



**A report on**

**Faculty Development Program on  
Python and Advanced Teaching Methodologies**

**In Association with**

**Commonwealth Educational Media Centre for Asia - CEMCA**

**New Delhi**

**&**

**Andhra Pradesh Information Technology Academy-APITA,**

**Vijayawada**

**Venue:**

**Avanthi Institute of Engineering and Technology, Vizianagaram**

**Resource Persons & Report Prepared by:**

**S. Dinakar Reddy**

Corporate Master Trainer & Facilitator  
Mentor in Innovations & Entrepreneurship  
Instructional Designer of LST

**Banoth Prasad**

CEO, CO-Founder  
Developer  
K4U Infotech, Hyderabad

## Table of Contents

<b>S. No.</b>	<b>Content</b>	<b>Page Number</b>
1	Acknowledgment	3
2	Background	4
3	Motivation	5
4	Introduction	6
5	Learning Objectives	7
6	Date and Time	9
7	Participants	9
8	Training Python and ATT	9
9	Course Content and Structure	11
10	Workshop Proceedings	12
11	Closing Ceremony	46
12	List of Participants	49

# 1. ACKNOWLEDGEMENT

We earnestly acknowledge the immense contribution of Dr. Dr. Basheerhmad Sadrach, Director of Commonwealth Educational Media Centre for Asia (CEMCA), and Mr. Saurabh Mishra, Programme Officer (Skills) of Commonwealth Educational Media Centre for Asia (CEMCA), for conceptualizing and floating the highly need-based and demand-driven workshops for upgrading the capacity of the FDP and sensitization program in association with Mr. Sunil Reddy (IFS), CEO of Andhra Pradesh Information Technology Academy (APITA) and Mr. Vishwanath, Director of Andhra Pradesh Information Technology Academy (APITA) for higher education faculties (engineering colleges) in Vizianagaram District, Andhra Pradesh.

The excellent endeavour of CEMCA is expected to have a profound impact on the qualitative enhancement of the Professors, Lecturers, and HODs & Trainers of all the Engineering Colleges of AP with the different methodology of “Python and Advanced Teaching Methodologies” with the Latest & Traditional, Easy for good, not good & mediocre participant’s perspective of both content & methodology.

It was a privilege for us to get the opportunity to contribute significantly to this value-driven and impact-oriented mission of CEMCA.

## 2.BACKGROUND

IT is possibly the fastest developing sector in the last few decades. Computers (in all shapes and sizes) and the Internet have acquired a very important role in most domains of our everyday life. In the same way, education is not imaginable anymore without a strong presence of teaching methodologies. Faculties have had to adapt teaching methodologies throughout curriculum delivery to perform the dual task of providing formal, subject-based education and of encouraging pupils' personal development in a wider social and cultural context.

On one hand, formal education involves understanding concepts and processes. It aims at acquiring knowledge and skills to be able to apply what has been learned within each subject area and across subjects. On the other hand, pupils need to learn to describe, analyze, interpret, have a critical view, and apply their acquired knowledge and skills creatively and originally in a wide range of social, scientific, and cultural contexts.

Since the world of python is changing so rapidly, an important goal is to provide hands-on experience to faculty to equip pupils with the basic knowledge and fundamental skills on one hand, and, on the other hand, to empower them to become independent and autonomous learners too. In this sense, faculty should acquire a range of skills that will allow them to continue learning and teaching with an improved Teaching-Learning Process.

### 3. MOTIVATION

#### **Python:**

Everyone is talking about Python, and for good reason. With an exponentially growing community around data science, machine learning, AI, web dev and more, Python is a language that opens programming access to the world.

In the realm of server side software languages, Python is considered easy to read, write and learn. Plus, it's extremely scalable. The world's largest companies run on it, and it's also great for quick and simple prototypes.

The list goes on for why we think it's a good idea to learn Python programming, whether you're new to coding or an experienced dev.

#### **Advanced Teaching Methods (ATM):**

Innovative teaching methods have improved the learning process and strengthened governance and methods are designed to improve the quality of education along with professors and student's involvement in the educational process. Method to enhance or expand upon the trainee's experience. One of the basic motives behind advanced teaching is to motivate students/trainees to actively take part in the learning process. When the level of interaction with trainees and peers increases, students gain knowledge that is practical and also, retain more information effectively from the classes.

This motivating the trainers/professors/lectures/students will make to understand and redeliver not only a subject or a technology but gives more accurate and more knowledge, data, statistics, case study with the current memory in the existing co related knowledge

## 4. INTRODUCTION

### **PYTHON:**

Python is a high-level programming language used for general-purpose software engineering. It's a server side language – which means it runs on the server, and is responsible for processing the logic behind user inputs, interacting with databases and other servers, etc.

Initially developed in the late 1980's by Guido Van Rossum, Python has been around for decades alongside other server side languages like Java and C. Van Rossum modeled Python after the English language, eliminating unnecessary syntax to make it easier to read and write than other programming languages.

Python is an open-sourced language, and in recent years has increased in popularity due to its use in data science. Python also has a strong community around machine learning, data modeling, data analysis and artificial intelligence (AI), with extensive resources and libraries built for these purposes.

And yes, the rumors are true. Python is named after the British comedy group Monty Python.

### **ADVANCED TEACHING METHODS (ATM):**

Advanced Teaching Methods for any trainer in the session provides a comprehensive, critical approach to meeting the new challenges in the session. This program gathers research on Advanced Teaching methods, principles, and content, and acts as a reference source for proven and innovative methods.

Advanced Teaching Methods for the trainers presents the style to teaching educational technology, design, and engineering. It also contains strategies for innovation by examining the what, why, and how of technology education.

The whole program is a nice balance between foundational and practical issues. It is quite an accomplishment to put together a comprehensive program such as this.

## **5. LEARNING OBJECTIVE**

The course itself is structured around four modules covering the topics ranging from basic concepts around Advanced Teaching Methodologies and Python, all the way to understanding python Solution capabilities and general scope for the introduction.

Upon completion of the training program, the faculty will be able to:

### **Python**

- Introduction to Python – Indentation, Python Variables, Python Data Types – Numbers, String
- Lists, Tuples, Sets and Dictionaries
- Python File Handling, Python String Formatting
- Working with Jupiter Notebook
- Role of Python in Data Science, Data Science Fundamentals, Statistical Tools, Techniques for data science
- Working with data pre-processing and data visualization using Python - Pandas , Numpy, Matplotlib
- Python tkinter - Widgets, Standard attributes, Geometry Management
- Python tkinter – Widgets - Labels, Buttons, Entry Field, Text Field, Check Buttons, List Box, Radio Buttons, Menu

- **Exercise :**
  - Random Password Generator in Python
  - Convert Text to Speech in Python

## **Advanced Teaching Methodology - ATM**

- Explain the principles of ANDRAGOGY viz. Adult Learning
- Enlighten learners regarding the features of the facilitation process
- Explain to learners the features of Facilitation that demarcate it from the Mixed traditional lecture method
- The Etymology of teaching techniques.
- Highlight the importance of Examples, Data, Statistics and History with JAM ( Just a Minute) Facilitation Storytelling (Panchatantra), in online & Offline learning/ Teaching
- Explain different psychosocial interventions with 'Panchatantra' that are deployed in facilitation.
- Make the learners illuminated with the beneficial aspects of different learning models, Remembering techniques, Corollaries important Vocabulary etc.
- Sensitization of learners with the different learning methods that are highly effective in online & Offline / Physical learning
- Administer practical exercises for fostering practice- teaching, with a goal-oriented approach.

## **6.DATE & TIME**



S. No.	Course	Date	Time	Participants	Count
1	Python	30th May to 01 <sup>st</sup> June 2022	09:30 AM to 05:00 PM	Higher Education Faculties from Andhra Pradesh	52
2	Advanced Teaching Methodologies	02 <sup>nd</sup> June to 03 <sup>rd</sup> June 2022			

## 7.PARTICIPANTS

52 faculties from 6 engineering colleges out of which 15 are from the department of CSE, 5 are from the department of CSIT, 28 are from the department of ECE, 1 from the department of EEE, 1 from the department of ME, 1 from the department of CIVIL and 1 from the department of MCA attended the workshop conducted between 23<sup>rd</sup> May to 27<sup>th</sup> May 2022 at Siddartha Institute of Science and Technology, Puttur, Andhra Pradesh.

A Detailed List of Participants for workshops is attached in **Annexure – A**

## 8.TRAINING PYTHON and ATM

The Training was conducted on the college lab premises. The methodology used was a live demonstration of Teaching Methodologies and Python followed by hands -on practice by participants using the assignment questions provided by the resource person. The organizing team shared a WhatsApp group to interact, share information, answer queries, and submit feedback.

S. No.	Event	Time	Participants
1	Demonstration Sessions	09:30 AM to 05:00 PM	Classroom
2	Interaction with Queries and Feedback	05:00 PM to 08:00 PM	WhatsApp

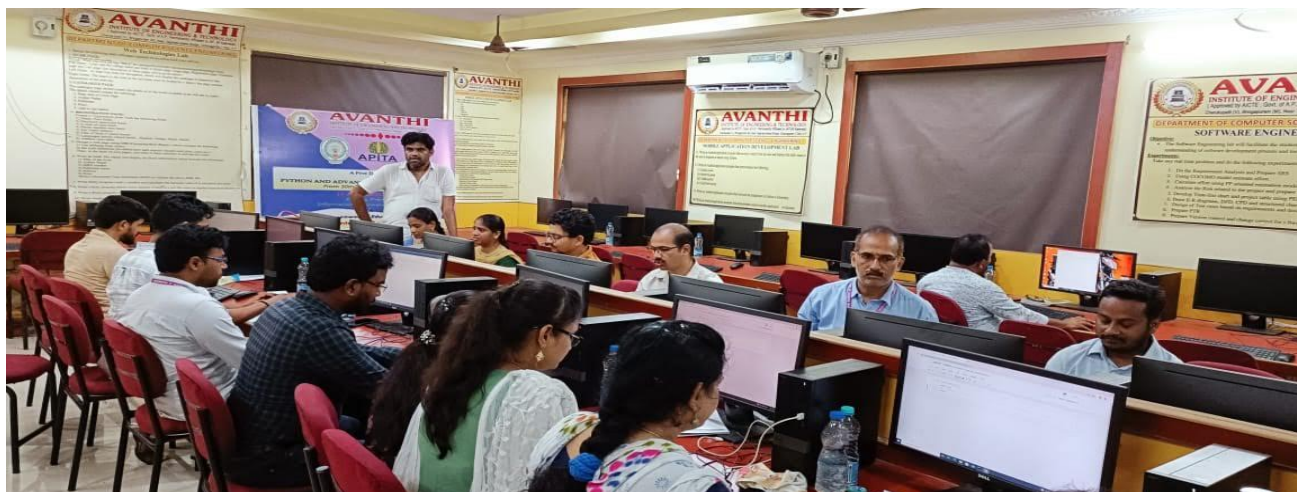


Fig: Live Demonstration on Python by Mr. Banoth Prasad



Fig: Live Demonstration on Advanced Teaching Methodologies by Mr. S. Dinakar Reddy

## 9. COURSE CONTENT & STRUCTURE

Day	Title	Session Objective	Session Objective
1	<ul style="list-style-type: none"> <li>Python Introduction</li> </ul>	<ul style="list-style-type: none"> <li>Introduction to Python – Indentation, Python Variables, Python Data Types – Numbers, String</li> <li>Lists, Tuples, Sets and Dictionaries</li> <li>Python File Handling, Python String Formatting</li> <li>Working with Jupiter Notebook</li> </ul>	<ul style="list-style-type: none"> <li>Python Introduction</li> <li>Python IDLE Installation</li> <li>Python Built in data types</li> <li>Working with Jupyter</li> </ul>
2	<ul style="list-style-type: none"> <li>Role of Python in Data Science</li> <li>Python Packages</li> </ul>	<ul style="list-style-type: none"> <li>Role of Python in Data Science, Data Science Fundamentals, Statistical Tools, Techniques for data science</li> <li>Working with data pre-processing and data visualization using Python - Pandas</li> <li>Working with data pre-processing and data visualization using Python - Numpy</li> <li>Working with data pre-processing and data visualization using Python - Matplotlib</li> </ul>	<ul style="list-style-type: none"> <li>Data Science</li> <li>Python Packages</li> </ul>
3	<ul style="list-style-type: none"> <li>Python Packages</li> <li>Hands on Exercise</li> </ul>	<ul style="list-style-type: none"> <li>Python tkinter - Widgets, Standard attributes, Geometry Management</li> <li>Python tkinter – Widgets - Labels, Buttons, Entry Field, Text Field, Check Buttons, List Box, Radio Buttons, Menu</li> <li><b>Exercise :</b> <ul style="list-style-type: none"> <li>Random Password</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Python Packages</li> <li>TK inter Exercise</li> </ul>

		<p>Generator in Python</p> <ul style="list-style-type: none"> <li>• Convert Text to Speech in Python</li> </ul>	
4	Teaching Methodology Principles	<ul style="list-style-type: none"> <li>• Principles of ANDRAGOGY</li> <li>• Beneficial Features of Observation &amp; Facilitation</li> <li>• Learning Models that are relevant to classroom learning</li> <li>• JAM – just a minute</li> <li>• Importance of Communication</li> <li>• From the baselines of Bloom’s Taxonomy to ARCS</li> </ul>	<ul style="list-style-type: none"> <li>• Dialogue-driven interaction</li> <li>• Role Play &amp; JAM</li> <li>• Case Study, Compare &amp; Contrast</li> <li>• Reading between the lines</li> <li>• Group Discussion</li> <li>• Motivational Stories</li> <li>• Motivational Interview</li> <li>• Constructive Feedback</li> <li>• Successful person’s Unsuccessful stories</li> <li>• Statistics, Data &amp; History</li> <li>• Appreciative Inquiry &amp; Assessment</li> <li>• Problem-solving through Design thinking</li> </ul>
5	Innovation Methodology and Tool	<ul style="list-style-type: none"> <li>• The world’s oldest Innovation Methodology of teaching “Panchatantra” to EdgarDale’s Model.</li> <li>• Tools &amp; Methods for fostering participative &amp; inductive learning</li> <li>• Importance of Mediocre.</li> <li>• Relevance of psychosocial interventions in the class, to ensure the psychological engagement of learners</li> <li>• Not to let down the last benchers</li> <li>• The Need for LST as a part of the regular curriculum.</li> </ul>	

## 10. WORKSHOP PROCEEDINGS

### Day – 1: 30<sup>th</sup> May 2022: Inaugural Session and Python

#### **Inaugural Session:**

The Training Programme began with the inaugural session. The inaugural session was graced by Dr. S.N.V.Ganesh Principal Avanthi Institute of Engineering and Technology, Dr A Chandra Sekhar, HOD-CSE, Avanthi Institute of Engineering and Technology, Mr Sridhar Co-ordinator, APITA and Banoth Prasad CEO/CO-Founder - K4U Infotech. Dr. S.N.V.Ganesh in his inaugural address highlighted the importance of FDP for Teaching Methodologies and made a note about the

importance of python.

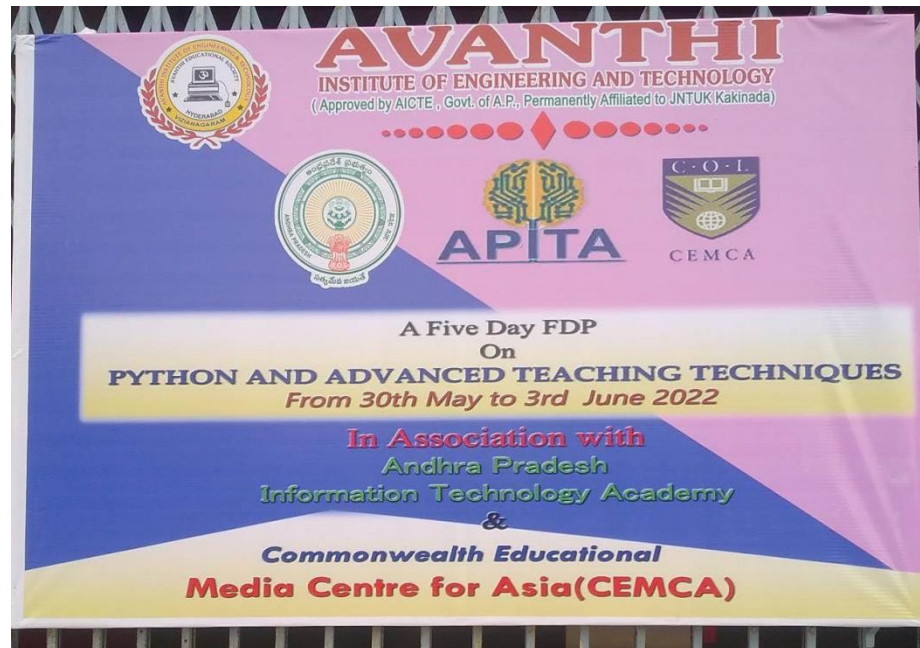


Fig: Banner of the FDP



Fig Inaugural Session on Day 4

## అవంతిలో ఉపాధ్యాయులకు శిక్షణ



### సదస్సులో మాట్లాడుతున్న ప్రిన్సిపాల్ గణేష్

తగరపువలస: చెరకుపల్లి అవంతి ఇంజనీరింగ్ కళాశాలలో కంప్యూటర్ సైన్స్ విభాగంలో 5 రోజులపాటు జరగనున్న ఉపాధ్యాయ శిక్షణ సదస్సు సోమవారం ప్రారంభమయింది. పైథాన్ అండ్ అడ్వాన్స్డ్ టీచింగ్ టెక్నిక్స్పై ఈ శిక్షణను ఆంధ్రప్రదేశ్ టెక్నాలజీ అకాడమీ, కామన్వెల్త్ ఎడ్యుకేషన్ మీడియా సెంటర్ ఫర్ ఆసియా సంయుక్తంగా నిర్వహిస్తున్నట్లు ప్రిన్సిపాల్ ఎస్ఎన్వీ గణేష్ తెలిపారు. పైథాన్పై ఉపాధ్యాయులకు నైపుణ్యాలు, సాంకేతిక రంగంలో వస్తున్న విప్లవాత్మక మార్పులు, మెలకువలు గురించి వివరిస్తారన్నారు. ఈ శిక్షణలో డైరెక్టర్ ఆకుల చంద్రశేఖర్, కంప్యూటర్ సైన్స్ విభాగాధిపతులు దినేష్చంద్ర, శ్రీలక్ష్మి పాల్గొన్నారు.

Fig: Article published in newspaper - Sakshi about the FDP

## Day 1: Session 1 – Python Overview and Built in Data Types

The first day session 1 started with

- Introduction to Python, Indentation, Python Variables, Python Data Types – Numbers, String, Lists, Tuples, Sets and Dictionaries

### What is Python?

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

### What can Python do?

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

### Why Python?

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

- Python can be treated in a procedural way, an object-orientated way or a functional way.

## Python Indentation

- Indentation refers to the spaces at the beginning of a code line.
- Where in other programming languages the indentation in code is for readability only, the indentation in Python is very important.
- Python uses indentation to indicate a block of code.

### Example

```
if 5 > 2:  
    print("Five is greater than two!")
```

Python will give you an error if you skip the indentation:

## Comments

- Python has commenting capability for the purpose of in-code documentation.
- Comments start with a #, and Python will render the rest of the line as a comment:

### Example

Comments in Python:

```
#This is a comment.  
print("Hello, World!")
```

## Python Variables

11. In Python variables are created the moment you assign a value to it:

### Example

Variables in Python:



x

= 5

```
y = "Hello, World!"
```

Python has no command for declaring a variable.

## Variable Names

A variable can have a short name (like x and y) or a more descriptive name (age, carname, total\_volume). Rules for Python variables:

- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and \_)
- Variable names are case-sensitive (age, Age and AGE are three different variables)

## Python Collections (Arrays)

There are four collection data types in the Python programming language:

- **List** is a collection which is ordered and changeable. Allows duplicate members.
- **Tuple** is a collection which is ordered and unchangeable. Allows duplicate members.
- **Set** is a collection which is unordered and unindexed. No duplicate members.
- **Dictionary** is a collection which is unordered, changeable and indexed. No duplicate members.

When choosing a collection type, it is useful to understand the properties of that type. Choosing the right type for a particular data set could mean retention of meaning, and, it could mean an increase in efficiency or security.

## List

- Lists are used to store multiple items in a single variable.

- Lists are one of 4 built-in data types in Python used to store collections of data, the other 3 are Tuple, Set, and Dictionary, all with different qualities and usage.
- Lists are created using square brackets:

### Example

Create a List:

```
thislist = ["apple", "banana", "cherry"]  
print(thislist)
```

### Tuple

- Tuples are used to store multiple items in a single variable.
- Tuple is one of 4 built-in data types in Python used to store collections of data, the other 3 are List, Set, and Dictionary, all with different qualities and usage.
- A tuple is a collection which is ordered and unchangeable.
- Tuples are written with round brackets.

### Example

Create a Tuple:

```
thistuple = ("apple", "banana", "cherry")  
print(thistuple)
```

### Set

- Sets are used to store multiple items in a single variable.
- Set is one of 4 built-in data types in Python used to store collections of data, the other 3 are List, Tuple, and Dictionary, all with different qualities and usage.
- A set is a collection which is *unordered*, *unchangeable\**, and *unindexed*.
- **Note:** Set *items* are unchangeable, but you can remove items and add new items.
- Sets are written with curly brackets.

### Example

Create a Set:

```
thisset = {"apple", "banana", "cherry"}  
print(thisset)
```

### Dictionary

- Dictionaries are used to store data values in key:value pairs.
- A dictionary is a collection which is ordered\*, changeable and do not allow duplicates.
- As of Python version 3.7, dictionaries are *ordered*. In Python 3.6 and earlier, dictionaries are *unordered*.
- Dictionaries are written with curly brackets, and have keys and values:

### Example

Create and print a dictionary:

```
thisdict = {  
    "brand": "Ford",  
    "model": "Mustang",  
    "year": 1964  
}  
print(thisdict)
```

## Day 1: Session 2 – Python File Handling and Working of Jupyter Notebook

The first day session 2 started with

- Python File Handling, Python String Formatting
- Working with Jupyter Notebook

## Python File Open

- File handling is an important part of any web application.
- Python has several functions for creating, reading, updating, and deleting files.

## File Handling

The key function for working with files in Python is the `open()` function.

The `open()` function takes two parameters; *filename*, and *mode*.

There are four different methods (modes) for opening a file:

- `"r"` - Read - Default value. Opens a file for reading, error if the file does not exist
- `"a"` - Append - Opens a file for appending, creates the file if it does not exist
- `"w"` - Write - Opens a file for writing, creates the file if it does not exist
- `"x"` - Create - Creates the specified file, returns an error if the file exists

In addition you can specify if the file should be handled as binary or text mode

- `"t"` - Text - Default value. Text mode
- `"b"` - Binary - Binary mode (e.g. images)

## Syntax

To open a file for reading it is enough to specify the name of the file:

- `f = open("demofile.txt")`

The code above is the same as:

- `f = open("demofile.txt", "rt")`

Because "r" for read, and "t" for text are the default values, you do not need to specify them.

**Note:** Make sure the file exists, or else you will get an error.

### **Web Integrated Development Environment (WIDE): Jupyter**

- With the advent of web applications, a new generation of IDEs for interactive languages such as Python has been developed.
- Starting in the academia and e-learning communities, web-based IDEs were developed considering how not only your code but also all your environment and executions can be stored in a server.
- One of the first applications of this kind of WIDE was developed by William Stein in early 2005 using Python 2.3 as part of his SageMath mathematical software.
- Since December 2011, IPython has been issued as a browser version of its interactive console, called IPython notebook, which shows the Python execution results very clearly and concisely by means of cells. Cells can contain content other than code.
- Since the project has grown so much, IPython notebook has been separated from IPython software and now it has become a part of a larger project: Jupyter.
- The name "Jupyter" was chosen to bring to mind the ideas and traditions of science and the scientific method.
- Additionally, the core programming languages supported by Jupyter are Julia, Python, and R.
- While the name Jupyter is not a direct acronym for these languages (Julia (Ju), Python (Py) and R), it does establish a firm alignment with them.

## Installing the classic Jupyter Notebook interface

### Prerequisite: Python

- While Jupyter runs code in many programming languages, **Python** is a requirement (Python 3.3 or greater, or Python 2.7) for installing the Jupyter Notebook.
- We recommend using the Anaconda distribution to install Python and Jupyter. We'll go through its installation in the next section.

## Installing Jupyter using Anaconda and conda

For new users, we **highly recommend** installing Anaconda. Anaconda conveniently installs Python, the Jupyter Notebook, and other commonly used packages for scientific computing and data science.

Use the following installation steps:

1. Download Anaconda. We recommend downloading Anaconda's latest Python 3 version (currently Python 3.9).
2. Install the version of Anaconda which you downloaded, following the instructions on the download page.
3. Congratulations, you have installed Jupyter Notebook. To run the notebook:

```
jupyter notebook
```

See Running the Notebook for more details.

## Alternative for experienced Python users: *Installing Jupyter with pip*

### Important

Jupyter installation requires Python 3.3 or greater, or Python 2.7. IPython 1.x, which included the parts that later became Jupyter, was the last version to support Python 3.2 and 2.6.

As an existing Python user, you may wish to install Jupyter using Python's package manager, [pip](#), instead of Anaconda.

First, ensure that you have the latest pip; older versions may have trouble with some dependencies:

```
pip3 install --upgrade pip
```

Then install the Jupyter Notebook using:

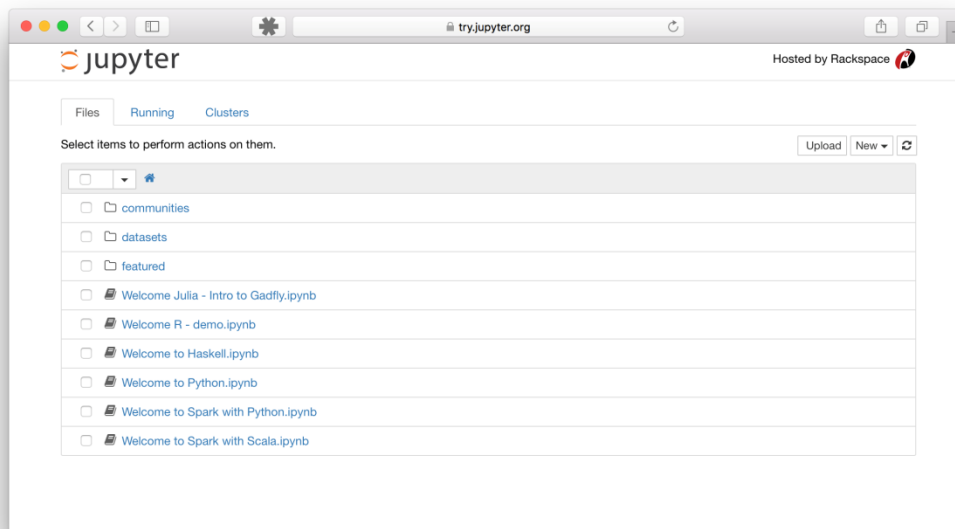
```
pip3 install jupyter
```

## Running the Notebook

1. Start the notebook server from the command line:

```
jupyter notebook
```

2. You should see the notebook open in your browser.



## The Jupyter Notebook Environment

Installing Jupyter Notebook using pip : **pip install notebook**

To run the notebook, run the following command at Command Prompt:

**jupyter notebook**

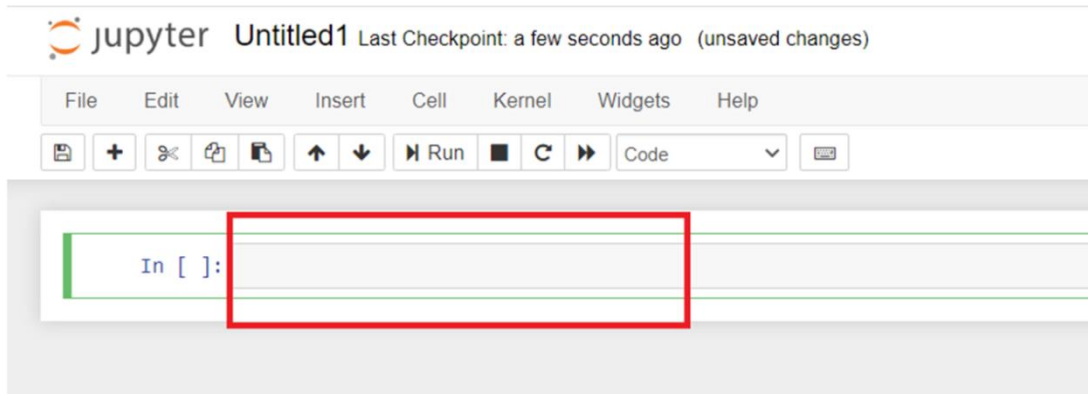
The browser will immediately be launched displaying the Jupyter notebook home-page, whose URL is <http://localhost:8888/tree>.

The image displays two screenshots of the Jupyter Notebook web interface. The top screenshot shows the 'tree' view of the file system. The browser address bar is highlighted with a red box, showing the URL 'localhost:8892/tree'. The Jupyter logo and 'Quit' and 'Logout' buttons are visible. Below the navigation tabs (Files, Running, Clusters), there is a prompt 'Select items to perform actions on them.' and buttons for 'Upload', 'New', and a refresh icon. A table lists files and folders:

	Name	Last Modified	File size
<input type="checkbox"/>	/		
<input type="checkbox"/>	catdog	4 days ago	
<input type="checkbox"/>	catdog1	4 days ago	
<input type="checkbox"/>	catdog2	3 days ago	
<input type="checkbox"/>	keras-io-master	4 days ago	
<input type="checkbox"/>	x-ray	7 hours ago	

The bottom screenshot shows the same interface, but with the 'New' dropdown menu open. The 'Python 3' option is highlighted with a red box. Other options include 'Text File', 'Folder', and 'Terminal'. A 'Create a new' button is also visible. The file list now includes 'Untitled.ipynb' which is 'Running seconds ago' and has a size of '72 B'.





```
In [1]: print("Hello, World!")
```

Hello, World!

```
In [2]: print("Hello, World!")  
#This is a comment
```

Hello, World!

```
In [3]: #This is a comment  
#written in  
#more than just one line  
print("Hello, World!")
```

Hello, World!

## Day 1 – Objectives

- Participants gained knowledge on
  - Understanding concepts of python
  - Understanding concepts of python –Built in data types
  - Understanding concepts of python file handling
  - Working with jupyter notebook



## **Day – 2: 31<sup>st</sup> May 2022: Python Package – Pandas, NumPy and Matplotlib**

### **Day 2: Session 1 – Package – Data Science and Pandas**

The second day session 1 started with

- Role of Python in Data Science, Data Science Fundamentals, Statistical Tools, Techniques for data science
- Working with data pre-processing and data visualization using Python - Pandas

#### **Toolbox for Data Scientists**

- The toolbox of any data scientist, as for any kind of programmer, is an essential ingredient for success and enhanced performance. Choosing the right tools can save a lot of time and thereby allow us to focus on data analysis.
- The most basic tool to decide on is which programming language we will use. Many people use only one programming language in their entire life: the first and only one they learn.

- For many, learning a new language is an enormous task that, if at all possible, should be undertaken only once.
- The problem is that some languages are intended for developing high-performance or production code, such as C, C++, or Java, while others are more focused on prototyping code, among these the best known are the so-called scripting languages: Ruby, Perl, and Python.

### **Fundamental Python Libraries for Data Scientists**

- The Python community is one of the most active programming communities with a huge number of developed toolboxes.
- The most popular Python toolboxes for any data scientist are NumPy, SciPy, Pandas, and Scikit-Learn.

### **What is Pandas?**

- Pandas is a Python library used for working with data sets.
- It has functions for analyzing, cleaning, exploring, and manipulating data.
- The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.

### **Why Use Pandas?**

- Pandas allows us to analyze big data and make conclusions based on statistical theories.
- Pandas can clean messy data sets, and make them readable and relevant.
- Relevant data is very important in data science.

### **What Can Pandas Do?**

Pandas gives you answers about the data. Like:

Is there a correlation between two or more columns?

What is average value?

Max value?

Min value?

Pandas are also able to delete rows that are not relevant, or contains wrong values, like empty or NULL values. This is called cleaning the data.

### **Where is the Pandas Codebase?**

The source code for Pandas is located at this github repository

<https://github.com/pandas-dev/pandas>

### **Installation of Pandas**

If you have Python and PIP already installed on a system, then installation of Pandas is very easy.

**Install it using this command:**

***C:\Users\Your Name>pip install pandas***

### **Import Pandas**

Once Pandas is installed, import it in your applications by adding the "import" keyword:

**Syntax : import pandas**

Now Pandas is imported and ready to use.

```
import pandas

mydataset = {
    'cars': ["BMW", "Volvo", "Ford"],
    'passings': [3, 7, 2]
}

myvar = pandas.DataFrame(mydataset)
```

```
print(myvar)
```

### OUTPUT :

```
cars passings
0  BMW      3
1  Volvo    7
2  Ford     2
```

### Example: Create DataFrame from List of Lists with Column Names & Index

```
import pandas as pd

#list of lists

data = [['a1', 'b1', 'c1'],
        ['a2', 'b2', 'c2'],
        ['a3', 'b3', 'c3']]

columns = ['C1', 'C2', 'C3']
index = ['R1', 'R2', 'R3']

df = pd.DataFrame(data, index, columns)
print(df)
```

### OUTPUT :

```
C1 C2 C3
R1 a1 b1 c1
R2 a2 b2 c2
R3 a3 b3 c3
```

## Day 2: Session 2 – Package – NumPy and Matplotlib

The second day session 2 started with

- Working with data pre-processing and data visualization using Python - Numpy

- Working with data pre-processing and data visualization using Python - Matplotlib

### **What is NumPy?**

- NumPy stands for Numerical Python.
- NumPy is a Python library used for working with arrays.
- It also has functions for working in domain of linear algebra, fourier transform, and matrices.
- NumPy was created in 2005 by Travis Oliphant. It is an open source project and we can use it freely.

### **Why Use NumPy?**

- In Python we have lists that serve the purpose of arrays, but they are slow to process.
- NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.
- The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.
- Arrays are very frequently used in data science, where speed and resources are very important.

### **Why is NumPy Faster Than Lists?**

- NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently.
- This behavior is called locality of reference in computer science.

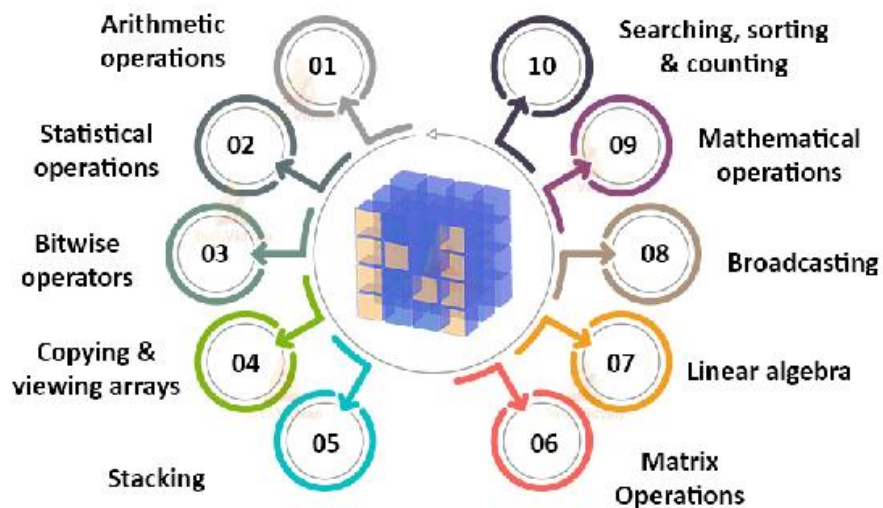
- This is the main reason why NumPy is faster than lists. Also it is optimized to work with latest CPU architectures.

### Where is the NumPy Codebase?

The source code for NumPy is located at this github repository

<https://github.com/numpy/numpy>

### Uses of NumPy



### Installation of NumPy

If you have Python and PIP already installed on a system, then installation of NumPy is very easy.

Install it using this command:

```
C:\Users\Your Name>pip install numpy
```

If this command fails, then use a python distribution that already has NumPy installed like, Anaconda, Spyder etc.

## Import NumPy

Once NumPy is installed, import it in your applications by adding the import keyword:

```
import numpy
```

Now NumPy is imported and ready to use.

### Example :

Use a List to create a NumPy array:

```
import numpy  
arr = numpy.array([1, 2, 3, 4, 5])  
print(arr)  
[1 2 3 4 5]
```

## Day 2 – Objectives

- Participants gained knowledge on
  - Understanding the role of data science,
  - Working with data pre-processing and data visualization using Python - Pandas, NumPy and Matplotlib





## Day – 3: 01<sup>st</sup> June 2022: Python Package – TK Inter and Exercise

### Day 3: Session 1 – Python Package – TK Inter

The third day session 1 started with

- Python Package TK Inter,
- Python tkinter - Widgets, Standard attributes, Geometry Management,
- Widgets - Labels, Buttons, Entry Field, Text Field, Check Buttons, List Box, Radio Buttons, and Menu.

### Python GUI Programming

Following are options for Python GUI Programming.

We may use one of the following alternatives for programming a Python GUI:

- a. Tkinter :** Python ships with the Tk GUI toolkit. Tkinter is an interface to this.
- b. PyQt :** A Python binding of the cross-platform GUI toolkit Qt, PyQt is implemented as a plug-in for Python.
- c. wxPython :** wxWidgets is a cross-platform GUI API for Python; wxPython is a wrapper for this.

### Python Tkinter

- Python provides the standard library Tkinter for creating the graphical user interface for desktop based applications.
- Developing desktop based applications with python Tkinter is not a complex task

### To create a tkinter app:

1. Importing the module – tkinter
2. Create the main window (container)
3. Add any number of widgets to the main window
4. Apply the event Trigger on the widgets.

### Syntax:

```
import tkinter
m = tkinter.Tk()
''' widgets are added here '''
m.mainloop()
```

There are two main methods used which the user needs to remember while creating the Python application with GUI.

### Example: How to Change Background Color of Tkinter Button during Mouse Click?

You can change the button's background color, while the button is pressed using mouse, using activebackground property of the button.

```
from tkinter import *
gui = Tk(className='Python Examples - Button')
gui.geometry("500x200")
# create button
button = Button(gui, text='My Button', width=40, height=3, bg='blue', fg='white',
activebackground='green', activeforeground='yellow')
# add button to gui window
button.pack()
gui.mainloop()
```

### Example : Create and add color to entry field using tkinter

```
from tkinter import *
gui = Tk(className='Python Examples - Entry Field')
gui.geometry("500x200")
# create label field
l1=Label(gui, text='First Name').grid(row=0)
l2=Label(gui, text='Last Name').grid(row=1)
# create entry field
e1 = Entry(gui,fg="white", bg="light green", width=20).grid(row=0, column=1)
```

```
e2 = Entry(gui, fg="white", bg="light green", width=20).grid(row=1, column=1)
gui.mainloop()
```

### Day 3: Session 2 – Python Exercise

The third day session 2 started with Exercise:

- Random Password Generator in Python
- Convert Text to Speech in Python

#### Password Generator Python Project

The objective of this project is to create a password generator using python. The password generator project will be build using python modules like Tkinter, random, string, pyperclip.

In this project, the user has to select the password length and then click on the “**Generate Password**” button. It will show the generated password below. If the user clicks on the “**Copy To Clipboard**” button, then it will copy the password automatically.

Project Prerequisites

To build this project we will use the basic concept of python and libraries – Tkinter, pyperclip, random, string.

- **Tkinter** is a standard GUI library and is one of the easiest ways to build a GUI application.
- **pyperclip** module allows us to copy and paste text to and from the clipboard to your computer
- **The random** module can generate random numbers
- **string** module contains a number of functions to process the standard python string.

To install the libraries we can use pip installer from the command line:

- pip install tkinter
- pip install pyperclip

- pip install random
- pip install strings

## Project File Structure

Let's check the step to build a Password Generator using Python

- Import modules
- Initialized Window
- Select Password Length
- Define Functions

### Steps to create random password generator

#### 1. Import Libraries

The first step is to import libraries

```
from tkinter import *  
import random, string  
import pyperclip
```

#### 2. Initialize Window

```
root = Tk()  
root.geometry("400x400")  
root.resizable(0,0)  
root.title("DataFlair - PASSWORD GENERATOR")
```

- **Tk()** initialized tkinter which means window created
- **geometry()** set the width and height of the window
- **resizable(0,0)** set the fixed size of the window
- **title()** set the title of the window

```
Label(root, text = 'PASSWORD GENERATOR' , font ='arial 15 bold').pack()
```

```
Label(root, text ='DataFlair', font ='arial 15 bold').pack(side = BOTTOM)
```

**Label()** widget use to display one or more than one line of text that users can't able to modify.

- **root** is the name which we refer to our window
- **text** which we display on the label
- **font** in which the text is written
- **pack** organized widget in block

### 3. Select Password Length

```
pass_label = Label(root, text = 'PASSWORD LENGTH', font = 'arial 10 bold').pack()
```

```
pass_len = IntVar()
```

```
length = Spinbox(root, from_ = 8, to_ = 32 , textvariable = pass_len , width = 15).pack()
```

- **pass\_len** is an integer type variable that stores the length of a password.
- To select the password length we use **Spinbox()** widget.
- **Spinbox()** widget is used to select from a fixed number of values. Here the value from 8 to 32

### 4. Function to Generate Password

```
pass_str = StringVar()
```

```
def Generator():
```

```
password = "
```

```
for x in range (0,4):
```

```
password = random.choice(string.ascii_uppercase) +
```

```
random.choice(string.ascii_lowercase) + random.choice(string.digits) +
```

```
random.choice(string.punctuation)
```

```
for y in range(pass_len.get()- 4):
```

```
password = password + random.choice(string.ascii_uppercase +
```

```
string.ascii_lowercase + string.digits + string.punctuation)
```

```
pass_str.set(password)
```

- **pass\_str** is a string type variable that stores the generated password
- **password** = "" is the empty string

- First loop will generate a string of length 4 which is a combination of an uppercase letter, a lowercase letter, digits, and a special symbol and that string will store in password variable.
- The second loop will generate a random string of length entered by the user – 4 and add to the password variable. Here we minus 4 to the length of the user because we already generate the string of length 4.

We have done this because we want a password which must contain an uppercase, a lowercase, a digit, and a special symbol.

Now the password is set to the **pass\_str()** variable.

```
Button(root, text = "GENERATE PASSWORD" , command = Generator ).pack(pady=5)
```

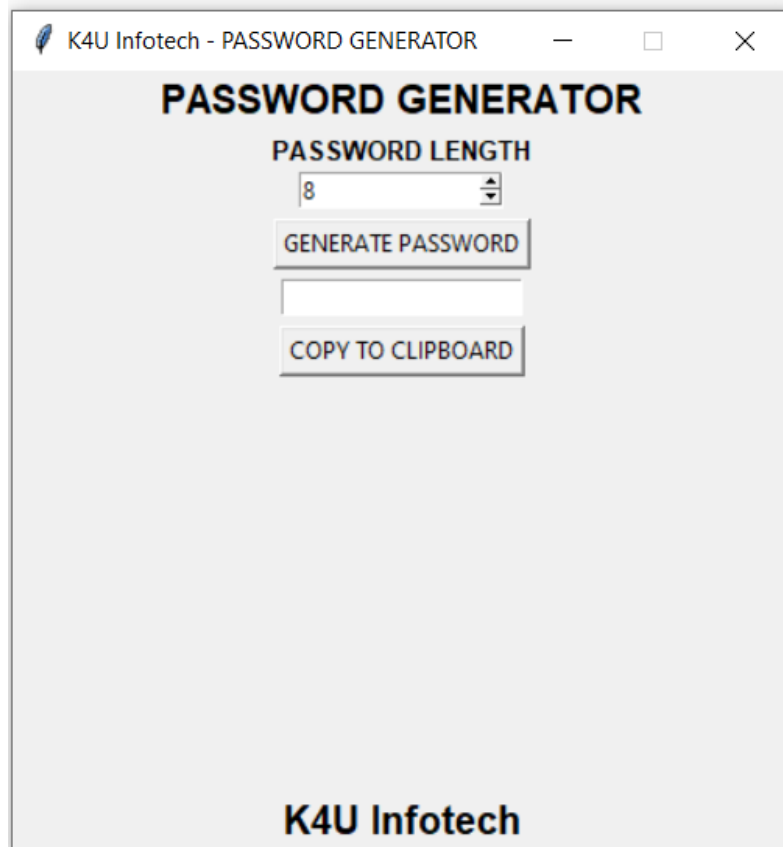
```
Entry(root , textvariable = pass_str).pack()
```

- **Button()** widget used to display button on our window
- **command** is called when the button is click
- **Entry()** widget used to create an input text field
- **textvariable** used to retrieve the current text to the entry widget

## 5. Function to Copy Password

```
def Copy_password():
    pyperclip.copy(pass_str.get())
    Button(root, text = 'COPY TO CLIPBOARD', command =
    Copy_password).pack(pady=5)
    pyperclip.copy() used to copy the text to clipboard
```

**Python Password Generator Output:**



### Convert Text to Speech in Python



### Day 3 – Objectives

- Participants gained knowledge on
  - Python Package TK Inter
  - Developed Application using TK inter - Random Password Generator in Python and Convert Text to Speech in Python



### Day 4: 02<sup>nd</sup> June 2022: Session 1 and 2 ATM

After the ICE BREAKING sessions, the first topic was about Individualized Instruction Model By **Keller Plan**, Success Approximation Model By **Dr. Michael Allen** .The break was up with **Edgar Dale’s Model**- Giving Following Opportunity To Participants For 90% Retention



<b>LEARNING MODELS EXPLAINED</b>	<b>MAIN THEME</b>	<b>OUTPUT OF EXPLANATION</b>
BENJAMIN BLOOM's COGNITIVE TAXONOMY	<p>This theory deals with the six levels of a learner's cognition and understanding of a specific subject.</p> <p>The Facilitator affirmed that the level is very much subjected specific.</p> <p>A Learner/student who is in the higher level in one subject may be in the lowermost level of another subject</p>	The learners achieved conceptual clarity regarding the different Cognitive Levels of individuals in different subjects
ROBERT GAGNE's Nine Steps of Instruction	<p>This model indicates that there are <i>nine types</i> of educational instructions arranged in a logical sequence.</p> <p>If an Educator follows the nine instructions in the structured logical sequence, then the learners will achieve the desired learning goal and transfer their learned inputs to others.</p>	The learners got apprised regarding the logical sequence of activities that are to be executed by a Trainer /Facilitator in a Learning Session for facilitating the learners in ensuring retention of the learned content and developing within them the ability to transfer the learned content to others
EDGAR DALE's Model of retention of learning inputs	The model suggests that different types of academic activities lead to different levels of retention of learned inputs	The learners got apprised regarding the percentage of retention of learned inputs by the learners under the influence of various sorts of Teaching Activities
MAY & DOOB's Collaborative Learning Model	This learning model indicates that cooperation and collaboration among a group of learners can lead to the achievement of learning outcomes	The learners comprehended how the exchange of views, group discussion, task distribution, and the convergence of thoughts can play extremely significant roles in the learning process.
CHARLES REIGELUTH'S Elaboration Model	The model suggests that an Educator must present the learning materials in a sequential pattern viz. from the simplest level to the most complex level	The learners understood the significance of arranging and presenting the learning content through logical sequences

<p>The Oldest of Innovation Teaching Methodology</p> <p><b>“Panchatantra”</b></p> <p>By</p> <p>- Vishnu Sharma.</p>	<p>This model is mainly concerned with the week &amp; mediocre / Average participants.</p> <p>First time in the history of “Gurukulas”, this is introduced on the special request by the then King for his three sons.</p>	<p>Faculties were overwhelmed to know , llisten and experience the relation and co-relation of connectivity to this present technologies and trends.</p>
---	--	--

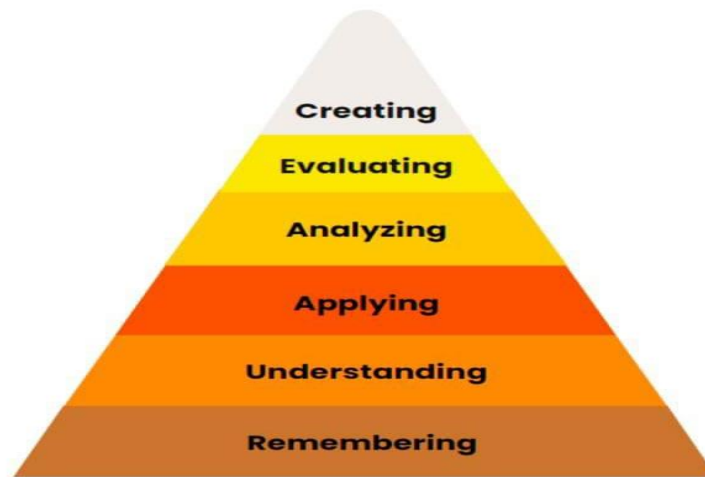
**5 E’s of effective teaching:**



**Passive v/s active learning:**



## **BLOOM'S TAXONOMY:**



### **Day 4 – Objectives**

- Inauguration & Ice Breaking
- Introduction to Value Education – Andragogy
- Introduction to 21st Century skills
- Advanced Teaching Techniques/ Methods
- Robert Gagne's Nine Steps Of Instruction
- Benjamin Bloom's Cognitive Taxonomy
- Edgar Dale's Model Of Retention Of Learning Inputs
- What I am expecting to get & what I do
- Harmony In The Family And Society
- PANCHATANHIRAM –Teaching Techniques
- May & Doob's Collaborative Learning Model
- Pre Training Assessment- Discussion

**Day 5: 03<sup>rd</sup> June 2022: Session 1 and 2 ATM**

<b>LEARNING MODELS EXPLAINED</b>	<b>MAIN THEME</b>	<b>OUTPUT OF EXPLANATION</b>
JOHN SWELLER'S Cognitive Load Model	<p>This model suggests that to ensure the retention of learning inputs in participants' long-term memory, the learning content should be delivered in a fragmented pattern, part by part.</p> <p>If a large volume of content is foisted upon the learners then the cognitive load will get enhanced to a large extent.</p> <p>Therefore, the learned inputs will not be retained in long-term memory.</p>	The learners understood the significance of presenting a large volume of content in a fragmented pattern instead of continually imposing a huge volume of content upon the learners.
JOHN KELLER'S ARCS MODEL	<p>This model reflects the linear sequence of four activities that are to be followed by educators to</p> <ul style="list-style-type: none"> <li>• Draw the attention of the participants</li> <li>• Establish relevance of the learning materials to the prior knowledge or the occupational life of the participants</li> <li>• Build up the confidence of the learners</li> <li>• Satisfying the learners by the achieved learning outcome</li> </ul>	The learners understood how to draw the attention of the participants, present relevant inputs to them, build up their confidence, the participants and satisfy the participants in learning sessions.
JEROME BRUNER'S Discovery Model	This model suggests that the Probing Inquiry from the Educator facilitates the participants to introspect and unleash their latent creativity and analytical competencies.	The learners to understand the significance of Inquiry-Based Instructions for the discovery of the hidden creative flair, insight, intuition as well as the analytical power of the participants, embedded within themselves
HOWARD BURROWS' Problem Based Learning Model	This model suggests that if the participants are provided with problems to solve rather than contents to memorize, then the creative and analytical competencies of the learners will get enhanced and unleashed.	The learners understood how problem-solving exercises can enhance critical thinking as well as creative thinking competency of the participant's

<p>MICHAEL ALLEN's Success Approximation Model (SAM)</p>	<p>This model encourages participants centric learn. Here, primarily the prototype of the learning material and finally the prepared instructional material are evaluated by the participants.</p> <p>If the participants suggest any changes, then to ensure the ease of learning for the participants, the changes are engineered accordingly both at the prototype as well as at the final material.</p>	<p>The learners perceived the significance of the customization /re-engineering of the prototype of learning materials as well as the final learning material based on the feedback of the participants</p> <p>The learners understood the significance of Student-Centric Instructional Design.</p>
<p>KELLER PLAN's Individualized Instructional Model</p>	<p>This model is mainly concerned with the heterogeneity of the learners. Since each learner differs from other learners in the learning capacity, style pace, and approach, hence the instructional material should be designed based on the unique capacity, need style standard, and pace of each learner. Within an allotted time frame each learner learns individually in his/her way</p>	<p>The learners deciphered the importance of designing learning materials and determining the methodology of facilitating the participants based on the intellectual standard, learning style, and the learning pace of each student.</p>
<p>The Oldest Innovation of Teaching Methodology</p> <p><b>"Panchatantra"</b></p> <p>By - Vishnu Sharma.</p>	<p>This model is mainly concerned with the weak &amp; mediocre / Average participants. First time in the history of "Gurukulas", this is introduced on the special request by the then King for his three sons. Panchatantra means Five Treatises/ chapters. Given learning capacity, style, pace, and approach, hence the instructional materials are all the stories about Animals, Birds &amp; Plants/ Nature. This Method of teaching gives not only the knowledge to the students but also the new experience with each &amp; every week student the guru faces every time.</p>	<p>The participants/ Shishyas for the imaginative thinking and stored in the permanent memories. Within an allotted time frame each learner learns individually in his perception like the Guru's Version. This will be the best example of the " Experience makes Man Perfect "</p>

## Day 5 – Objectives

- John Sweller's Cognitive Load Model
- Values In Human-To-Human Relationship
- John Keller's Arcs Model
- Innovation – Andragogy – Management
- 'Respect' – As The Right Evaluation
- 'Trust' – The Foundational Value In Relationship

- Jerome Bruner's Discovery Model
- Howard Burrows' Problem Based Learning Model
- Assessment result
- PANCHATANTHIRAM – Teaching Techniques
- What we got & how to do



## 12. CLOSING CEREMONY

The Training Programme concluded with the closing ceremony. The closing ceremony was graced by Dr. Muppala Janardhana Raju Principal - Siddartha Institute of Science and Technology, Dr. M A Manivasagam HOD - CSE Dept - Siddartha Institute of Science and Technology, S. Dinakar Reddy (Corporate Master Trainer & Facilitator, Mentor in Innovations & Entrepreneurship, Instructional Designer of LST) and participants given feedback on the FDP.



Fig: Feedback from Participant

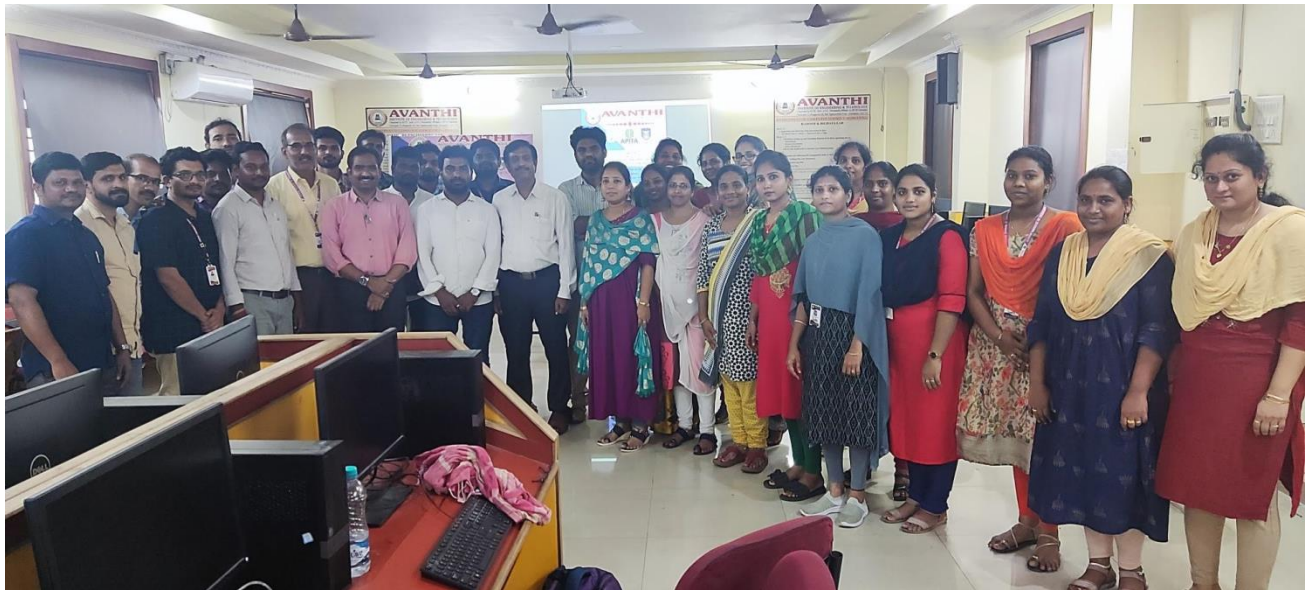


Fig: Closing Ceremony on day 3



Fig: Closing Ceremony on Day 5



Fig: Group picture of participants on day 5 on Closing Ceremony



## 13. Participant's List (Annexure – A)

S. No.	College Name	Faculty NAME	Department	Mobile No	E-mail ID
1	Sri Venkatesa Perumal College of Engineering & Technology	SREEJYOTHSNA ANKAM	CSE	9985461569	jyothimymail@gmail.com
2	Sri Venkatesa Perumal College of Engineering & Technology	ALUMURU MAHESH REDDY	ECE	9110581143	alumuru.mahesh@gmail.com
3	HIMAJA DEGREE COLLEGE PUTTUR	ERASAPPA MURALI	CSE	9618976513	murali.e19@gmail.com
4	SreeRama Engineering College	T.L.Rekha Reddy	ECE	9347321955	rekhareddy@gmail.com
5	SRI VENKATESWARA UNIVERSITY COLLEGE OF ENGINEERING	Dr.Sandhya Tatekalva	CSE	9701110222	dr.sandhyasatish@gmail.com
6	Siddartha Institute of Science and Technology, Puttur	Dr.M.JANARDHANA RAJU	ECE	8297114999	raju243102@gmail.com
7	Siddartha Institute of Science and Technology, Puttur	Dr.B.SAROJA	ECE	8291728999	sistk4e.ece@gmail.com
8	Siddartha Institute of Science and Technology, Puttur	Dr.P.CHANDRA SEKHAR	ECE	9865919953	chandrushiva2013@gmail.com
9	Siddartha Institute of Science and Technology, Puttur	Dr.BASAVARAJ G KUDAMBLE	ECE	9741345013	sistk.bgk2021@gmail.com
10	Siddartha Institute of Science and Technology, Puttur	Dr.GUNASEKARAN K	ECE	9790130194	guna.k77@gmail.com
11	Siddartha Institute of Science and Technology, Puttur	Dr. SAVITESH MADHULIKA SHARMA	ECE	8126761060	college.savitesh@gmail.com
12	Siddartha Institute of Science and Technology, Puttur	Dr.R.RAJ KUMAR	ECE	9842524419	rajcumarramasami@gmail.com
13	Siddartha Institute of Science and Technology, Puttur	Dr.J.V.ANAND	ECE	9994088309	jvanand87@gmail.com
14	Siddartha Institute of Science and Technology, Puttur	DABBARA JAYANAYUDU	ECE	9866280186	d.jayanaidu@gmail.com
15	Siddartha Institute of Science and Technology, Puttur	L.SIVAPRASAD	ECE	9581258789	sivaprasadsistk@gmail.com
16	Siddartha Institute of Science and Technology, Puttur	M VIJAYA LAKSHMI	ECE	7702956668	viji2024@gmail.com
17	Siddartha Institute of Science and Technology, Puttur	P MANI MOHAN	ECE	9000220230	manimohanpydi@gmail.com
18	Siddartha Institute of Science and Technology, Puttur	A J REUBEN THOMAS RAJ	ECE	8248486219	reuben.thms007@gmail.com
19	Siddartha Institute of Science and Technology, Puttur	C LAXMANA SUDHEER	ECE	9985249947	laxmana018@gmail.com
20	Siddartha Institute of Science and Technology, Puttur	K.SANDEEP KUMAR	ECE	9705422293	krishnam.sandeep@gmail.com
21	Siddartha Institute of Science and Technology, Puttur	T PRASAD	ECE	8096497112	sistkprasad.nss@gmail.com
22	Siddartha Institute of Science and Technology, Puttur	A.HESANTH KUMAR	ECE	9500779255	iamhesanth@gmail.com
23	Siddartha Institute of Science and Technology, Puttur	N CHAMANTHI	ECE	8897903327	chamanthi.472@gmail.com
24	Siddartha Institute of Science and Technology, Puttur	P MADHAVI CHOWDARY	ECE	9676938374	madhavichowdary66@gmail.com
25	Siddartha Institute of Science and Technology, Puttur	R LEELAVATHI	ECE	8825819891	leelavathi.leelar@gmail.com
26	Siddartha Institute of Science and Technology, Puttur	A KIRAN KUMARI	ECE	9642770530	kirankumari424@gmail.com
27	Siddartha Institute of Science and Technology, Puttur	C.THAMBI DORAI	ECE	8886808515	thambiedison@gmail.com

28	Siddartha Institute of Science and Technology, Puttur	MA PANDIYAN	ECE	8686428869	mapandiyani431@gmail.com
29	Siddartha Institute of Science and Technology, Puttur	Dr.M.A. MANIVASAGAM	CSE	7010444267	mvsistk@gmail.com
30	Siddartha Institute of Science and Technology, Puttur	V.GOPI	CSE	9491448976	nadh.gopi@gmail.com
31	Siddartha Institute of Science and Technology, Puttur	E.MURALI	CSE	8297229655	sai4murali@gmail.com
32	Siddartha Institute of Science and Technology, Puttur	P DEVAKI	CSE	9010436000	mrsdevaki0@gmail.com
33	Siddartha Institute of Science and Technology, Puttur	M MANIVANNAN	CSE	9444209668	sistkcse.manivannan@gmail.com
34	Siddartha Institute of Science and Technology, Puttur	R.PRIYADARSHINI	CSE	9952896341	sistkcse.priyadarshini@gmail.com
35	Siddartha Institute of Science and Technology, Puttur	R PURUSHOTHAMAN	CSE	9445655293	purush82@gmail.com
36	Siddartha Institute of Science and Technology, Puttur	G.RAVI KUMAR	CSE	9581997453	sistkcse.ravikumar@gmail.com
37	Siddartha Institute of Science and Technology, Puttur	S.JYOSHNA	CSE	9963632746	jyoshnasriram@gmail.com
38	Siddartha Institute of Science and Technology, Puttur	A.SUREKHA	CSE	9177748694	surekhavitw530@gmail.com
39	Siddartha Institute of Science and Technology, Puttur	T.KATAIAH	CSE	7569636537	kata09573@gmail.com
40	Siddartha Institute of Science and Technology, Puttur	P.GOVINDAIAH	CSE	9704582398	govindcse513@gmail.com
41	Siddartha Institute of Science and Technology, Puttur	D SREEKANTH	CIVIL	8331022890	dsreekanthcivil@gmail.com
42	Siddartha Institute of Science and Technology, Puttur	Dr.K.MURALI KUMAR	EEE	8179253254	muralikumaphd@gmail.com
43	Siddartha Institute of Science and Technology, Puttur	A R INBARAJ	ME	9949069048	chinna4795@gmail.com
44	Siddartha Institute of Science and Technology, Puttur	SHARAD KULKARNI	ECE	8886168587	keshavsharad68@gmail.com
45	SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY	BHANUPRAKASH M	CSIT	7981504855	mmbaluprakash@gmail.com
46	SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY	P.BALAJI	MCA	9885482633	sietkmca.balaji@gmail.com
47	SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY	Ch SIVASANKAR	CSIT	9885826865	chsivasankar55@gmail.com
48	SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY	BHUKYA RAJA KUMAR	CSIT	9030496306	rajakumar.bhukya@gmail.com
49	SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY	Dr.TAMMINENI SREELATHA	ECE	9490865096	sreelatha457@gmail.com
50	SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY	P PAVAN KUMAR	ECE	7989198636	pavan.android961@gmail.com
51	SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY	DAKSHIRAJU VISWASAHITHYA	CSIT	8712232756	vishnudakshi@gmail.com
52	SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY	NEELAM POORNIMA	CSIT	8247702669	neelampoornimaa@gmail.com

Thank you