each)
  + Numbers, Boolean, Strings
* **Built-in Containers in Python** (enlist 5 functions for each)
  + Lists, Dictionaries, Sets, Tuples.
* **Looping & Control Statements in Python** (enlist them)
  + Looping statements - While, For, nested loops
  + Control statements - continue, break & pass
* **Arrays in Python** (enlist 5 functions)

***Programs to be performed :***

1. a. To swap two numbers

b. check if the first number is positive or negative or zero.

2. a. To check whether the entered string is palindrome

b. Find the factorial of the input number.

3. Perform the following operations using Lists

* Separate even and odd nos from the list
* merge and sort the two list
* update the first element with x value and delete the middle element of the list.
* find the minimum and maximum element from the list.
* add n names in to the existing list and check if the word "python" is present in the list

4. Perform the following operations using Tuples

* Create a Tuple to store Student details (rno, name , subjects marks, total)
* Display the details
* Sort the tuples wrt to total

5. Perform the following operations using Sets

* Accept two strings using variable declarations.
* Display the common letters.
* Display letters present only in the first string
* Display all letters of both string
* Display letters which are not common in both strings. (Symmetric Difference)

6. Create a dictionary to perform the following operations.

* Update, concatenate, delete
* search a key
* mapping two list into dictionary

7. To search a given element in the Array.

==========================================================================

**Lab - 2 : Experiment No : 2 Dated : 28th Jan 2022**

***Aim :*** Exploring Python - Functions, Classes & Objects

***Theory :*** (**Note** write the Theory by hand)

* **Functions**
  + Syntax for defining Function
  + Advantages of function (atleast 5)
  + Types of arguments (enlist them)
  + Difference between Local & Global variables (at least 2)
* **Classes & Objects**
  + Syntax for defining Class
  + Creating an instance of a class.
  + What is the role of the **self** keyword? (explain in 1 line)
  + Types of Constructors (enlist them)

***Programs to be performed :***

**Functions (Attempt any 3)**

1. Write a function to display Fibonacci Numbers using Recursion. Invoke the function to display the first 20 numbers from the sequence.
2. Write a function to display the ASCII value of the character passed to it while invocation.
3. Write a **menu driven program** to implement a Simple Calculator.
4. Write a Python function to check whether a number is perfect or not.
5. Write a Python function that prints out the **first n rows of Pascal's triangle**
6. Write a Python function to check whether a string is a **pangram or not**.

**Hint** : Pangrams are words or sentences containing every letter of the alphabet at least once. For example : "The quick brown fox jumps over the lazy dog".

**Classes (Attempt any 2 out of 3)**

1. Write a Python class named Student with 3 attributes **student\_id, student\_name and student\_class (class variable)**. Create a function to display the entire attribute and their values in Student class. Create two instances student1, student2 and assign given values to the said instances attributes. Print all the attributes of student1, student2 instances with their values
2. Write a Python class named **Circle** constructed by a radius and two methods which will compute the area and the perimeter of a circle. Create an instance and calculate its area & perimeter by invoking the methods.
3. Write a Python class to **find the validity of a string of parentheses**, '(', ')', '{', '}', '[' and ']. These brackets must be close in the correct order, for example "()" and "()[]{}" are valid but "[)", "({[)]" and "{{{" are invalid.

**Lab - 3 : Experiment No : 3 Dated : 28th Jan 2022**

***Aim :*** Exploring Python - Exception Handling & Inheritance

***Theory :*** (**Note** write the Theory by hand)

* **Exception Handling**
  + Difference between Syntax Error & Exception (at least 2)
  + Common types of Built-In Exceptions. (enlist 5)
  + Syntax for Try-except-else-Finally clause
  + Syntax for Raising Exceptions
  + User Defined Exceptions (explain)
* **Inheritance**
  + Significance of Inheritance (at least 3)
  + Types of Inheritance (enlist them with diagrams)

***Programs to be performed :***

**Exceptions (attempt any 1)**

1. Create User Defined Exception Class. Write a new class (says MyExceptionClass ) for custom exceptions and inherit it from an in-build Exception class. Raise ZeroDivisionError Exception. Incorporate scenarios to catch NameError, ValueError and Arithmetic Error using try-except-else clause. Later display that “The program is successfully executed” via the finally clause.
2. Interactive Calculator : User input is assumed to be a formula that consists of a number, an operator (at least + and -), and another number, separated by white space (e.g. 1 + 1). Split user input using [str.split()](https://docs.python.org/3/library/stdtypes.html#str.split), and check whether the resulting list is valid:

* If the input does not consist of 3 elements, raise a FormulaError, which is a custom Exception.
* Try to convert the first and third input to a float (like so: float\_value = float(str\_value)). Catch any ValueError that occurs, and instead raise a FormulaError
* If the second input is not '+' or '-', again raise a FormulaError
* If the input is valid, perform the calculation and print out the result. The user is then prompted to provide new input, and so on, until the user enters quit.

**Inheritance (attempt any 2 out of 3)**

1. Create a Vehicle class with **name,** **max\_speed, capacity** and **mileage** instance attributes. Create a Bus child class that inherits from the Vehicle class. Give the capacity argument of Bus.seating\_capacity() a **default** value of 50. Define a **class** attribute ”**color**” with a default value **white**. i.e, Every Vehicle should be white. The default fare charge of any vehicle is seating capacity \* 100.
2. If Vehicle is Bus instance, we need to add an extra 10% on full fare as a maintenance charge. So total fare for bus instance will become the final amount = total fare + 10% of the total fare. Override the fare() method of a Vehicle class in Bus class.
3. Write a code snippet to determine which class a given Bus object belongs to.

### Create an object named School\_bus and determine if School\_bus is also an instance of the Vehicle class.

### 2. Implement the any of the following scenarios in Python

3. Implement the following Scenario :

class Shoe:

Attributes: self.color, self.brand

class Converse(Shoe): # Inherits from Shoe

Attributes: self.lowOrHighTop, self.tongueColor, self.brand = "Converse"

class CombatBoot(Shoe): # Inherits from Shoe

Attributes: self.militaryBranch, self.DesertOrJungle

class Sandal(Shoe): # Inherits from Shoe

Attributes: self.openOrClosedToe, self.waterproof

**Lab - 4 : Experiment No : 4 Dated : 4th Feb 2022**

***Aim : Exploring Files, Directories, Modules, Packages, Text Processing using regular expressions***

***Theory :*** (**Note** write the Theory by hand)

* **Files & Directories**
  + Enlist the basic File operations
  + Enlist the basic operations on Directories (at least 4)
* **Modules & Packages**
  + Difference between Modules & Packages (at least 2)
* **Text Processing & Regular Expressions**
  + Enlist at least 4 functions in re module

***Programs to be performed :***

**Exploring File Handling (attempt any 2)**

1. Python program to append data to existing file and then display the entire file
2. Python program to count the word frequency in a file.
3. Python program to open a file & convert the text in a file to upper case and save it.
4. Python program to find the most repeated word in a text file

**Exploring Directories (attempt any 2)**

1. Python program to display file available in current directory
2. Write a program to list all files in the given directory
3. Write a program extcount.py to count the number of files for each extension in the given directory. The program should take a directory name as argument and print count and extension for each available file extension.
4. Write a program to print directory-tree. The program should take the path of a directory as an argument and print all the files in it recursively as a tree.

**Modules & Packages (attempt any 1)**

1. Create a package named “arithmetic\_op” and add modules named, “add.py”, “diff.py”, “square.py” and “modulo.py”. Demonstrate the usage of these modules in your program.
2. Write a Python program to generate a random integer between 0 and 6 - excluding 6, random integer between 5 and 10 - excluding 10, random integer between 0 and 10, with a step of 3 and random date between two dates. Use random.randrange()
3. Write a Python program to configure the rounding to round to the floor, ceiling. Use decimal.ROUND\_FLOOR, decimal.ROUND\_CEILING
4. Write a Python program to read and display the content of a given CSV file. Use csv.reader

**Text Processing (attempt any 1)**

1. Python program to print even length words in a string
2. Python program to create a list of words from a given string.
3. Python program that will read a given text through each line and look for sentences. Print each sentence and divide two sentences with "==============".
4. Python program to remove stopwords from a Text.

==========================================================================

**Lab - 5 : Experiment No : 5 Dated : 14th Feb 2022**

***Aim :* Exploring Data Structures in Python**

***Theory :*** (**Note** write the Theory by hand)

* **Stacks, Queues, Deques & Linked List**
  + Difference between Stacks & Queues (enlist at least 3)
  + Difference between Arrays & Linked List (enlist at least 5)
  + Difference between Queue & Deque (enlist at least 3)

***Programs to be performed : (Attempt any 1)***

1. Create a python package called **‘stack\_app**’ where the following modules are implemented : **(at least 2 modules)**  and demonstrate their usage.
   1. in\_post.py (for Infix to Postfix conversion)
   2. post\_eval.py (for Postfix Evaluation)
   3. in\_pre.py (for Infix to Prefix conversion)
   4. pre\_post.py (to convert prefix to postfix)
2. Create a python package called **‘linkedlist\_ops’** where the following modules are implemented: **(at least 2 modules)**  and demonstrate their usage.
   1. del\_dup.py (to remove the duplicate elements in the LL)
   2. sort\_ll.py (to sort the elements in a LL)
   3. merge\_ll.py (to merge the two given LLs after sorting)
   4. add\_ll.py (to sum all the elements in the LL)
3. Write an implementation of the Priority Queue ADT using List. Also keep the list sorted so that removal is a constant time operation.

**Lab - 6 : Experiment No : 6 Dated : 17th Feb 2022**

***Aim :* Exploring GUI Programming in Python**

***Theory :*** (**Note** write the Theory by hand)

* **Tkinter :** Steps for creating a GUI using Tkinter.

***NOTE:* Perform the experiment in any IDE for Python and create a document with handwritten Theory. For Program and output take printout.**

***Programs to be performed : (Attempt any 1)***

1. Design a Login window and check the credentials of the user. If successful, open up a new window. Otherwise, allow the user to re-enter the credentials (at most 3 attempts).
2. Design a Form to enter the credentials of a Student such as Name, Age, Gender & Address. After collecting the data, display the same in a new window.

==========================================================================

**Lab - 7 : Experiment No : 7 Dated : 24th Feb 2022**

***Aim :* Exploring Database Programming in Python**

***Theory :*** (**Note** write the Theory by hand)

* Explain ACID Properties in DBMS
* Sqlite (Enlist 3 features)

***NOTE:* Perform the experiment in any IDE for Python and create a document with handwritten Theory. For Program and output take printout.**

***Programs to be performed : (Attempt any 1)***

1. Create a Student database and demonstrate CRUD operations on it.
2. Create an Employee database and demonstrate CRUD operations on it.

==========================================================================

**Lab - 8 : Experiment No : 8 Dated : 10th March 2022**

***Aim :* Exploring Network Programming in Python**

***Theory :*** (**Note** write the Theory by hand)

* What is a socket ?
* Syntax to create a socket
* Diagram to illustrate the working of TCP Client Server Application.

***NOTE:* Perform the experiment in any IDE for Python and create a document with handwritten Theory. For Program and output take printout.**

***Tasks to be performed :***

1. Execute the sample program in the [document](https://drive.google.com/file/d/1OCNC_NlYtfMubm_i1E742ACs_Lnq2GwT/view) shared in the Google Classroom
2. Create a Chat Program between TCP Client & Server

==========================================================================

**Lab - 9 : Experiment No : 9 Dated : 16th March 2022**

***Aim :* Exploring Django Web Development Framework for Python**

***Theory :*** (**Note** write the Theory by hand)

* Features of Django
* Draw the workflow of the Django Framework

***NOTE:* Perform the experiment in any IDE for Python and create a document with handwritten Theory. For Program and output take printout.**

***Tasks to be performed :***

1. Install the Django Framework (as per the instructions in the [Cookbook](https://drive.google.com/file/d/1c9gwDKg-HzX3skom37ThnsCXWtCXIaSL/view) shared)
2. Try out creating an App with a simple view as per the Cookbook.
3. Create a Django Project with a Login App connected to the sqlite3 database to check the credentials. If successful, show a welcome message with the username.

==========================================================================

**Lab - 10 : Experiment No : 10 Dated : 7th Apr 2022**

***Aim :* Programs on Threading using Python**

***Theory :*** (**Note** write the Theory by hand)

* Draw the Lifecycle of a Thread
* Modules used for threading and enlist 5 functions for each
* Methods used for Synchronization

***NOTE:* Perform the experiment in any IDE for Python and create a document with handwritten Theory. For Program and output take printout.**

***Tasks to be performed :***

1. Write a method to generate ten 6 digit random numbers and store them in a list
2. Write a method to compute the square-root and another to compute the cube-root of a given number.
3. Write a program to create two threads one to invoke each of the above methods. Show the demonstration for MultiThreading with and without Synchronization

==========================================================================

**Lab - 11 : Experiment No : 11 Dated : 20th Apr 2022**

***Aim :* Exploring basics of NumPy Methods & demonstrating use of NumPy: Array objects**.

***Theory :*** (**Note** write the Theory by hand)

* Why is NumPy faster than Lists?
* Write 5 methods to demonstrate NumPy Array Manipulations

***Tasks to be performed : (Attempt any 5)***

## Convert 1-D array with 8 elements into a 2-D array in Python

# Count frequency of unique values in a NumPy array in Python

## Create a 3D NumPy array of Ones in Python

## Calculate the sum of every column in a NumPy array in Python

## Calculating Mean of Numpy Arrays with different lengths

## Selecting specific rows and columns from NumPy array

## Append a NumPy array to an empty array in Python

## Get the transpose of a NumPy array in Python

## Sum last column only in Numpy

## Check if a NumPy array is empty in Python

## Find the length, dimensions, size of a Numpy array in Python

## Find the index of the max value in a NumPy array in Python?

## Sort a NumPy array in descending order

## Generate a random number in NumPy

## Convert a NumPy array into a list in Python

==========================================================================

**Lab - 12 : Experiment No : 12 Dated : 27th Apr 2022**

***Aim :* Program to demonstrate Data Series and Data Frames using Pandas.**

***Theory :*** (**Note** write the Theory by hand)

* List 5 features of Pandas
* Differentiate between Series & DataFrames
* Enlist frequently 5 methods in Pandas packages

***Tasks to be performed : (Attempt any 5)***

1. Given a dictionary, convert it into corresponding dataframe and display it
2. Given a CSV file, read it into a dataframe and display it
3. Given a dataframe, change the index of a dataframe from the default indexes to a particular column
4. Given a dataframe, sort it by multiple columns
5. Given a dataframe, select a particular column and display it
6. Given a dataframe, select rows based on a condition
7. Given is a dataframe showing name, occupation, salary of people. Find the average salary per occupation
8. Given is a dataframe showing Company Names (cname) and corresponding Profits (profit). Convert the values of Profit column such that values in it greater than 0 are set to True and the rest are set to False.
9. Given are 2 dataframes, with one dataframe containing Employee ID (eid), Employee Name (ename) and Stipend (stipend) and the other dataframe containing Employee ID (eid) and designation of the employee (designation). Output the Dataframe containing Employee ID (eid), Employee Name (ename), Stipend (stipend) and Position (position).
10. Given a dataframe, generate the statistical summary of all the numerical features present in it
11. Write a Pandas program to count the occurrence of a specified substring in a DataFrame column.
12. Write a Pandas program to split the following dataframe into groups based on school code. Also check the type of GroupBy object.

***School class name date\_Of\_Birth age height weight address***

***S1 s001 V Alberto Franco 15/05/2002 12 173 35 street1***

***S2 s002 V Gino Mcneill 17/05/2002 12 192 32 street2***

***S3 s003 VI Ryan Parkes 16/02/1999 13 186 33 street3***

***S4 s001 VI Eesha Hinton 25/09/1998 13 167 30 street1***

***S5 s002 V Gino Mcneill 11/05/2002 14 151 31 street2***

***S6 s004 VI David Parkes 15/09/1997 12 159 32 street4***

1. Write a Pandas program to split the above dataframe by school code and get mean, min, and max value of age for each school.
2. Write a Pandas program to split a dataset, group by one column and get mean, min, and max values by group. Using the following dataset find the mean, min, and max values of purchase amount (purch\_amt) group by customer id (customer\_id).

***ord\_no purch\_amt ord\_date customer\_id salesman\_id***

***0 70001 150.50 2012-10-05 3005 5002***

***1 70009 270.65 2012-09-10 3001 5005***

***2 70002 65.26 2012-10-05 3002 5001***

***3 70004 110.50 2012-08-17 3009 5003***

***4 70007 948.50 2012-09-10 3005 5002***

***5 70005 2400.60 2012-07-27 3007 5001***

***6 70008 5760.00 2012-09-10 3002 5001***

***7 70010 1983.43 2012-10-10 3004 5006***

***8 70003 2480.40 2012-10-10 3009 5003***

***9 70012 250.45 2012-06-27 3008 5002***

***10 70011 75.29 2012-08-17 3003 5007***

***11 70013 3045.60 2012-04-25 3002 5001***

1. Write a Pandas program to display the dimensions or shape of the World alcohol consumption dataset. Also extract the column names from the dataset [Click to see world\_alcohol.csv](https://www.w3resource.com/python-exercises/pandas/filter/world_alcohol.php)

==========================================================================

**Lab - 13 : Experiment No : 13 Dated : 27th Apr 2022**

***Aim : Program to send email and read content of URL.***

***Theory :*** (**Note** write the Theory by hand)

* Enlist 5 methods from **smtplib** package
* Enlist 5 methods from **urllib** package

***Tasks to be performed : (Attempt any 5)***

1. Write a Python code to send a email
2. Write a Python code to read the content of URL