LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

Unit – 1

Day – 1

Lec #	Topics to be covered:	Assignments:
1 & 2	History & need of Python, Application of Python, Advantages of Python, Disadvantages of Python, Installing Python, Program structure, Interactive Shell, Executable or script files. , User Interface or IDE Python Character Set, Python Tokens, Keywords, Identifiers, Arithmetic Operators	**Essential Assignment:** 1. Write a python program to print "Hello World" 2. Write a python program to get user input using input( ) function 3. WAP to take only numerical input 4. Convert the number to floating point number 5. WAP to Add, Subtract, Multiply and Divide 2 numbers 6. Print the quotient of remainder separately for division operation **Desirable Assignment:** **7.** Write a Python Program to Print Bio Data. **8.** Print your name with Hello.....using input() function **9.** Take an input string and print it three times using one single print statement

Unit – 1

Day – 2

Lec #	Topics to be covered:	Assignments:
3 & 4	Python Character Set, Python Tokens, Keywords, Identifiers, Arithmetic, logical and bitwise Operators	**Essential Assignment:** 1. Write a program to find meter to kilometer. 2. Write a program to find area of a rectangle. (Area=l*b). Take input parameters from user 3. Write a program to find volume of cube. (Area=l*b*h). Take input parameters from user 4. Write a program to find area of triangle. (Area=(l*b)/2). Take input parameters from user 5. WAP to convert the given temperature from Fahrenheit to Celsius using the formula C = (F – 32) / 1.8 6. WAP to convert temperature from Celsius to Fahrenheit where temperature in Celsius is entered by user. (F= C(9/5) + 32) 7. Take a number as user input and convert it into Binary, Octal and Hexadecimal numbers 8. Take binary, octal and hexadecimal numbers as an input and convert them to Decimal number 9. Without applying condition statement display output as “true” if the 1st number greater than the 2nd number and “false” if 2nd number is larger than the 1st one. 10. Take 3 different inputs from user and display their data type. 11. Find the minimum and maximum of 2 numbers 12. Perform binary “AND” and “OR” operation for given 2 integer numbers from user input
	Datatypes in Python int(), float(), bool() Implicit and explicit type conversion	

		Desirable Assignment: 1. Write a program to calculate area of circle. (pi*r*r) 2. Write a program in C to calculate simple interest using formula SI = (P*R*N) / 100. Take all parameters as input from user 3. Without applying condition statement display output as “true” if a number is an even number and “false” if the number is an odd number 4. Find the minimum and maximum of 2 numbers
Unit – 1		
Day – 3		
Lec #	**Topics to be covered:**	**Assignments:**
5 & 6	Sequences in Python: Immutable ordered sequence of elements – strings Slice operator with str datatype, +, * , IN and NOT IN operators with string for s in seq:	**Essential Assignment:** 1. Print multiple lines using single print statement. as – I like “Python Programming” very much It is my favorite subject 2. Print a part of the above string “very much” using the slice operator. 3. Print the last 5 characters from the above given string 4. Print only the second line of the given string 5. Take two strings as input from the user and concatenate them. 6. Take a number and a string from the user and repeat the string for that many times. 7. Take an input character from the user and check whether that character is present in the above given string or not. – Using ‘in’ operator and using ‘not in’ operator
	Practical	
Unit – 1		
Day – 4		
Lec #	**Topics to be covered :**	**Assignments:**
7 & 8	String methods s.count(), s.find(), s.rfind(), s.index(), s.rindex(), s.lower(), s.replace(), s.split()	**Essential Assignment:** 1. Create a menu driven program for string manipulation a. Find the length of a string b. Print the string in upper case c. Print the string in lower case d. Print the string with initial capital e. Split the string **Desirable Assignment:** 2. Take two strings as input s1 and s2 and check whether s2 is present in s1 or not. 3. If s2 is a part of s1 then print the 1st and last occurrences of it 4. If s2 is present in s1 then also count number of times it occurs in s1. 5. Count total number of words in the string input by user
	Practical	

LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

Unit – 1

Day – 5

Lec #	Topics to be covered:	Assignments:
9 & 10	Immutable ordered sequence of elements - Tuple	**Essential Assignment:** 1. Create a tuple for name say t1 (FirstName, MiddleName, LastName) 2. Create a tuple say t2 for marks of 5 subjects 3. Make a total of all the marks and print it. (with and without using sum() method) 4. Make a tuple t3 having 2 elements as t1 and t2 (tuples created above) – It is called a nested tuple 5. Take an input number and find whether that is present as an element in the tuple t3 or not. 6. Create a tuple of 5 fruits. Ask the user to input a fruit name and search that name in the given fruit tuple. Display suitable messages 7. Create a tuple of cities of Gujarat by taking user input. 8. Find the length of name of each city in the above tuple. With and without len() method **Desirable Assignment:** 9. Create a nested tuple t4 of your (name, (hobbies), (friends), degree) 10. Find an element in the nested tuple (t4) and print its position if found, otherwise print "Not found" 11. Take a tuple of 10 integer numbers and segregate odd and even numbers in 2 different tuples.
	Practical	

Unit – 1

Day – 6

Lec #	Topics to be covered:	Assignments:
11 & 12	Range in Python Using range() iteration of tuple and string elements	**Essential Assignment:** 1. Print 1 to 10 numbers in ascending and descending order using range 2. Print odd numbers between 1 to 50 3. Print the '*' patterns using range() 4. Print the number pyramid using range() 5. Take a string input from user and print it in reverse using range 6. Take a tuple input and print all the elements of it in reverse sequence 7. Copy the inputted string to another string by replacing the character 'o' with '@' Eg. 'Hello' will be copied to another string as 'Hell@' and 'Good Morning' will become 'G@@d M@rning' (Without using replace()) 8. Take a string as an input from the user. Find total number of vowels in it. (Hint: take a tuple of vowels) **Desirable Assignment:** 1. Print all prime numbers between 10 to 50 2. Take 10 numbers (min 3 digits) as user input and check whether

		a number is palindrome or not. 3. Find the maximum and minimum numbers from the tuple without using min() and max(). 4. Take a tuple of string elements and print all the strings in reverse. Eg. st=('Pen', 'Marker', 'Pencil','Eraser','Scale') o/p – ('neP', 'rekraM', 'licnep', 'resarE', 'elacS')
Unit – 1		
Day – 7		
Lec #	**Topics to be covered:**	**Assignments:**
13 & 14	Lists and mutability in Python Difference between mutable and immutable objects in python Methods Associated with Lists	**Essential Assignment:** 1. Create a list of students say L1 2. Count total number of students from the list 3. Add one more student in the list L1 4. Display all the students in the sorted order 5. Check a particular student's name is present in the list or not 6. If the student's name is present in the list, print total number of same name students in the list L1 and display the position of 1st student 7. Remove the last student from the list L1 8. Remove a particular student from the list. (Take a name of student from the user.) 9. While removing the student from the list, if multiple students have same name then remove all of them from the list. 10. Create a list of 10 numbers and find the maximum and minimum numbers from it. 11. Create a list of alphabets and count total number of vowels in it. 12. Create a list of even numbers between 1 to 21 using range() 13. Create a list of employees (nested list) with their personal details like [name, age, salary, expertise] in a list. Ask the user to enter name and display the details of that employee. If the employee is not in the list, print error message. **Desirable Assignment:** 1. Create a list by taking any 5 inputs from the user. 2. Display the students from L1 list, whose name contains the character 'a'. 3. Create a list of 10 numbers and find the total of odd numbers and even numbers 4. Put all the odd numbers in 1 list and even numbers in another list. 5. Create a list having mixed elements like string and integers. Print only integer elements from the list with and without using list comprehension
	Practical	

LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

Unit – 1		
Day – 8		
Lec #	**Topics to be covered:**	**Assignments:**
15 & 16	Dictionaries in Python	**Essential Assignment:** 1. Create a dictionary of employees where empId will be the key and value will be the name of an employee 2. Display how many employees are there in the dictionary. 3. Display all empID and add them in a separate list. 4. Display all employee names and take them to a separate list 5. Take an empId from the user and check if that employee is there in the dictionary or not. 6. If an empID is there in the dictionary then display the name of that employee or if not available then add an ID and Name of the employee in the dictionary 7. Change the name of the employee of empID taken by the user 8. Remove an employee whose ID is provided by the user
Unit – 1		
Day – 9		
Lec #	**Topics to be covered:**	**Assignments:**
17 & 18	Revision Practice programs based on list and dictionary object	1. Take 5 names of students as an input from the user and create a dictionary with keys as their initials and value is a list as [age, degree, favorite subject] 2. Display the youngest student from the above dictionary. 3. Create a dictionary of students having rollno of the student is as key and value is a list of marks obtained by that student in 5 subjects 4. Create a dictionary from the above one, where key is rollno and value is (total of all subjects, percentage and grade) a tuple of his result 5. Display the rollno who has scored highest marks (total) 6. Take 10 numbers from the user and create a list, apply bubble sort and arrange the elements in the list
Unit – 1		
Day – 10		
Lec #	**Topics to be covered:**	**Assignments:**
19 & 20	**Class Test – 1** **Unit – 1 Completed**	1. Take an input string from the user and count total number of vowels in it. Print total count of each vowel separately. 2. Take user input for roll no of 5 students and their coma separated marks of 6 subjects (out of 50). 3. Display the minimum and maximum marks of each subject in a separate dictionary object.

LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

Unit – 2

Day – 11

Lec #	Topics to be covered:	Assignments:
21 & 22	Introduction to Functions, Using a Functions Inbuilt functions User defined functions	**Essential Assignment:** 1. Take user input and create a menu driven program to perform mathematical operations like addition, subtraction, multiplication, division, integer division, power. Return values from the functions 2. Create functions to calculate a. Area of a rectangle = width * length b. Area of a triangle = ½ * Height * Base c. Area of a circle = pi*r*r 3. Create functions to convert decimal numbers to binary, octal and hexadecimal numbers. Always return values from the functions 4. Write an UDF to return a list having only unique values by removing duplicate values from the provided input list. Eg. Sample List : [1,2,3,3,3,3,4,5] Unique List : [1, 2, 3, 4, 5] **Desirable Assignment:** 1. Write a Python function to multiply all the numbers in a list. 2. Write a UDF to check the inputted number is between specified range or not. 3. Write a function to calculate total number of Uppercase and lowercase characters in the string. 4. Write an UDF to check if the user given number is a prime number or not.
	Practical	

Unit - 2

Day-12

Lec #	Topics to be covered:	Assignments:
23 & 24	Scope of the variable with functions in python Keyword: global and In-built function globals() to return a dictionary object containing the current global symbols Formal and actual arguments in python 1. Positional arguments 2. Keyword arguments 3. Default arguments 4. Variable length arguments	**Essential Assignment:** 1. Write a findString() function to find all the positions of occurrences of string2 in string1 and return that value. If string2 is not present in string1 then display suitable message. Eg. Str1 = Hello all, Good Morning to all. (pass it as a parameter in the function) Str2 = 'all' (pass it as a parameter, but f not passed take a default argument) o/p: String 2 found at positions: [6, 27] 2. Create a list of fruits and using different functions perform the following operations: Show the use of globals() and don't return from the functions a. Add a fruit at the last b. Insert a fruit at a particular position (pass it as an argument. If the position is not passed then take default argument as 1) c. Update the fruit (use keyword arguments) d. Remove a fruit from the list (pass an index position/ pass a name of the fruit as an argument)

		e. Arrange the fruits in an order **Desirable Assignment:** 1. Create a function to generate prime numbers. Ask total numbers form the user and pass in the function which will return a list of prime numbers. Eg. GeneratePrime(10) function will return 1st 10 prime numbers starting from 2 like 2,3,5,7,11,13,15,1719,23
Unit - 2 **Day-13**		
Lec #	**Topics to be covered:**	**Assignments:**
25 & 26	Using lambda And lambda with filter()	**Essential Assignment:** 1. Create a lambda function that will return maximum of two numbers 2. Create a lambda function that will return maximum of three numbers 3. Write a lambda function that takes one number and if the number is even, returns that number multiplied by 5 else if the number is odd, returns that number multiplied by 10 4. Take a list of mixed elements and a. Write a lambda function to separate integer elements as an output list. b. Write another lambda function to separate string elements as an output list. 5. Modify the above program using filter () 6. Filter all vowels from the given string. **Desirable Assignment:** 1. From the provided list filter, the even numbers and odd numbers as a separate output list 2. Write a lambda function that will 2 inputs. If inputs are integers, it will return the product of 2 numbers. Else perform concatenation.
Unit - 2 **Day-14**		
Lec #	**Topics to be covered:**	**Assignments:**
27 & 28	Using lambda with sorted(), map() and reduce()	**Essential Assignment:** 1. Sort the list elements using lambda 2. Find the average of all the elements passed as an argument in lambda (using variable length arguments) 3. Find the square of each element of a list (using map()) 4. Use a lambda function to calculate grades for a list of scores (using map()) Eg scores = [88, 92, 78, 95, 86] 5. Add all the elements of the list (using reduce()) 6. Multiply all the elements of the list (using reduce()) 7. Find the maximum element from the list using reduce() **Desirable Assignment:** 8. Sorting the dictionary elements using lambda (by using sorted () method) according to age and if age is same then sort my name

		Eg. stud= [{'name': 'Amit', 'age': 25}, {'name': 'Bina', 'age': 22}, {'name': 'Dax', 'age': 25}] NOTE: Solve these exercises as a H.W. https://www.w3resource.com/python-exercises/lambda/index.php
Unit - 2 **Day-15**		
Lec #	**Topics to be covered:**	**Assignments:**
29 & 30	Practice Exercise based on lambda, filter(), map() and reduce()	**Essential Assignment:** 1. Take 2 lists and add the elements of it if the 1st number is greater than the other else find the difference between them Eg. nums1 = [6, 5, 3, 9] nums2 = [0, 1, 7, 7] O/P [6, 4, 10, 2] 2. Take a list of person names and display them all in upper case using map() 3. Take a list of floating-point numbers and display list of all round numbers. Also round them with just 2 decimal points. Using map() Eg. [6.56773, 9.57668, 4.00914, 56.24241, 9.01344] o/p [7, 10, 4, 56, 9] and [6.57, 9.58, 4.01, 56.24, 9.01] 4. Take a list of words and print all palindrome numbers using filter() [Hint: string slicing str1[::-1]]

LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

Unit - 2 Day-16		
Lec #	**Topics to be covered:**	**Assignments:**
31 & 32	Exercise based on lambda()	1. Take a list of students and filter the students whose name is less than 6 characters. 2. Take a string as an input and display the output to analysis the string based on separate words. Using map() a. Display the words in upper case along with the length of each word b. Display total number of each vowel in each word Eg. Str1 = 'Hello how are you?' o/p: [{'a': 0, 'e': 1, 'length': 5}, {'a': 0, 'e': 0, 'length': 3}, {'a': 1, 'e': 1, 'length': 3}, {'a': 0, 'e': 0, 'length': 4}] 3. Take a matrix as input and transpose its elements using lambda Eg. matrix = [[1, 2],[3,4],[5,6],[7,8]] o/p: [[1, 3, 5, 7], [2, 4, 6, 8]]
Unit - 2 Day-17		
Lec #	**Topics to be covered:**	**Assignments:**
33 & 34	Revision Programs based on user defined functions, recursion and lambda()	1. Find the factorial of a number using lambda (recursive) 2. Create a menu driven program with user defined functions to insert update delete elements in the dictionary object of employees Emp = {empCode:[name, age, salary, (expert areas)],.....}
Unit - 2 Day-18		
Lec #	**Topics to be covered:**	**Assignments:**
35 & 36	Class Test – 2 Unit -2 Completed	Write a python script to generate result for a particular student. 1. Create a student data in a dictionary object as shown below: stud = {1: {"name":'Amit', "age":23,"marks": [(10,15,12), (11,12,13)]}, 2: {"name":'Bhumi',"age":22,"marks": [(13,15,11), (10,10,13)]}, 3: {"name":'Bharat',"age":23,"marks": [(12,12,14), (13,14,15)]} } NOTE: Here students are getting marks of 3 subjects in 2 attempts of test in a form of tuple 2. Create separate user defined functions to (Create a menu for options) a. Take user input for creating entry of any new student (addStud()) b. Print all marks of a specific student. (Result(name)) c. Display overall result of all students in a given format

LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

LJ University
University with a Difference

Unit – 3		
Day – 19		
Lec #	**Topics to be covered:**	**Assignments:**
37 & 38	Types of files in Python FILE OPERATIONS Text files - Opening a file, write in a file, append at the last in the file Reading the data from the file	**Essential Assignment:** 1. Create a text file with different modes like w, w+, a, a+ and write few lines in it 2. Read the content of the whole file together 3. Print the length of the file data 4. Read the file content line by line **Desirable Assignment:** 10. Print total number of words in each line in the file 11. Print all the words in reverse.
Unit – 3		
Day – 20		
Lec #	**Topics to be covered:**	**Assignments:**
39 & 40	Working with text files containing strings	**Essential Assignment:** 13. Write multiple lines in a text file. Using list object 14. Take a filename from the user to read that file 15. If the file to be read is not available then print suitable message 16. After reading the file content, append the text at the end of the file. 17. Open a file and append a line at the beginning of the file content 18. Copy the content of one file to another 19. Read 10th to 15th byte from the file and print. **Desirable Assignment:** 5. Read an existing file and take a user input string to be appended in that file. Also ask the position where new line need to be appended. Update the file content and print the updated file. [Hint: Make a file with new line character after each line] 6. Read an alternate bytes/ character from the file. 7. Read alternate lines from the file.
	Practical os.path.isfile() read().splitlines() seek()	
Unit – 3		
Day – 21		
Lec #	**Topics to be covered:**	**Assignments:**

41 & 42	For loop to read line by line The with Statement	**Essential Assignment:** 8. Write a python script to read the text file content and print the output in form of line wise total words in the file. File Content as below: Hello, How are you? Very good morning Have a nice day to all Good Bye… Output: [(1,4), (2,3), (3,6), (4,2)] 9. Open a text file using with statement and write and read the content from that file. 10. Take the user input for data to be written in the text file. Enter the data line by line, till '@' character is entered by the user at the end. 11. Create a text file having string and numeric data. Write a script to separate the string and numbers in two different files [Hint: ch.isdigit() method will return true , if character is a number] 12. Create a menu driven program to perform various file operations through python functions as: a) Create a file – (define the filename as a default argument) b) Read the content of a specified file – Return the content in a string c) Append the content in the specified file d) Rename a file – (Take filename as keyword arguments) e) Delete a file - (define the filename as a default argument) f) Create a directory / folder g) Display all the files present in the specified folder h) Display only .txt file names from the specified folder i) Display the files, starting with letter 't' in their filename. j) Display all python files from a specified folder.(Either .py extension or filename/ folder name contains 'py' in between) k) Display the file names having .txt extension
	File modules in Python os module and its methods os.rename() os.remove() os.mkdir() / os.makedirs() os.listdir() fName.endswith() fName.startswith() fnmatch module and method fnmatch (fname,"reg. exp pattern") * indicates 0 to n number of occurrences ? indicates exactly 1 occurrence . indicates file with extension	

Unit – 3

Day – 22

Lec #	Topics to be covered :	Assignments:

43 & 44	Exception Handling in python
	try:
	Write a code to check
	It's a compulsory block
	except:
	it's an optional block
	else:
	it's an optional block
	finally:
	it's an optional block
	
	ZeroDivisionError
	NameError
	TypeError
		Essential Assignment:
		1. Write a Python script to take a user input to enter a list of elements for employee data and write it into a file. Handle the following exceptions for it.
		a. If an age entered is not a number
		b. If salary is not defined and trying to append in the list
		c. Entered age must be between 18 and 25 only
		d. Salary must be greater than or equal to 10000
		e. Calculate HRA and check for the ZeroDivisionError
		f. Open a file in read mode and try to write into it
		g. Try to write the whole list object into a file
		
		NOTE:

	ValueError AssertionError	• **TypeError**: This exception is raised when an operation or function is applied to an object of the wrong type, such as adding a string to an integer. • **NameError**: This exception is raised when a variable or function name is not found in the current scope. • **IndexError**: This exception is raised when an index is out of range for a list, tuple, or other sequence types. • **KeyError**: This exception is raised when a key is not found in a dictionary. • **ValueError**: This exception is raised when a function or method is called with an invalid argument or input, such as trying to convert a string to an integer when the string does not represent a valid integer. • **AttributeError**: This exception is raised when an attribute or method is not found on an object, such as trying to access a non-existent attribute of a class instance. • **IOError**: This exception is raised when an I/O operation, such as reading or writing a file, fails due to an input/output error. • **ZeroDivisionError**: This exception is raised when an attempt is made to divide a number by zero. • **ImportError**: This exception is raised when an import statement fails to find or load a module.

Unit – 3

Day – 23

Lec #	Topics to be covered:	Assignments:
45 & 46	User defined Exception Pickle Module: pickle.dump(obj, fp) obj = pickle.load(fp)	**Essential Assignment:** 6. Create a dictionary object for student details (Rollno, name, age, hobby, marks…) 7. Create a user defined exception if entered marks are > 50 8. Store all those dictionary data in a binary file. 9. Create separate functions for addData, updateData, deleteData from the binary file

LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

Unit – 3		
Day – 24		
Lec #	**Topics to be covered:**	**Assignments:**
47 & 48	MySQL Database with Python	**Essential Assignment:** 9. Create a Database in MySQL – name MyDB 10. Create a table – employee having fields – eno, ename, age, salary, doj 11. Insert, update and delete specific rows in that table using python script 10. Create a menu driven CRUD operation program to perform above tasks. Help: mysql.connector module connector.connect(host="localhost",user="root",password="", atabase="myDB") db.cursor() cur.execute() row = cur.fetchall() cur.executemany() db.commit()
Unit – 3		
Day – 25		
Lec #	**Topics to be covered:**	**Assignments:**
49 & 50	Text file and MySQL Database with Python Binary file with MySQL Database in Python	**Essential Assignment:** 1. Create a MySQL database table called Tournament. Use exception handling while creating a database table. 2. Insert the records from the readymade text file. 3. Use exception handling while opening and reading the text file. 4. Separate Text files for each sport contains Players details like (name, age, no_of_Tournaments_Played) 5. There should be only one DB table for all sports players, having fields like Name, Age, Sport_Play, Tournaments 6. After inserting data in the table, write a menu driven program to a. Search sport wise no. of players, Tournament wise players list b. Update the tournament field (increment by 1) for a user specified player (when he has played match) c. Take the DB table backup in the binary file. (Use Pickle.dump())
Unit – 3		
Day – 26		
Lec #	**Topics to be covered:**	**Assignments:**
51 & 52	Tkinter module in Python GUI interface	**Essential Assignment:** 1. Create a GUI program that takes user input in the Entry widget and on button click that text must be displayed on the Label widget 2. Insert a record in the DB table by taking user input from the GUI 3. Update and delete the record in the DB table.

LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

Unit – 3		
Day – 27		
Lec #	**Topics to be covered:**	**Assignments:**
53 & 54	Calculator application development	**Essential Assignment:** 1. Create a Calculator application using tkinter module in Python 2. Perform different mathematical operations using GUI – button widget
Unit – 3		
Day – 28		
Lec #	**Topics to be covered:**	**Assignments:**
55 & 56	GUI and File Handling	**Essential Assignment:** 1. Create a GUI for File handling operations like: a. Open a text file – the name provided by the user b. Create a new text file c. Edit the content of the text file d. Delete the specified text file e. Search a specific word in the text file. 2. Create a GUI based program to take a number from the user and separate that number in 2 different files based on if the number is a prime number or not.

LOK JAGRUTI KENDRA UNIVERSITY

Day wise Schedule of MCA, 2st Semester

Course Name: Python Programming (Python) - 140110208

Unit – 4		
Day – 29		
Lec #	**Topics to be covered:**	**Assignments:**
57 & 58	DataFrames in python Operations on Dataframes Queries based on dataframes	**Essential Assignment:** 1. Create a dataframe from an excel sheet. (Student data: RollNo, Name, Age, Marks) 2. Create a dataframe from a .csv file 3. Create a dataframe from the dictionary object 4. Create a dataframe from a list object 5. Display total number of rows and columns in the dataframe 6. Display only 1st 3 rows from dataframe 7. Display only last two rows from dataframe 8. Display 3rd to 7th row of the dsataframe 9. Display all the rows in reverse order. 10. Display all column names of the dataframe. 11. Display only name and age of all students from the dataframe 12. Display maximum and minimum marks from the dataframe. 13. Display the statistical analysis of marks from the student dataframe 14. Display the name of the student having marks > 50 15. Display the rollno and name of the student whose age is > 20 16. Display the students having age between 20 and 25 17. Display the name of the student who has scored maximum marks 18. Display the students who have scored more than average marks (use mean)
Unit – 4		
Day – 30		
Lec #	**Topics to be covered:**	**Assignments:**
59 & 60	Operations on DataFrame	**Essential Assignment:** 8. Change the index in DataFrame and create a new Dataframe 9. Modify the original DataFrame by changing the Index inplace. 10. Search for a particular row using index value 11. Reset the index 12. Arrange all the students in alphabetical order of their names 13. Arrange all the students according to their marks in descending order 14. Display the missing marks with 0 15. Display only those students who have scored more than 0. (drop the missing value row)

		16. Display the DataFrame with suitable message for missing data
Unit – 4		
Day – 31		
Lec #	**Topics to be covered:**	**Assignments:**
61 & 62	Series and Data Frames MultiIndex GroupBy with dataframes	**Essential Assignment:** 1. Create a scalar series (dictionary object with single value) and convert it into dataframe 2. Create MultiIndex.from_arrays like Students [ ], Score [ ], Age [ ] 3. Create MultiIndex.from_frame – a. Use dictionary object for employee data for empId, Name and Salary b. Create DataFrames by read_excel(), read_csv() and set multiindex 4. Create DataFrame from List object with Index values List object contains details for 5 students' result: Pass /Fail with scores in 3 subjects 5. Find total number of 'pass' students and 'fail' students in each subject from above list – use of groupby 6. Find minimum and maximum marks of each subject
Unit – 4		
Day – 32		
Lec #	**Topics to be covered:**	**Assignments:**
63 & 64	Merge, join, concatenate and compare pd.concat(frames) Concatenating Series and DataFrame together Copy Dataframe to excel (df.to_excel ( ))	**Essential Assignment:** 6. Create separate data frames with some common index and some common columns 7. Concat dataframes with outer join (axis=1) 8. Concat dataframes with inner join (join="inner") 9. Concat dataframes with "left" join (.reindex( )) 10. Concat dataframes by Ignoring indexes on the concatenation axis 11. Concat named series with a dataframe 12. Save all the resultant dataframes in the separate sheets of an excel file
Unit – 4		
Day – 33		
Lec #	**Topics to be covered:**	**Assignments:**

65 & 66	Create dataframe from SQL query Use inbuilt sqlite module for creating database Use merge() to merge dataframes on some keys (parameter on=) Merge methods are Left, right, inner (parameter how =) using matplotlib.pyplot and numpy modules plt.plot(), plt.show() methods	**Essential Assignment:** • Create 2 data frames from SQL tables and merge them based on common column (keys) • Plot a line graph, • Make a letter 'A' using plot method with x and y points and then after passing the array of points (multiple lines) • Make letters 'E', 'F', 'H', 'I', 'K', 'L', 'M', 'W', 'X' and 'Z' using plot() • Use of linestyle as 'dotted' , 'dashed', 'dashdot' for plotting above characters, Make all letters colourful, set different line width, use different markers • Make a diamond shape using multiple lnes • Set the graph title, xlabel and ylabel • Draw grids with (x and y axis , linestyle, linewidth) • Save all line graphs in separate .png files Help: pd.read_sql_query (sql, conn) conn = sqlite3.connect('MyDb') c = conn.cursor() c.execute(sql_query)
Unit – 4		
Day – 34		
Lec #	**Topics to be covered:**	**Assignments:**
67 & 68	Matplotlib Bar graphs Matplotlib Histogram Matplotlib Pie Charts	**Essential Assignment:** 11. Draw bargraph for selling of different electronic gadgets 12. Draw a histogram for the yearly performance of a student 13. Draw a pie chart for the student's participation in different games
Unit – 4		
Day – 35		
Lec #	**Topics to be covered:**	**Assignments:**
69 & 70	Revision	**Essential Assignment:** 1. Revision for all incomplete exercise and doubt solving
Unit – 4		
Day – 36		
Lec #	**Topics to be covered:**	**Assignments:**
71 & 72	Unit Test for Unit 3 and 4	**Essential Assignment:** 1. Question Paper