# 

# School of Computer Engineering KIIT deemed to be University

**Tools and Techniques Laboratory** **Lesson Plan – 2024 (6thSemester)**

**Program** : B.Tech. (CS, IT, CSSE, CSCE, ECS)

**Academic Session** : Spring 2024

**Semester** : 6th

**Subject Code** : CS-3096

**Subject** : Tools and Techniques Laboratory

**L-T-P** : 0-0-2

**Faculty of Tools and Techniques Laboratory**

**Dr. Soumya Ranjan Nayak (Coordinator)**

**List of Experiments (Day wise):**

**LAB-1**

***Objective: Introduce students to the basics of setting up Python, exploring various Python IDEs and code editors, and running a sample program.***

**Introduction:**

Installing Python & getting familar with different python IDE & code editors (Anaconda, PyCharm, Spyder, Jupyter etc) for running a sample program, Identifiers & Keywords; Variables, Constants & Literals; Comments; Input & Output; Data types; Numbers & Type conversion; Operators & expressions.

**Experiment-1:** Write program to familiarize yourself with fundamental Python concepts such as variables, constants, literals, comments, input/output, data types, numbers, type conversion, operators, and expressions.

**Experiment Overview:**

* Declare variables with meaningful identifiers, assign values, and explore constants and literals.
* Add comments to your code to provide explanations and improve readability.
* Use the input() function to get user input and print() function for output.
* Explore different data types such as int, float, str, and bool.
* Perform operations with numbers and explore type conversion.
* Use operators (+, -, \*, /, //, %, \*\*) to create expressions.

**LAB-2**

***Objective: Gain hands-on experience with conditional statements in Python.***

**Flow Control:** Branching: if..else, Looping: while, for, Others: break, continue & pass statements in flow control.

**Experiment-2:** Guess the Number Game

Create a guessing game where the user has to guess a randomly generated number. Use branching, looping, and flow control statements to manage the game's flow.

**Experiment Overview:**

* Generate a Random Number:
* Use the random module to generate a random number between a specified range. This will be the number the user needs to guess.
* Use a while loop to repeatedly prompt the user for their guess until they correctly guess the number.
* Use if..else statements to check if the user's guess is correct, too high, or too low. Provide appropriate feedback.
* Introduce flow control statements like continue to skip certain parts of the loop or break to exit the loop when the correct guess is made.

**LAB-3,4**

***Objective: Understand the basics of defining and calling user-defined and built-in functions in Python. Apply the knowledge of key OOP concepts, including classes, objects, constructors, operator overloading, and inheritance***

**Function , Oops Concept**

Defining a user defined function, calling a function, parameter passing mechansim to a function, Lambda function in python. Modules in Python, Well known Built-in Functions (or modules) in Python.

**Experiment-3:** Write a program to create a set of user-defined functions to perform basic mathematical operations (addition, subtraction, multiplication, and division) and explore different ways of passing parameters to functions.

**Experiment-4:** Create simple lambda functions for basic operations like addition, subtraction, multiplication, and division.

* Use **lambda functions** with built-in functions like filter() to filter elements from a list.
* Use lambda functions with built-in functions like map() to perform operations on each element of a list.
* Use lambda functions with the sorted() function to customize sorting.

**Experiment-5:** Create a Python module by defining a few functions and variables and Import the module created in step 1 into another Python script and use its functions and variables.

**Experiment-6:** Define a base class, e.g., Vehicle, with attributes such as make, model, and year.

* Implement a constructor to initialize these attributes.
* Create instances of the Vehicle class, representing different vehicles with distinct attributes.
* Display the information for each vehicle using appropriate methods.
* Overload the + operator to combine the make and model attributes of two instances.
* Display the result of this operator overloading operation.
* Create a derived class, e.g., Car, that inherits from the Vehicle class.
* Add specific attributes to the Car class, such as num\_doors and fuel\_type.
* Instantiate objects of both the Vehicle and Car classes.
* Display the information for each object, showcasing the inheritance structure.

**LAB-5**

**Arrays, List, Dictionary, Tuples, Set**

***Objective: Explore array,List, Dictionary, Tuples, Set operations in Python***

**Experiment-7:** Create a list of numbers and perform various operations such as adding elements, removing elements, and accessing elements by index.

**Experiment-8:** Create a tuple to represent an immutable collection and try accessing elements and performing basic operations.

**Experiment-9:** Create a set to store unique elements and perform set operations such as union, intersection, and difference.

**Experiment-10:** Create a dictionary to represent key-value pairs and perform operations like adding, updating, and accessing values.

**LAB-6**

**NumPy:**

***Objective: Gain hands-on experience with NumPy, a powerful numerical computing library.***

**Experiment-11:** Create NumPy arrays with different methods (lists, zeros, ones, arange).

* Perform mathematical operations on NumPy arrays (addition, subtraction, multiplication, division).
* Practice indexing and slicing NumPy arrays.
* Use NumPy functions for basic statistical analysis like mean,median and mode.
* Experiment with reshaping and transposing NumPy arrays

**LAB-7**

**Pandas**

***Objective: Understand and apply the Pandas library for data manipulation and analysis.***

**Experiment-12:** Create a Pandas Data Frame from .CSV files (Student can download any CSV file like iris.csv, Indiacrime.csv etc from kaggle )

* Explore basic DataFrame operations, such as head(), tail(), info(), describe().
* Practice selecting and indexing data from a DataFrame.
* Filter data based on conditions.
* Perform basic data cleaning operations, such as handling missing values.
* Practice grouping and aggregating data.
* Use Pandas for basic data visualization

**LAB-8**

**Matplotlib and Seaborn**

***Objective: Explore Matplotlib, Seaborn for data visualization.***

**Experiment-13:** Perform Data Visualization with Matplotlib and Seaborn.

* Create a simple line plot using Matplotlib.
* Create a scatter plot using Matplotlib.
* Create a bar plot using Seaborn
* Create a histogram using Seaborn.
* Create a box plot using Seaborn.
* Create a heatmap using Seaborn.

**LAB-9**

**SciPy( Optional)**

***Objective: Apply SciPy for scientific and technical computing.***

**Experiment-14:** Perform basic operations such as integration, differentiation, and optimization using SciPy and also solve linear algebra problems using SciPy functions.

**LAB-10**

**Scikit-Learn (sklearn)**

***Objective: Introduce the Scikit-Learn library for machine learning and TensorFlow for deep learning and neural networks.***

**Experiment-15:** Load a built-in dataset from Scikit-Learn to understand its structure. Perform basic data preprocessing steps such as handling missing values and encoding categorical variables. Also use Scikit-Learn to build a simple machine learning model (e.g., a classifier or regressor).

**LAB-11**

**Keras / TensorFlow**

***Objective: Understand and apply Keras / TensorFlow API to train a neural network***

**Experiment-16:** Create a basic neural network using TensorFlow, including defining layers and compiling the model. Train the neural network on a dataset and evaluate its performance. Learn how to save and load trained models for future use.

**LAB-12**

**PyTorch( Optional)**

***Objective: Introduce PyTorch for deep learning and neural network research.***

**Experiment-17:** Define and build a neural network using PyTorch's tensor operations. Train the neural network on a dataset using PyTorch's autograd capabilities.

**LAB-13**

**Introduction to OpenCV( Optional)**

***Objective: Familiarize students with OpenCV, a popular computer vision library in Python, and perform basic image and video processing operations.***

**Experiment-18:** Use OpenCV to load and display images from files. Perform basic image processing operations, such as resizing, rotating, and flipping images.

* Apply various image filters (e.g., Gaussian blur, median blur) using OpenCV to observe their effects.
* Convert color images to grayscale using OpenCV functions.
* Apply edge detection techniques (e.g., Canny edge detector) to highlight edges in images.
* Use OpenCV to detect contours in images and draw them.

**Mini Project Work : Student should submit a mini project group wise (min 4,5 members in a group)**

**Grading Policies:**

**Continuous Evaluation components: Continuous evaluation for 60 marks Consists following components:**

* **Lab participation (10 Marks)**: Students' participation in the lab based on their attendance and engagement.
* **Lab records (10 Marks)**: Neatly written lab records based on the assignments to be evaluated.
* **Continuous evaluation (based on Lab skills, 20 Marks)**: Students' lab skills will be assessed through hands-on activities and involvements in doing assignments during the lab hour. 
* C**omprehensive assessment** (**(20 Marks))** of student learning

**End semester evaluation: Comprehensive assessment of student learning and Performance**

**(40 marks)**.

**List of Mini Projects with Python**

1.Demand & Supply Analysis

2.Recession Analysis

3.Algorithmic Trading

4.A/b Testing

5.T20 World Cup 2023 Analysis

6.Twitter Timeline in the Stock Market Analysis

7.Instagram Reach Analysis

8.Screen Time Analysis

9.Stock Market Analysis

10.Smartwatch Data Analysis

11.Virat Kohli Performance Analysis

12.Google Search Analysis

13.Survey Analysis

14.Social Progress Index Analysis

15.Career Aspirations Survey Analysis

16.iPhone Sales Analysis

17.Diamond Price Analysis

18.Covid-19 Impacts Analysis

19.IPL 2022 Analysis

20.Ukraine Vs Russia: Twitter Sentiment Analysis

21.Waiter Tips Analysis and Prediction

22.Flipkart Reviews Sentiment Analysis

23.Time Series Analysis

24.TikTok Reviews Sentiment Analysis

25.Tinder Reviews Sentiment Analysis

26.Uber Trips Analysis

27.WhatsApp Chats Analysis

28.Water Quality Analysis

29.Squid Game Sentiment Analysis

30.Twitter Sentiment Analysis

31.Movie Rating Analysis for Beginners

32.Worldwide Billionaires Analysis

33.Unemployment Analysis

34.WhatsApp Chats Sentiment Analysis

35.Covid-19 Vaccine Analysis

36.Financial Budget Analysis

37.Best Streaming Service Analysis

38.Data Science Project on Birth Rate Analysis

39.Data Science Project on Time Series

40.Data Science Project on Area and Population

41.Amazon Product Reviews Sentiment Analysis

42.Hotel Reviews Sentiment Analysis

43.Google Play Store Sentiment Analysis

44.Instagram Reach Analysis

45.Amazon Alexa Reviews Sentiment Analysis

46.Real-time Sentiment Analysis

47.Customer Personality Analysis

48.Netflix Data Analysis

49.Covid-19 Vaccine Sentiment Analysis

50.Cohort Analysis