SCRIPTING LANGUAGES LABORATORY

Course Code: ISL58 **Credits: 0:0:2:0**
Course Coordinator: Dr.Mydhili K Nair **Contact Hours: 28P**

PART A

- **Python Introduction** - Simple Problems in Python based on:
  1) Selection Constructs **2)** Looping Constructs
- **Using Python -** Simple Problems in Python based on:
  1) Lists 2**)** Tuples 3**)** Dictionary
- **Combining Python Basics and its Data Structures -** Problems in Python based on:
  Selection, Looping with Lists, Tuples, Dictionaries
- **Using Python -** Problems in Python based on:
  1) Functions **2)** Class
- **Python for Data Science** - Data Manipulation on IRIS Data Set
- **Python for Data Science** -Data Visualization based on IRIS Data Set
- **HTML Basics Introduction** - Creating Static Pages with different HTML Components
- **HTML Basics Introduction** - Creating Static Pages with hyperlinks and images
- **Javascript Introduction** - Basics of client side scripting
- **Using Javascript** - HTML along with Javascript on client side
- **Combining HTML & Javascript** - Programs with Client side event handling scenarios
- **Using Javascript** - Loading JSON data dynamically on the client side.

PART B

1. **Introduction to Python:** Write Python programs to do the following:
   a) Read a list of elements. Create a new list having all the elements minus the duplicates (Use functions). Use one-line comprehensions of create a new list of even numbers. Create another list reversing the elements.
   b) Write a python program to count the frequency of words in a given file.
   c) Read a list of numbers. Uses a recursive function to find the maximum of 'n' numbers.
2. **Introduction to Python Functions:** Write a temperature converter python program, which is menu driven. Each such conversion logic should be defined in separate functions. The program should call the respective function based on the user's requirement. The program should run as long as the user wishes so. Provide an option to view the conversions stored as list of tuples with attributes - from unit value, to unit value sorted by the user's choice (from-value or to-value).
3. **Python Classes:** Write a python class to reverse a sentence (initialized via constructor) word by word. Example: "I am here" should be reversed as "here am I". Create instances of this class for each of the three strings input by the user and display the reversed string for each, in descending order of number of vowels in the string.
4. **Python for Data Science:** Load *Titanic Dataset* into one of the data structures (*NumPy* or *Pandas*). Perform data pre-processing on this dataset. Create dataframes, perform computations and visualize the results appropriately.
5. **Python File Handling & List Comprehension**: Write a python program to read contents of a file (filename as argument) and store number of occurrences of each word in a dictionary. Display the top 10 words with most number of occurrences in descending order. Store the length of each of these words in a list and display the list. Write a one-line reduce function to get the average length and one-line list comprehension to display squares of all odd numbers and display both.
6. **Python for Data Science:** Download the *Weather Dataset* from any online repository.
   a) Change the Column Attributes in Data-frame to read as specified. **E.g**.

Old Attribute	New Attribute	Old Attribute	New Attribute
Mean Sea Level PressureIn	mean_pressure	Max VisibilityMiles	max_visibilty

   b) Use the inbuilt functions to evaluate and analyze the weather data set.
   c) Perform suitable data visualizations.
7. **JavaScript – Client Side Validation:** Design any two case studies that creates and validates a HTML form at the client side using Javascript. Case Study examples could be Grade Calculation from student marks entered, Bakery Menu & Price calculation of items bought etc.
8. **Introduction to JavaScript (JS):** Dynamically loading JSON data - Implement a HTML+JS application that has a JSON Array with details of different kinds of data. Example: Model, Name, Price, Year. Display details of each vehicle dynamically by only showing details of the vehicle that the user has selected (via mouse-over).
9. **Python and JavaScript - Form Validation:** Design any case study that creates and validates a HTML form. On the server side, use Python to display the details entered by the user at the client side along with appropriate validation messages. Use Display appropriate messages if error occurs, and redirect to the another HTML page if successful.

10. **Python and JavaScript - ATM Application:** Design a HTML form that displays user's current balance, an input field to enter amount and buttons to withdraw or deposit money. Validate the form such that
    i) Negative amount cannot be entered and Users cannot withdraw more than 5000 at one time
    ii) Users cannot withdraw amount greater than their balance and cannot deposit more than 10000 at one time. Also users can perform at most 5 transactions. Update the balance accordingly and ensure relevant data is not lost on closing the browser.
11. **Python and JavaScript - Shopping Cart Application:** Design a simple Shopping Cart application which allows users to add items to their cart from a list of products. Allow users to view their cart (items and quantities of each). Ensure that items in the cart persist even after closing the application. On selecting buy, print out a bill of items in the cart. Perform any necessary validation. Demonstrate data persistence even after the browser is closed.
12. **Python for Data Science -** Data Science and Machine Learning on *Boston Housing dataset* - Download Boston Housing dataset from online archives. Load the dataset into one of the data structures (*NumPy* or *Pandas*). Visualize the dataset (Features vs price plot) using *matplotlib*. Apply Linear Regression (Source to learn:*scikit*) on the loaded dataset to predict prices of new data.

Reference Books:

1. Paul Barry, Head First Python, O'Reilly Publication,2010.
2. Shelley Powers, Learning JavaScript, O'Reilly Publication, 2nd Edition, 2012

Course Outcomes:

1. Use internal and external Python libraries, data structures, functions inherent to Python in-order to handle data and use JavaScript to develop command line applications that create child processes to execute other programs and capture their output.(PO-1, 2, 5, 6, 10,12) (PSO-1, 2, 3)
2. Apply Python as a scripting language to analyze huge datasets, apply data science related statistics on datasets (PO-1, 2, 5, 6, 10, 12) (PSO-1, 2, 3)
3. Design and develop a simple web application with client-side JavaScript, server-side Python, using Flask – a micro-framework and develop JavaScript applications that can manipulate HTML pages dynamically, in response to user driven events and do client-side form validations. (PO-1, 2, 5, 6, 10, 12) (PSO-1, 2, 3)

SCRIPTING LANGUAGES LAB

PART A Questions	
Course Code: ISL58	**Credits: 0:0:2:0**
Course Coordinator: Dr.Mydhili K Nair	**Sem 5**

PART A Javascript Questions	
1	Create a HTML form to accept a number. Include button to check whether the input is divisible by 3 or 7. Display the result. Ensure the data entered in the text box is a number only. Use Javascript for this client side scripting. Display appropriate error texts. Also show the execution when the javascript code in a different file.
2	Create the details (name and story title) of four authors as a JSON object. Display a web page displaying the details of ● First two authors in a tabular form using HTML Table Tag ● Other two authors as plain text
3	Create a HTML form which contains one text area. Enter a sentence. Find the length of the longest word of that sentence using javascript. Also show the execution of this code with the javascript code as a ".js" file outside the HTML.
4	Create a HTML form to accept a number. Include two buttons which when clicked a) Button#1 displays value got when the number multiplied by 2 b) Button#2 displays value got when the number is multiplied by itself Display the result. Ensure the data entered in the text box is a number only. Use Javascript for this client side scripting. Display appropriate error texts. Also show the execution when the javascript code in a different file.
5	Create the details (name, native country, story title and publication year) of four authors as a JSON object. Display a web page displaying the details of ● First two authors in a tabular form using HTML Table Tag ● Other two authors as plain text
6	Create a HTML form to simulate a simple calculator. The arithmetic operations are to be displayed as radio buttons. Enter two numbers in two text boxes. Show the result. Handle the error case of 'divide by zero'. Also show the execution of this code with the javascript code as a ".js" file outside the HTML.
PART A Python Questions	
7	**(i)** Create a dictionary that contains the atomic element symbol and its name. **(ii)** Add a unique & duplicate element into this dictionary by interacting with the user. Observe the output and justify it. **(iii)** Display the number of atomic elements in this dictionary **(iv)** Ask user to enter an element to search in the dictionary. Display appropriate results. Rewrite this program so that these operations are inside a function called 'AtomicDictionary'. Create another python file called "Atomic.py" and execute this function in it.

8	Create a list of 6 numbers. Use 'list-comprehension' to create a new list where each element in the original list is multiplied by 3. Use 'lambda' and 'reduce' function find the sum of the elements of the original liat as well as the new list.
9	**(i)** Create a Python class called 'Student' having 'name', 'age' as attribute along with a list having the marks obtained for three subjects. **(ii)** Create a constructor to initialize two objects of this class. **(iii)** Create a member function called 'display' printing the details of a specific object **(iv)** Ask user to enter the values for an object through an 'accept' member function. **(v)** Display these details.
10	Create a dictionary to hold student details with register numbers as the key. Print those student details whose register numbers are even.
11	Write python program to do the following: **(i)** Create a list of strings. **(ii)** Print every string at the even position. **(iii)** Convert every 3 rd string of the list to uppercase, reverse the contents of all the strings in the list and extracts numbers from all the strings in the list.

SCRIPTING LANGUAGES LAB

PART B Questions	
Course Code: ISL58	**Credits: 0:0:2:0**
Course Coordinator: Dr.Mydhili K Nair	**Sem 5**

PART B Python Questions	
1	**Introduction to Python Functions:** Write a temperature converter python program, which is menu driven. Each such conversion logic should be defined in separate functions. The program should call the respective function based on the user's requirement. The program should run as long as the user wishes so. Provide an option to view the conversions stored as list of tuples with attributes - from unit value, to unit value sorted by the user's choice (from-value or to-value).
2	**Python Classes:** Write a python class to reverse a sentence (initialized via constructor) word by word. Example: "I am here" should be reversed as "here am I". Create instances of this class for each of the three strings input by the user and display the reversed string for each, in descending order of number of vowels in the string.
3	**Python for Data Science:** Load Titanic Dataset into one of the data structures (NumPy or Pandas). Perform data pre-processing on this dataset. Create dataframes, perform computations and visualize the results appropriately.
4	**Python File Handling & List Comprehension:** Write a python program to read contents of a file (filename as argument) and store number of occurrences of each word in a dictionary. Display the top 10 words with most number of occurrences in descending order. Store the length of each of these words in a list and display the list. Write a one-line reduce function to get the average length and one-line list comprehension to display squares of all odd numbers and display both.
5	**a)**Load the 'Student Performance' dataset into one of the data structures (NumPy or Pandas). **b)**Display header rows and description of the loaded dataset. **c)** Remove unnecessary features (E.g. drop unwanted columns) from the dataset such as 'lunch' and 'test preparation course' . **d)** Manipulate data by replacing empty column values in 'parental level of education' with a default value. **e)** Convert the attribute 'race/ethnicity' to have 'groupA' to be 'Asian Students', 'groupB' to be 'African Students' , 'groupC' to be 'Afro-Asian Students', 'groupD' to be 'American Students' and 'groupE' to be 'European Students'. **f)** Perform the following visualizations on the loaded dataset: **i)** Tally of the Number of Male & Female students who took up the 'test preparation course' and those who did not. **ii)** Total Number of Male & Female Students belonging to each student group **iii)** No of students who 'failed'(less than 40), 'second class'(between 40 & 50). 'first class'(between 60 & 75) and 'distinction'(above 75) in 'Maths', 'Reading' and 'Writing'.

6	**a)**Load the 'Black Friday' dataset into one of the data structures (NumPy or Pandas). **b)**Display header rows and description of the loaded dataset. **c)** Remove unnecessary features (E.g. drop unwanted columns) from the dataset such as 'User_ID', 'Product_ID ' 'Stay_In_Current_City_Years' . **d)** Manipulate data by replacing empty column values in 'City_Category' with a default value for the city. **e)** Convert the attribute 'City_Category' to have 'A' to be 'Metro Cities', 'B' to be 'Small Towns' , 'C' to be 'Villages'. **f)** Convert the attribute 'Product_Category_1' to have 'Baseball Caps', 'Product_Category_2' to have 'Wine Tumblers' and 'Product_Category_3' to have 'Pet Raincoats' **g)** Convert the attribute 'Marital_Status' to have '1:Married' and '0:Un-Married' **h)** Perform the following visualizations on the loaded dataset: **i)** Tally of the Number of Male & Female persons who bought 'Product_Category_1' and 'Product_Category_2'. **ii)** Total Number of Male & Female persons belonging to each city category
PART B Javascript Questions	
7	**JSON JavaScript (JS):** Dynamically loading JSON data - Implement a HTML+JS application that has a JSON Array with details of different kinds of data. Example: Model, Name, Price, Year. Display details of each vehicle dynamically by only showing details of the vehicle that the user has selected (via mouse-over).
8	**Python and JavaScript - ATM Application:** Design a HTML form that displays user's current balance, an input field to enter amount and buttons to withdraw or deposit money. Validate the form such that negative amount cannot be entered and Users cannot withdraw more than 5000 at one time
9	**Python and JavaScript - Shopping Cart Application**: Design a simple Shopping Cart application which allows users to add items to their cart from a list of products. Allow users to view their cart (items and quantities of each).
10	**Python and JavaScript – Student Registration:** Design a HTML form that displays ● Two text fields to input the user's USN and Date of Birth. ● Three text boxes to input three marks. Validate the data entry on the server side using Javascript so that null values are not accepted for all the five text boxes. Validate the entry on server-side using Python to ensure that USN is accepted in a proper pattern as well as date validations are done. Calculate the average using Python on server-side and display the result.
11	**JSON JavaScript (JS):** Create two JSON objects. One contains the details of a 'Patient' as "name", "AadharNumber" and a JSON array which has the "lab-tests" the patient has taken. The other contains the 'Hospital' details as "hospital-name" and "location". Create a web page that displays the Hospital details when the page loads along with the text "Patient Details:". On mouse-hover the text "Patient Details:" changes colour and displays the details of the patient stored in the JSON object.