CSL405: Open Source Technology Lab

Course Objectives

The course will help the students to get familiar with:

1. Basics of Python programming
2. Decision Making and Functions in Python
3. Object Oriented Programming using Python
4. Files Handling in Python
5. GUI Programming and Web applications in Python
6. Network Programming in Python
7. Basics of Perl Programming, Decision making and File handling in Perl.

Course Outcomes:

1. To understand basic concepts in python and perl.
2. To explore contents of files, directories and text processing with python
3. To develop program for data structure using built in functions in python.
4. To explore django web framework for developing python based web application.
5. To understand file handling and database handling using perl.
6. To explore basics of two way communication between client and server using python and perl

Prerequisites: Knowledge of some programming language like C, Java

Content

Prerequisites: Knowledge of some programming language like C, Java

SN	Module Name	Detailed Content	Ref. Book	No. of hours
1	Python basics	Data types in python, Operators in python, Input and Output, Control statement, Arrays in python, String and Character in python, Functions, List and Tuples, Dictionaries Exception, Introduction to OOP, Classes, Objects, Interfaces, Inheritance	TB1	6
2	Advanced Python	Files in Python, Directories, Building Modules, Packages, Text Processing, Regular expression in python.	TB1,TB 2	2
3	Data Structure in Python	Link List, Stack, Queues, Dequeues	TB1	2

Content

SN	Module Name	Detailed Content	Ref. Book	No. of hours
4	Python Integration Primer	Graphical User interface, Networking in Python, Python database connectivity, Introduction to Django	TB1,TB 2	6
5	Basics of Perl	Perl Overview, Variables, Control Statements, Subroutines, Objects, Packages and Modules	TB3	4
6	Perl advanced	Working with Files, Data manipulation, Database Systems, Networking	TB3	4

Books

Text Books

1. Core Python Programming, Dr. R. Nageswara Rao, Dreamtech Press
2. Beginning Python: Using Python 2.6 and Python 3.1. James Payne, Wrox publication
3. Perl: The Complete Reference. Second Edition. Martin C. Brown, McGraw-Hill
4. Introduction to computing and problem solving using python , E Balagurusamy,McGraw Hill Education

Reference Book

1. Perl Black Book, 2nd Edition: Steven Holzner,Dreamtech Press
2. Learn Python the Hard Way: (3rd Edition) (Zed Shaw's Hard Way Series)
3. Python Projects , Laura Cassell,Alan Gauld,wrox publication

Digital Links

1. "The Python Tutorial", http://docs.python.org/release/3.0.1/tutorial/
2. Beginning Perl,https://www.perl.org/books/beginning-perl/

Term Work:

Students will submit term work in the form of journal that will include:
1. At least 12-14 programs.
2. One mini-project in a group 2-3 student.
3. Two assignments covering whole syllabus.

Term Work (25) = 15 marks (Experiments & Assignments)
+ 10 marks (Mini Project)
+ 05 marks (Attendance)

Practical and oral examination will be based on suggested practical list and entire syllabus.

PYTHON

Python basics

Python basics: Data types in python ,Operators in python, Input and Output, Control statement, Arrays in python, String and Character in python, Functions,List and Tuples, Dictionaries, Exception, Introduction to OOP, Classes ,Objects , Interfaces, Inheritance

A.

How to write ,save, run first python program
Refer (B1/T)

B.

Python Identifiers
Reserved Words
Lines and Indentation
Multi-Line Statements
Comments in Python
Refer (B1/T)

C.

Data types in python

1. Numbers
2. String
3. List
4. Tuple
5. Dictionary
6. Naming Convention in python

(Overview of it)

Refer (B1/T)

D.

Operators in python

Arithmetic ,Assignment,Relational,Logical,Membership,Identity Operator,

Mathematical Function

Refer (B1/T)

E.

Input and Output

Command Line Arguments

Refer (B1/T)

F.

Control statement If , if else, ef elif else,while, for, else suite*,break,continue,pass,assert*,return Refer (B1/T)

G.

Arrays in python Creating array,array methods,Indexing and Slicing Array Refer (B1/T)

H.

String and Character in python Creating Strings,Length of Strings, Indexing in Strings,String Operators,String Formatting,String Methods (capitalize,count,endswith,find,isalnum,isalpha,isdigit,islower,isupper,istitle,join,len,lower,replace,split,strip,upper) String Sorting, String Searching, Working Character Refer (B1/T)

I.

Functions

Defining Function, Calling Function, Returning Results from a function,Returning multiple values from a Function,Functions are First Class Objects,Pass by object Reference, Formal and Actual Arguments, Positional, Keyword, Default,Variable Length Arguments,Local and global variable, Recursive functions,Anonymous Functions or Lambdas,The special variable _name_

Refer (B1/T)

J.

List and Tuples

Creating Lists, Accessing Values in Lists, Updating Lists,Delete List Elements,Indexing, Slicing,Built-in List Functions and Methods(cmp,len,max,min,append,count,extend,index,insert,pop,remove, reverse,sort), List Comprehensions

Creating Tuples, Accessing Tuple Elements, Tuple operations,Functions to process tuples(cmp,len,max,min,tuple),Nested tuples, N Delete Tuple Elements,

Refer (B1/T)

K.

Dictionaries
Creating Dictionaries,Operations on Dictionaries ,Dictionary Methods,Converting List to Dictionaries
Refer (B1/T)

L.

Exception
Exception handling,The Except Block, Types of Exception,User-Defined Exceptions
Refer (B1/B2/T)

M.

Introduction to OOP
Overview of OOP Terminology
Refer (B1/T)

N.

Classes and Objects
Creating a class,The Self Variable, Constructor, Types of variables,Namespaces, Types of Method
Refer (B1/T)

O.

Inheritance Constructor in Inheritance,Overriding super class constructor and methods,The super method,types of inheritance,Polymorphism,Operator Overloading Refer (B1/T)

P.

Abstract Class and Interfaces Abstract Method and Abstract Class, Refer (B1/T)

Advanced Python

Files in Python, Directories, Building Modules, Packages, Text Processing, Regular expression in python.

A.

Files in Python, Directories

Opening and Closing Files,Reading and Writing Text Files,Appending Text to Files, Directory Contents,Obtaining Information about FIles,

Explain OS module

Refer (T/B1/B2)

C.

Building Modules
Creating modules, The import Statement, The from…import Statement,The from…import * Statement
Refer (B2/T)

D.

Packages
Creating Package and Subpackages, use of __init__.py
Refer (B2/T)

E.

Text Processing
Navigate File System using OS module
Refer (B2)

F.

Regular expression in python
Regular expression,Sequence Characters of Regular Expression, Quantifiers in Regular Expression, Special Character in Regular Expression
Refer (B1)

Data Structure in Python

Link List, Stack, Queues, Dequeues

A.

Link List,
Refer (B1)

B.

Stack
Refer (B1)

C.

Queues
Refer (B1)

D.

Dequeues
Refer (B1)

Python Integration Primer

Graphical User interface ,Networking in Python , Python database connectivity, Introduction to Django

A.

Graphical User interface,
Tkinter,The root window,Canvas and Frames, Widgets (Button,Label,Message,text,Scrollbar,CheckButton,RadioButton,Entry,Spinbox,Listbox,Menu)
Refer (B1/T)

B.

Networking in Python
Socket, Knowing IP Address,URL,Reading Source Code of a Web Page,TCP/IP server, TCP/IP Client, Sending a Simple Mail
Refer (B1)

C.

Python database connectivity
Installing MySQL connector, Using MySQL from Python,Retrieving all rows from a table, Inserting Rows in a Table, Deleting row from a table,Updating Rows in a Table,Create table using Python
Refer (B1/T)

D.

Introduction to Django https://tutorial.djangogirls.org/en/ Refer (B2/T)

B1 : Core Python Programming, Dr. R. Nageswara Rao, Dreamtech Press
B2 : Beginning Python: Using Python 2.6 and Python 3.1. James Payne, Wrox publication

Online Resource Links for Python

Module 1: Python basic

1. **Data types in python:** Boolean (True/False), Numeric (int, long, float, complex), Sequences (Strings, Lists, Tuple, Bytearray, Xrange), Sets, Mappings.

- https://www.slideshare.net/sujithkumar9212301/python-datatypes?qid=b07f3dca-9520-4fbd-abbd-8337b7c22b9a&v=&b=&from_search=6

2. **Operators in python**

- https://www.slideshare.net/MSSalaquzzaman/python-basic-operators-80510232?qid=91500189-6960-4bb0-af1c-bea0255c546c&v=&b=&from_search=1

3. **Input and Output**

- https://www.slideshare.net/MartinMcBride5/print-inputpresentation?qid=ae4eb6a3-9bbe-4749-93ad-18725ca31cdc&v=&b=&from_search=5 ,
- https://www.slideshare.net/mukeshnt/python-reading-and-writing-files?qid=ae4eb6a3-9bbe-4749-93ad-18725ca31cdc&v=&b=&from_search=2
- https://www.slideshare.net/rmarieswaran/input-and-output-39282273?qid=ae4eb6a3-9bbe-4749-93ad-18725ca31cdc&v=&b=&from_search=1

4. Control statement :

- https://www.slideshare.net/p3infotech_solutions/python-programming-essentials-m16-control-flow-statements-and-loops?qid=8d4ab28a-b88a-48e3-bd90-565d53b931ab&v=&b=&from_search=7
- https://www.slideshare.net/rmarieswaran/flow-39280725?qid=8d4ab28a-b88a-48e3-bd90-565d53b931ab&v=&b=&from_search=4

5. Arrays in python ((ppt 85 to 93) and (ppt 16-21, 30-31))

- https://www.slideshare.net/tariq_rashid/a-gentle-introduction-to-coding-with-python?qid=8800a974-c329-4a2a-80b3-272f9e6a4019&v=&b=&from_search=59
- https://www.slideshare.net/pycontw/largescale-arrayoriented-computing-with-python?qid=65b121af-663a-4657-9aa7-ff0682906d49&v=&b=&from_search=4

6. String and Character in python: https://www.slideshare.net/rmarieswaran/strings-39337586

7. Functions :https://www.slideshare.net/rmarieswaran/func-39337687

8. List : https://www.slideshare.net/rmarieswaran/lists-39337266

9. Tuples : https://www.slideshare.net/rmarieswaran/tuples, https://www.slideshare.net/ShahjalalHossain/python002?next_slideshow=1 (list, tuple)

10. Dictionaries Exception:

https://www.slideshare.net/SagarKumar58/python-dictionary-40995279?qid=98d841c6-5755-45d1-b7b0-e8b3e1259a94&v=&b=&from_search=12

11. Introduction to OOP: https://www.slideshare.net/sujithkumar9212301/oop-in-pyhton-new , https://www.slideshare.net/sujithkumar9212301/advance-oop-concepts-in-python,

12. Classes:
https://www.slideshare.net/ranelpadon/python-programming-vi-classes-and-objects?qid=5fbc62b2-71c9-4def-9224-97b52ba7395c&v=&b=&from_search=1

13. Objects:
https://www.slideshare.net/ranelpadon/python-programming-vi-classes-and-objects?qid=5fbc62b2-71c9-4def-9224-97b52ba7395c&v=&b=&from_search=1

14. Interfaces: https://www.python-course.eu/python3_abstract_classes.php

15. Inheritance:
https://www.slideshare.net/AleksanderFabijan/introduction-to-oop-in-python-inheritance?qid=d15f55ab-c0f4-43d9-8912-74b0dec777a6&v=&b=&from_search=2

Module 2: Advanced Python

1.Files in Python,Directories

https://www.slideshare.net/drjimanderson/an-introduction-to-python-files-part-1?qid=395dddff-9b3e-4e9b-a0c5-4b8f7f995ace&v=&b=&from_search=5

https://www.slideshare.net/drjimanderson/python-19?qid=395dddff-9b3e-4e9b-a0c5-4b8f7f995ace&v=&b=&from_search=7

2. Building Modules :

https://www.slideshare.net/k_nitin_r/python-modules?qid=70c7a192-1dbd-4977-9d59-eea91d35b789&v=&b=&from_search=8

https://www.slideshare.net/DamianGordon1/python-modules-and-packages?qid=70c7a192-1dbd-4977-9d59-eea91d35b789&v=&b=&from_search=4

3. Packages:

https://www.slideshare.net/danielhepper/using-python-packages-an-overview?qid=517a22b7-0422-4fd0-b0f4-2d3f1e8fda21&v=&b=&from_search=2

4. Text Processing:

https://www.slideshare.net/kisonlee/python-for-text-processing?qid=a9fec575-22a6-44bc-a4bc-698df59417f1&v=&b=&from_search=8

5. Regular expression in python:

https://www.slideshare.net/sujithkumar9212301/regular-expressions-in-python-37231829

Module 3 : Data Structure in Python

1. Link List

https://www.codefellows.org/blog/implementing-a-singly-linked-list-in-python/

http://interactivepython.org/runestone/static/pythonds/BasicDS/ImplementinganUnorderedListLinkedLists.html

https://gist.github.com/ptigas/2820165

https://www.pythoncentral.io/singly-linked-list-insert-node/

https://medium.com/@kojinoshiba/data-structures-in-python-series-1-linked-lists-d9f848537b4d

https://pythonschool.net/data-structures-algorithms/linked-lists/

2. Stack

3. Queues

4. Dequeues

https://www.slideshare.net/Xlol/lecture08-stacks-andqueuesv3?qid=9dd1a4e2-8949-407e-b603-c798bcf24240&v=&b=&from_search=7 (concept of pt 2,3)

Module 4: Python Integration Primer

1. Graphical User interface

- https://www.slideshare.net/ranelpadon/python-programming-xiii-gui-programming?qid=2eb677a4-a896-4aaa-be27-4253ea621e4f&v=&b=&from_search=1

2. Networking in Python

- https://www.slideshare.net/xibotal/download-mastering-python-networking-ebook-85425635?qid=5d534d44-2533-4b6d-b407-9800db64c7f3&v=&b=&from_search=1

3. Python database connectivity

- https://www.slideshare.net/BaabtraMentoringPartner/database-connectivity-in-python?qid=521421aa-787d-4788-bc7d-6affef4cab33&v=&b=&from_search=1

4. Introduction to Django

- https://www.slideshare.net/solutionstreet/introduction-to-python-and-django?qid=e46bf745-b021-4926-99ec-84b567fea8e2&v=&b=&from_search=2
- https://www.slideshare.net/dibau_naum_h/python-django-intro-v01?qid=e46bf745-b021-4926-99ec-84b567fea8e2&v=&b=&from_search=4
- https://www.slideshare.net/balakumarp/django-framework?qid=e46bf745-b021-4926-99ec-84b567fea8e2&v=&b=&from_search=5
- https://www.slideshare.net/mpirnat/web-development-with-python-and-django?qid=e46bf745-b021-4926-99ec-84b567fea8e2&v=&b=&from_search=6

Django

https://tutorial.djangogirls.org/en/ tutorial for Django

https://www.pythonanywhere.com/ to host Django web app

https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Tutorial_local_library_website Sample Library Web App based on Django

Suggested experiments using Python:

1. Exploring basics of python like data types (strings,list,array,dictionaries,set,tuples) and control
2. statements.
3. Creating functions, classes and objects using python. Demonstrate exception handling and inheritance.
4. Exploring Files and directories
   a. Python program to append data to existing file and then display the entire fille
   b. Python program to count number of lines, words and characters in a file.
   c. Python program to display file available in current directory
5. Creating GUI with python containing widgets such as labels, textbox,radio,checkboxes and custom dialog boxes.

6. Menu driven program for data structure using built in function for link list, stack and queues.
7. Program to demonstrate CRUD(**create, read, update and delete**) operations on database (SQLite/ MySQL) using python.
8. Creation of simple socket for basic information exchange between server and client.
9. Creating web application using Django web framework to demonstrate functionality of user login and registration (also validating user detail using regular expression).

Experiment List Python

1. Write a python program to swap two numbers and check if the first number is positive or negative or zero.
2. Write a menu driven python program to check if number and string is palindrome and find the factorial of the input number
3. Write a menu driven program to demonstrate use of list in python
    a. Put Even and Odd elements into Two Different Lists.
    b. Merge and sort the two list.
    c. Update first element with X value and delete the middle element of list.
    d. Find max and min element from the list.
    e. Add N names into the existing number list and check if word python is present in list.
4. Write a menu driven program to demonstrate use of tuples in python
    a. Add and show N student roll number ,name and 3 subject marks in a list of tuples.
    b. Display student roll number and marks whose name is Python
    c. Demonstrate nested tuple and sort nested tuple by name.

Experiment List Python

5. Write a menu driven program to demonstrate use of set in python
    a. Accept two strings from the user
    b. Display Common Letters in Two Input Strings (Set Intersection)
    c. Displays Letters which are in the First String but not in the Second (Set Difference)
    d. Displays set of all Letters of Both the Strings (Set Union)
    e. Displays Letters which are in the Two Strings but not in Both (Symmetric Difference)
6. Write a menu driven program to demonstrate use of dictionary in python
    a. Create key/value pair dictionary
    b. Update/concatenate and delete item of the existing dictionary
    c. Find a key and print its value
    d. Map two list into dictionary
7. Design an employee class using Python for reading and displaying the employee information,.
8. Write a program to demonstrate single and multiple inheritance in python (with method overloading and overriding)

Experiment List Python

9. Exception Handling

   A. Write a program to demonstrate exception handling using try,multiple except and finally.

   B. Write a python program to create user defined exception.

10. Exploring Files and directories

   a. Python Program to read the content of file and write it in another file

   b. Python program to append data to existing file and then display the entire fille

   c. Python program to count number of lines, words and characters in a file.

   d. Python program to display file available in current directory

11.Create a package and module for data structure: stack and queues.

12.Creation of simple socket for basic information exchange between server and client.

   Reading content of URL

Experiment 1 CODE

```
"""
Write a python program to swap two numbers and check if the first number is positive or negative or zero.
"""

# Take two numbers from user
firstNumber = float(input("Enter first number: "))
secondNumber = int(input("Enter second number: "))

print("Before swapping\n First Number =", firstNumber, " Second Number =", secondNumber)

#Swapping of two numbers
firstNumber, secondNumber = secondNumber, firstNumber

print("After swapping\n First Number =", firstNumber, " Second Number =", secondNumber)

#checking if the first number is postive negative or zero

print("The First Number :",firstNumber,"is : ")

if firstNumber >= 0:
   if firstNumber == 0:
       print("Zero")
   else:
       print("Positive number")
else:
   print("Negative number")
```

Experiment 1 CODE

```
'''
OUTPUT

Enter first number: 10
Enter second number: -20
Before swapping
 First Number = 10.0  Second Number = -20
After swapping
 First Number = -20  Second Number = 10.0
The First Number : -20 is :
Negative number

'''
```

Experiment 6 CODE

```
"""
Write a menu driven program to demonstrate use of dictionary in python
Create key/value pair dictionary
Update/concatenate and delete item of  the existing dictionary
Find a key and print its value
Map two list into dictionary
"""
pyDictionary = {}


def getChoice():
    print("\nMenu\n","(I) Create key/value pair dictionaryr\n",
"(A) Update/concatenate and delete item of  the existing dictionary \n",
"(B) Find a key and print its value\n",
"(C) Map two list into dictionary\n","(Q)uit")
    choose=input(">>> ")
    choice=choose.lower()

    return choice


def dictionaryFirstFunc():

        global pyDictionary
        key=input("Enter the key (int) to be added:")
        value=input("Enter the value for the key to be added:")
        pyDictionary.update({key:value})
        print("Dictionary is:")
        print(pyDictionary)
```

Experiment 6 CODE

```
def dictionarySecondFunc():

        global pyDictionary
        secodDictionary={'C':3,'test':'test'}
        print ("This dictionary",secodDictionary,"will be concatenated to the exixting one")

        pyDictionary.update(secodDictionary)
        print("Concatenated dictionary is:")
        print(pyDictionary)

        key= input("Enter the key to delete():")
        if key in pyDictionary:
                del pyDictionary[key]
                print("Updated dictionary")
                print(pyDictionary)
        else:
                print("Key not found!")



def dictionaryThirdFunc():
        global pyDictionary
        key= input("Enter key to check:")
        if key in pyDictionary.keys():
                print("Key is present and value of the key is:")
                print(pyDictionary[key])
        else:
                print("Key isn't present!")
```

Experiment 6 CODE

```
def dictionaryFourthFunc():

        keys=[]
        values=[]
        n=int(input("Enter number of elements for dictionary:"))

        for x in range(0,n):
                element=input("Enter key " + str(x+1) + " :")
                keys.append(element)
                element=input("Enter value " + str(x+1) + " :")
                values.append(element)


        dictD=dict(zip(keys,values))
        print("The dictionary is:")
        print(dictD)


choice = getChoice()

while choice!="q":
        if choice == "i":
                dictionaryFirstFunc()
        elif choice=="a":
                dictionarySecondFunc()
        elif choice=="b":
                dictionaryThirdFunc()
        elif choice=="c":
                dictionaryFourthFunc()
        else:
                print("Invalid choice, please choose again")
                print("\n")

        choice = getChoice()
```

PERL

Basics of Perl

Perl Overview, Variables, Control Statements, Subroutines, Objects, Packages and Modules

A.

Perl Overview

Installing and Using Perl

Writing a Perl Script

Comments

Operators

Refer (B1/T)

B.

Variables

Literals

Scalar Variable

Array

Hashes

Refer (B1/T)

C.

Control Statements

Conditional Statements (if statement,if...else,if...elsif...else,unless statement,unless...else,unless...elsif..else,switch statement)

Loops (while,utill,for,foreach)

Refer (B1/T)

D.

Subroutines

Define and Call a Subroutine,Passing Arguments to a Subroutine, Passing Lists to Subroutines,Passing Hashes to Subroutines,Private Variables in a Subroutine

Refer (B1/T)

E.

Packages and Modules

Create Package

BEGIN and END block

Creating Modules

The Exporter Module

Require,Use

Refer (B1/B2/T)

F.

Objects and Classes

Object Basics, Defining a Class, Creating and Using Objects,Defining Methods

Refer (B1/B2/T)

Perl advanced

Working with Files, Data manipulation, Database Systems, Networking

A.

Working with Files,

Creating file, Writing, Reading and Appending content to file.

Refer (B1/B2/T)

B.

Data manipulation

Working with Numbers(functions like abc,int,exp,sqrt,log,sin,cos,rand)

Working with Strings(concatenation,length,case modification)

Regular Expression(Pattern Modifiers, The Match Operator,Match Operator Modifiers,The Substitution Operator,Substitution Operator Modifiers)

Refer (B1/T)

C.

Database Systems

Basic Connection and Simple Operations (INSERT,UPDATE,DELETE)

Refer (B1/T)

D.

Networking

Sending mail through Gmail

http://search.cpan.org/~lbrocard/Email-Send-Gmail-0.33/lib/Email/Send/Gmail.pm

Refer (B1/T)

B1 : Perl: The Complete Reference. Second Edition. Martin C. Brown, McGraw-Hill

B2 : Perl Black Book, 2nd Edition: Steven Holzner,Dreamtech Press

Online Resource Links for Perl

Perl Advanced

1. Working with Files

https://www.slideshare.net/kberov/06-io-14473265?qid=804675a1-f319-4800-9be3-0664d1d5240b&v=&b=&from_search=6

2. Data manipulation

https://www.slideshare.net/davorg/lpw-begin?qid=2fecd849-1d69-4bc1-83df-20350acb8769&v=&b=&from_search=17

3. Database Systems

https://www.slideshare.net/DanairatThanabodithammachari/perl-programming-04-programming-database?qid=f3dbae56-a101-483b-a71b-ceae972d507b&v=&b=&from_search=9

https://www.slideshare.net/ldami/working-with-databases-in-perl-1819918?qid=95dfb5ec-d8bc-46b0-a579-99ab07a1aa1d&v=&b=&from_search=2

https://www.tutorialspoint.com/perl/perl_database_access.htm

4. Networking

http://search.cpan.org/~lbrocard/Email-Send-Gmail-0.33/lib/Email/Send/Gmail.pm

Suggested PERL LAB

Suggested experiments using Perl:

10. Exploring various data type , loops and conditional statement in perl. And Creating functions, packages and modules in perl.
11. Program to demonstrate use of objects and classes in perl.
12. Program to demonstrate file handling, data manipulation and use of regular expression for text processing in perl
13. Program to send email and read content of URL.

Experiment List Perl

1. A.Write a perl script find factorial of a number using for and until loop if the number is even number

B. Demonstrate use of array

C. Demonstrate use of hash

2. Write a perl script to create math package and demonstrate use of module

3. Design a class Person and create its object

4. A. Program to demonstrate file handling (Create,read file)

B. Use of regular expression for text processing in perl

5. Write a perl script to send email

Mini Project

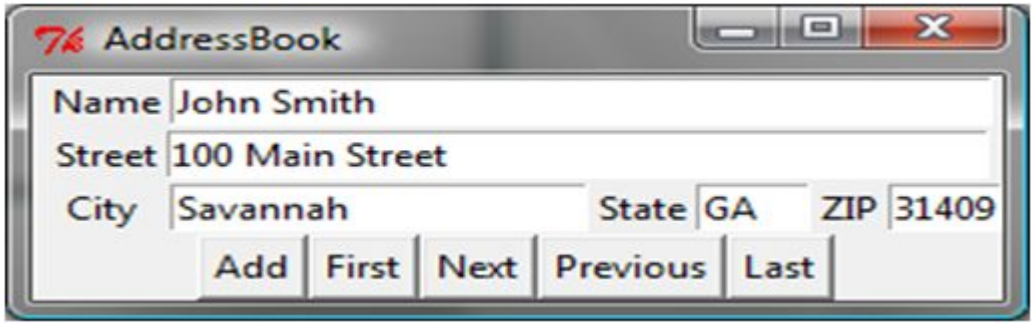
(Mini Project can be web application using Django or Desktop application using Tkinter)

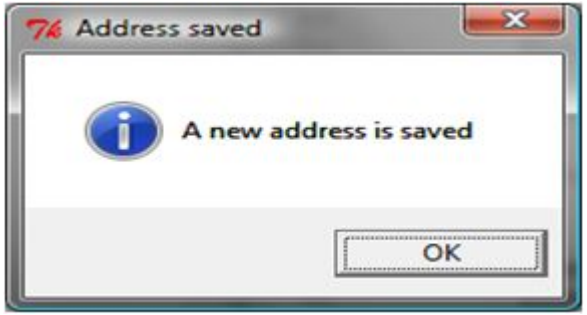
Topics

1. Online/Offline Quiz application
2. Student Feedback Portal
3. Student Portal
4. Teacher Assessment Portal
5. Online chat portal
6. Dice Roll game

Mini Project : Address Book

Use object IO to create a useful project for storing and viewing an address book. The user interface of the program is shown below. The *Add* button stores a new address at the end of the file. The *First*, *Next*, *Previous*, and *Last* buttons retrieve the first, next, previous, and last addresses from the file, respectively.





Implementation of Regular Expression with File

Generate a phone list .There are some people with the surname “thakare”. We are looking for a Thakare, but we don't know the first name, we just know that it starts with a J. Let's write a Python script, which finds all the lines of the phone book, which contain a person with the described surname and a first name starting with J. Develope GUI for the same.

Verifying PANcard

Pancard has standard convention. Like fourth letter will be always “P” if pancard belongs to a person followed by sur-name initial .Develop a GUI based project to verify PANCard .