ACADEMIC PLANNER

For

***Introduction to python programming***

***Subject code: AI503PC***

***Presented***

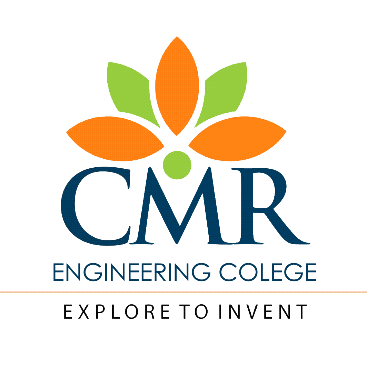
***by***

**Mrs. M.SRIKALA(Asst.Professor)**

Department of

**Computer Science and Engineering [AI&ML]**

**(AY:2025-26)**



**CMR ENGINEERING COLLEGE**

(Approved by AICTE-New Delhi, Affiliated to J.N.T.U, Hyderabad)

Kandlakoya(v),Medchal Road,Hyderabad-501 401,Telangana State, India .Website: [www.cmrec.ac.in](http://www.cmrec.ac.in/)

ACADEMIC PLANNER

**SUBJECT: INTRODUCTION TO PYTHON PROGRAMMING**

**(III B.TECH –I SEM)**

**S.NO CONTENT**

1. **- Preamble/Introduction**

# - Prerequisites

1. **- Objectives and Outcomes**

# - Syllabus

* 1. **CMREC Autonomous-R22 2.GATE**

1. **- List of Expert Details** (Local/National/International with Contact details/Profile link/Blogs/their research Contribution towards the subject)

# - Journals with minimum 5 reference paper for literature study

1. **- Subject -Lesson plan**
2. **- Suggested Books (**prescribed and References)
3. **- Websites for self-learning Resources like**  [www.geeksforgeeks.org,](http://www.geeksforgeeks.org/) www.w3schools.com, [Coursera](https://www.theeducationmagazine.com/word-art/best-educational-websites/#Coursera), [edX](https://www.theeducationmagazine.com/word-art/best-educational-websites/#edX), [Udemy](https://www.theeducationmagazine.com/word-art/best-educational-websites/#Udemy), [Khan Academy,](https://www.theeducationmagazine.com/word-art/best-educational-websites/#Khan%20Academy) NPTEL etc along Registration procedures.

# - Question Banks

* 1. **JNTUH/Model papers 2.GATE**

# - Two case study presentations with Project/ Product/

**Model /prototypes/ Industrial applications.**

**(12) - Assignment Question/Innovative Assignments sets. (13) - List of topics for students Seminars with Guidelines (14) - STEP/Course material in softcopy**

# (15) - Expert Lectures with topics & Schedules (if any)

## Preamble/Introduction:-

Python is **a widely used general-purpose, high level programming language**. It was created by Guido Van Rosum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability and its syntax allows programmers to express their concepts in fewer lines of code

## Prerequisites:-

* 1. A course on “C Programming and Data Structures”.
  2. A course on “Object Oriented through C++”.

## Objectives:-

1. T**o introduce the fundamentals of Python programming**, including syntax, variables, and data types.
2. **To develop logical thinking and problem-solving skills** using control structures such as loops and conditional statements.
3. **To implement modular programming** using functions and understand the concept of code reusability.
4. **To handle files and exceptions** effectively for robust program execution.
5. **To apply object-oriented programming concepts** such as classes and objects in Python.

## Course Outcomes:-

## By the end of this course, students will be able to:

## Write and execute Python programs using appropriate syntax, data types, and variables.

## Apply decision-making and looping constructs to solve computational problems.

## Develop modular programs using functions and built-in modules.

## Perform file operations and handle exceptions for reliable program execution.

## Implement object-oriented programming concepts like classes and inheritance in Python applications.

## 1. AUTONOMOUS-R22-SYLLABUS :-( AI503PC) UNIT –I

**Introduction to Python**, Installing Python. How a Program Works, Using Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, Operators. Type conversions, Expressions, More about Data Output.

**Decision Structures and Boolean Logic**: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables.

**Repetition Structures:** Introduction, while loop, for loop, Calculating a Running Total, Input Validation Loops, Nested Loops.

**Data types and Expressions:** Strings, Assignment and Comments, Numeric Data Types and Character Sets, Expressions, Functions and Modules.

## UNIT - II

**Control Statements:** Definite Iteration, Formatting Text for Output, Selection, Conditional Iteration. **File and Exceptions:** Introduction to File Input and Output, Using Loops to Process Files, Processing Records, Exceptions.

**Functions:** Introduction, Defining and Calling a Void Function, Designing a Program to Use Functions, Local Variables, Passing Arguments to Functions, Global Variables and Global Constants, Value-Returning Functions-Generating Random Numbers, The math Module, Storing Functions in Modules.

## UNIT - III

**Strings and Text Files:** Accessing Characters and Substrings in a String, Strings and Number System, String Methods, Basic String Operations, String Slicing, Testing, Searching, and Manipulating Strings. Text Files, Data Encryption, Lists, Introduction to Lists, List slicing, Finding Items in Lists with the in Operator, List Methods and Useful Built-in Functions, Copying Lists, Processing Lists, Two- Dimensional Lists, Tuples Sequences, Tuples**.**

**Dictionaries and Sets:** Dictionaries, Sets, Serializing Objects.

**Recursion:** Introduction, Problem Solving with Recursion, Examples of Recursive Algorithms

## UNIT - IV

**Design with Classes:** Classes and Objects, Classes and Functions, Classes and Methods, Working with Instances, Inheritance and Polymorphism.

**Object-Oriented Programming:** Procedural and Object Oriented Programming, Classes, techniques for Designing Classes

## UNIT - V

**Graphical User Interfaces:** Behavior of terminal based programs and GUI-based programs, Coding simple GUI-based programs, other useful GUI resources. GUI Programming: Graphical User Interfaces, Using the tkinter Module, Display text with Label Widgets, Organizing Widgets with Frames, Button Widgets and Info Dialog Boxes, Getting Input with Entry Widget, Using Labels as Output Fields, Radio Buttons, Check Buttons.

**Simple Graphics and Image Processing:** Overview of Turtle Graphics, Two dimensional Shapes, Colors and RBG System, Image Processing.

# GATE:-

**NA**

1. **LIST OF EXPERT DETAILS:-**

## INTERNATIONAL

* 1. Mr. Guido Van Rossum

San Francisco,California,United States.

Creator of the [Python programming language](https://en.wikipedia.org/wiki/Python_(programming_language))

<https://gvanrossum.github.io/>

Email: guido@python.org

* 1. Cen Li, Professor

Middle Tennessee State University

Email: [cen.li@mtsu.edu](mailto:cen.li@mtsu.edu)

PhNo: (615)904-8168

* 1. Allen B. Doweny

Currently working as a staff Scientist at DriyenData.

Farmer Professor of Computer science at the Frankin W.Olin College of Enginnering

Website: [https://allendowney.com](https://allendowney.com/)

## NATIONAL

1. Mr. Pankaj Jalote. Ph.D. Distinguished Professor(CSE),IIIT-Delhi. Email: [jalote@iiitd.ac.in](mailto:jalote@iiitd.ac.in)

PhNo: 001-26907499

2. Mr. Amey Karkare Ph.D. Professor(CSE),IIT-Kanpur.karkare@cse.iitk.ac.in, +91 512 259 7520

3. Mr. Vikram Goyal Ph.D. Professor(CSE),IIIT-Delhi [vikram@iiitd.ac.in](mailto:vikram@iiitd.ac.in), 011-26907474

## REGIONAL

1. Dr. Raja Shekar, Professor,CSE VNRVJIET Hyderabad, Email: [rajasekarm9@gmail.com](mailto:rajasekarm9@gmail.com). 9849296429
2. Dr. Prashanth Rao, Professor, IT Anurag University,Hyderabad,

Email:prashanthrao@cvsr.ac.in,9490232922

## JOURNALS: -

* 1. **TITLE:** An overview of Python Pgrogramming.

**AUTHORS:** Swapnil Raj

**LINK:** [**https://www.jetir.org/view?paper=JETIRFH06168**](https://www.jetir.org/view?paper=JETIRFH06168)

* 1. **TITLE:** Python – The Fastest Growing Programming Language

**AUTHORS:** K. R. Srinath

**LINK**: [https://www.irjet.net/archives/v4/i12/irjet -v4i1266.pdf](https://www.irjet.net/archives/v4/i12/irjet%20-v4i1266.pdf)

* 1. **TITLE:** Prediction Of Credit Card Approal

**AUTHORS:** Harsha Vardan Peela, Tanuj Gupta, Nishit Rathod

**LINK:** [International Journal of Soft Computing and Engineering - Regular Issue](https://www.sciencegate.app/source/1134558688) ◽

[10.35940/ijsce.b3535.0111222](https://www.sciencegate.app/app/redirect#aHR0cHM6Ly9keC5kb2kub3JnLzEwLjM1OTQwL2lqc2NlLmIzNTM1LjAxMTEyMjI=)

* 1. **TITLE:** PROGRAMMING LANGUAGE PYTHON: A REVIEW

**AUTHORS**: Preetee K. Karmore, Gaurishankar L. Girhe

**LINK:** [http://ijariie.com/AdminUploadPdf/PROGRAMMING\_LANGUAGE\_PYTHON A\_REVIE](http://ijariie.com/AdminUploadPdf/PROGRAMMING_LANGUAGE_PYTHON__A_REVIEW_ijariie11892.pdf) [W\_ijariie11892.pdf](http://ijariie.com/AdminUploadPdf/PROGRAMMING_LANGUAGE_PYTHON__A_REVIEW_ijariie11892.pdf)

## SUBJECT -LESSON PLAN

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.NO** | **Topic** | **Sub-Topic** | **NO. OF**  **LECTURES REQUIRED** | **Suggested Books** | **Teaching Methods** |
| **UNIT – I** | | | | | |
| **1** | **Python Operators, Data types & Variables, Loops, Strings** | Introduction to Python | **L1** | **T1,R1** | **M1** |
| **2** | Installing Python. How a Program Works, Using Python | **L2-L3** | **T1** | **M1** |
| **3** | Program Development Cycle, Input, Processing, and Output,  Displaying Output with the Print Function | **L4** | **T1** | **M1** |
| **4** | Comments, Variables, Reading Input from the Keyboard, Performing Calculations, | **L5-L6** | **T2,R1** | **M1** |
| **5** | Operators. Type conversions, Expressions | **L7** | **T1** | **M1** |
| **6** | More about Data Output. Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, | **L8** | **T2** | **M1** |
| **7** |  | Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables. | **L9** | **T1** | **M1** |
| **8** | Repetition Structures: Introduction, while loop, for loop | **L10-L11** | **T1,R1** | **M1** |
| **9** | Calculating a Running Total, Input Validation Loops, Nested Loops. | **L12** | **T1,R2** | **M1** |
| **10** | Data types and Expressions: Strings, Assignment and Comments, Numeric Data Types and Character Sets | **L13** | **T1,R2** | **M4** |
| **11** | Expressions, Functions and Modules. | **L14** | **T1,R2** | **M1** |
| **UNIT – II** | | | | | |
| **12** | **Files, Exceptions & Modules** | Control Statements: Definite Iteration, | **L15** | **T1,R3** | **M1** |
| **13** | Formatting Text for Output | **L16** | **T1** | **M4** |
| **12** | Selection, Conditional Iteration.  File and Exceptions | **L17** | **T1** | **M5** |
| **13** | Introduction to File Input and Output, | **L18** | **T1** | **M1** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **14** |  | Using Loops to Process Files, | **L19** | **T1,R3** | **M5** |
| **15** | Processing Records, Exceptions.  Functions: | **L20** | **T1,R3** | **M4** |
| **16** | Introduction, Defining and Calling a Void Function | **L21** | **T1,R3** | **M5** |
| **17** | Designing a Program to Use  Functions, | **L22** | **T1** | **M5** |
| **18** | Local Variables, Passing Arguments to Functions | **L23** | **T1,R2** | **M4** |
| **19** | Global Variables and Global Constants, | **L24** | **T1** | **M4** |
| **20** | Value-Returning Functions- Generating Random Numbers, | **L25** | **T2** | **M5** |
| **21** | The math Module, Storing Functions in Modules. | **L26,L27** | **T1** | **M4** |
| **UNIT-III** | | | | | |
| **22** | **String and Text Files** | Strings and Text Files: Accessing Characters and Substrings in a String, Strings and Number System | **L28** | **T1** | **M5** |
| **23** | , String Methods, Basic String Operations, String Slicing, | **L29** | **T1** | **M5** |
| **24** | Testing, Searching, and Manipulating Strings. Text Files, | **L30** | **T1,R2** | **M4** |
| **25** | Data Encryption, Lists, Introduction to Lists, List slicing | **L31** | **T1** | **M4** |
| **26** | Finding Items in Lists with the in Operator, | **L32** | **T1** | **M4 &M5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | List Methods and Useful Built-in Functions, Copying Lists, Processing Lists, | **L33** | **T1,R2** | **M5** |
|  | , Two-Dimensional Lists, Tuples Sequences, Tuples | **L34** | **T1,R2** | **M4** |
|  | Dictionaries and Sets: Dictionaries, Sets, Serializing Objects. | **L35** | **T2,R2** | **M4** |
|  |  | Recursion: Introduction, Problem Solving with Recursion, Examples of Recursive Algorithms | **L36** | **T1,R1** | **M5** |
| **UNIT-IV** | | | | | |
| **27** | **Design with Classes** | Design with Classes: Classes and Objects, Classes and Functions | **L37** | **T1,R1** | **M5** |
| **28** | Classes and Methods, Working with Instances, Inheritance and Polymorphism | **L38** | **T1,R1** | **M5** |
| **29** | .Object-Oriented Programming: Procedural and ObjectOriented Programming | **L39** | **T2,R1** | **M4** |
| **30** | Classes, techniques for Designing Classes | **L40,L41** | **T1** | **M4** |
| **UNIT –V** | | | | | |
| **34** |  | Graphical User Interfaces: Behavior of terminal based programs and GUI-based  programs, | **L42** | **T1** | **M3** |
| **35** | Coding simple GUI-based programs, other useful GUI resources. | **L43** | **T1,R2** | **M4** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **36** | **GUI, GUI**  **Programming, Simple Graphics and Image Processing** | GUI Programming: Graphical User Interfaces, Using the tkinter Module Display text with Label Widgets | **L44-L47** | **T1** | **M4** |
|  | , Organizing Widgets with Frames, Button Widgets and Info Dialog Boxes | **L48-L50** | **T1** | **M4** |
|  | Getting Input with Entry Widget, Using Labels as Output Fields, Radio Buttons, Check Buttons | **L51-L52** | **T2,R1** | **M5** |
|  | Simple Graphics and Image Processing: Overview of Turtle Graphics, Two dimensional Shapes, | **L53-L54** | **T1,R1** | **M4** |
| **37** | Colors and RBG System, Image Processing. | **L55** | **T1** | **M5** |

**Methods of Teaching**

|  |  |
| --- | --- |
| **M1 : Lecture Method** | **M6 : Tutorial** |
| **M2 : Demo Method** | **M7 : Assignment** |
| **M3 : Guest Lecture** | **M8 : Industry Visit** |
| **M4 : Presentation /PPT** | **M9 : Project Based** |
| **M5 : Lab/Practical /Activity** | **M10 : Charts / OHP** |

## Suggested Books (Prescribed and References) Prescribed Book

* 1. Kenneth A. Lambert, The Fundamentals of Python:First Programs,2011,Cengage Learning.
  2. Think Python First Edition,by Allen B. Downey, Orielly Publishing.

## Reference Books

1. Introduction to Computation and Programming Using Python. Jhon V. Guttag, The MIT Press.
2. James Payne, Beginning Python using Python 2.6 and Python 3, Wrox publishing.
3. Paul Gries, Practical Programming: An Introduction to Computer Science using Python 3, The Pragmatic Bookshelf,2nd edition(4 oct,2013)
4. Charles Dierach, Introduction to Computer Science using Python.

## Websites for self- learning Resources like

1. <https://www.python.org/>
2. <https://nptel.ac.in/courses/106106182>
3. <https://youtu.be/9kNDT-0yAEM>
4. <https://www.programiz.com/python-programming>
5. <https://www.geeksforgeeks.org/python-programming-examples/>
6. <https://www.w3schools.com/python/default.asp>
7. <https://www.tutorialspoint.com/python/index.htm>
8. <https://www.javatpoint.com/python-tutorial>

## OLD QUESTIN PAPER

## python qp.jpg

## PART-A

1. a) State any four applications where python is more popular.
2. List out the main differences between lists and tuples.
3. What are the uses of file object?
4. Give a brief description of several built in attributes related to File objects.
5. Summarize the purpose of pipe and dot symbols used for pattern matching
6. Explain the basic functionality of match() function.
7. What is the need of Tkinter module in python?
8. How to Label widget in Python?
9. State the need of persistent storage.
10. Discuss the SQL commands/statements used for creating, using and dropping a database.

## PART-B

1. a) How to declare and call functions in python programs? Illustrate with an example script.

b) List and explain few most commonly used built - in types in python.

1. Summarize various operators, built- in functions and standard library modules that deals with python numeric types.
2. Explain the following file built- in functions and methods with clear syntax, description and illustration:

a) open() b) file() c) seek() d) tell() e) read()

5.a) How does try-except statement work? Demonstrate with example python code.

* 1. Illustrate the concept of importing module attributes in python scripts.

1. Examine how python supports regular expressions through there module with brief introduction and various built - in methods related to it.
2. a) What is the motivation behind parallelism and state how python achieves parallelism?

## JNTUH/Model Papers



PREVIOUS QP.zip

## Two case study presentations with Project/Product/Model/ Prototypes/ Industrial applications.

## Python – Basic Chatbot using Python:

## Many real-world applications require basic interaction through conversation-like interfaces, such as customer support bots, personal assistants, and informational guides. A basic chatbot is an excellent entry-level AI tool that demonstrates how Python can handle language-like inputs, make decisions, and return appropriate responses.

## Tools & Technologies Used:

## Language: Python 3.x

## Libraries: None (basic version), optionally nltk for NLP enhancements

## Environment: Jupyter Notebook / VS Code / IDLE

## Python: Handwritten Digit Recognition:

In real-world scenarios like postal services, banking (cheque processing), or form scanning, there’s a need to **automatically identify handwritten digits**. Manual digit entry is time-consuming and error-prone. This project demonstrates how Python and deep learning can be used to automate digit recognition reliably.

**Tools & Technologies Used:**

* **Programming Language:** Python

**Libraries:**

* tensorflow or keras – model building
  + numpy – numerical operations
  + matplotlib – visualization
  + sklearn – metrics and preprocessing

## ASSIGNMENT QUESTIONS

**ASSIGNMENT –I**

* 1. a. Explain Python objects characteristics.[CO1]

b. Summarize the primitive data types of the Python with example.[ CO1]

* 1. a. Compare mutable and immutable data types.[ CO1]

b. Contrast String and List data structures in Python.[ CO1]

* 1. a. Define the Dictionary data structure in Python.[ CO1]

b. Write Python File Modes in Python.[ CO2]

* 1. a. Explain the usage of Command-Line Arguments in Python with an example.[ CO2]

b. Write about the errors and exceptions in Python. Give suitable examples.[ CO2]

* 1. a. Explain the standard exceptions with examples.[ CO2]

b. Write about modules and import in Python with examples.[ CO2]

## ASSIGNMENT –II

1. What are regular expressions? [ CO3]
2. How to find whether an email id entered by useris valid or not using Python ‘re’ module. [ CO3]
3. Differentiate tupel and list. [ CO3]
4. Explain Tinker module. [ CO5]
5. List web address components and explain them. [ CO5]

## INNOVATIVE ASSIGNMENT QUESTIONS

1. How can you generate random numbers in Python ?
2. What are negative indexes and why are they used?
3. What advantages do NumPy arrays offer over (nested) Python lists?
4. Describes anonymous functions examples?
5. Write a brief notes on PIP, Explain installing package via PIP.

## UNIT WISE QUESTIONS

**UNIT-1**

1. List the standard type operators in Python with examples. [ CO1]

a) Give a note on each of the following constructs in Python language.

(i)quotes (single, double and triple) (ii) multiline statements (iii) indentation [ CO1]

b) How Python is different from C++.[ CO1]

1. a) Narrate the other built in data types of Python[ CO1]

b) List the unsupported types in Python along with explanation. [ CO2]

3.a)Explain Python bitwise operators with example. [ CO1]

b) Compare and contrast the List and Tuple. [ CO1]

4. What is Python? Explain in detail. [ CO1]

1. Explain about the type of operators used in Python? [ CO1]
2. How to declare and call functions in python programs? Illustrate with an example script. [ CO1]
3. List and explain few most commonly used built - in types in python. [ CO1]
4. State any four applications where python is more popular. [ CO1]
5. .List out the main differences between lists and tuples. [ CO1]

## UNIT-II

1. What is the need of Exception in python. Explain ‘Now’ exception. [ CO2]
2. Explain the importing module attribute with suitable examples. [ CO2]
3. What are the two ways of importing a module? Which one is more beneficial? Explain[ CO]
4. a)Briefly discuss about Python packages. [ CO2]

b) Explain about handling an exception. [ CO2]

1. a) How to handle an exception using try except block? Explain with the help of a program[ CO2]

b) Why Exceptions is needed? Discuss with detailed examples. [ CO2]

6. Demonstrate usage of void functions in Python? [ CO2]

7. Explain in detail about functions in Python? [ CO2]

8. Give a short note on Python built in functions? [ CO2]

## UNIT-III

## How to access substring in a string illustrate with program.[ CO2]

1. Differentiate List and tuple and dictonaries.[ CO3]
2. a) What are the threads in Python? [ CO3]

b) Differentiate thread and threading classes. [ CO3]

1. a) List special symbols and characters while forming regular expressions. [ CO3]

b) Explain various String pattern matching functions in Python. [ CO3]

1. a) What is List slicing? Discuss about various methods in lists[CO3].

b) Explain the methods of list. [ CO3]

6. Give a short note on Regular Expressions (Res)? [ CO3]

7. Explain methods of dictionaries. [ CO3]

8. Define built in functions in python? [ CO3]

## UNIT-IV

**1.** a) Explain about Classes and Objects in python. [ CO4]

b) Write a program to implement Classes and objects.[ CO4]

**2.** Describe about the functions in python programming. [ CO4]

3. a) Explain about classes and methods in python[ CO4]

b) Explain working instances of classes. [ CO4]

4. a) Explain a procedure to create a class in python. [ CO4]

b) Explain techniques for designing classes. [ CO4]

5. a) Explain inheritance in Python. [ CO4]

b) Explain polymorphism . [ CO4]

## UNIT-V

1. a) Explain Graphical user interface in python. [ CO5]

b) Database Connection Objects in Python. [ CO5]

2. a) Discuss database adapter with examples. [ CO5]

b) Explain the widgets with frames and button widgets. [ CO5]

3. a) Narrate the Tinker Module attributes with description. [ CO3]

b) Write about Image processing. [ CO5]

4. a) Explain the Database connectivity procedure with an example. [ CO5]

b) Explain the Turtle Graphics. [ CO5]

5. Write a Python program that creates a GUI with a textbox, Ok button and Quit button. On clicking Ok, the text entered in textbox is to be printed in Python shell; on clicking Quit, the program should terminate. [ CO5]

## (13) List of topics for students Seminars

1. Oops concepts
2. Math Module
3. Dictionaries and sets
4. Tkinter Module

(14) STEP/Course material in soft copy

[Python Notes. zip](file:///C:\Users\kala\Downloads\Python%20Notes.zip)

Step Material For python programming.rar

## (15) Expert Lectures with topics &Schedules (if any)

**TOPICS:**

1. GUI based programs
2. Graphics & Image processing.

## SCHEDULES:

Planning to conduct Guest Lectures