



BY CHANDRA SEKHAR

TREENETRA EDUCATION

PYTHON AUTOMATION TESTING SYLLABUS

TREENETRA EDUCATION, #15, 1ST
CROSS, MUNNEKOLLAL MAIN RD,
NEAR SGR DENTAL COLLEGE ROAD,
MARATHAHALLI, BENGALURU,
KARNATAKA 560037

(HR ISHWARI)
+91-7058-475-504 (BATCH ADMISSION)
(HR OFFICE)
+91-8296-789-108 (INQUIRY) (INTERVIEW)

TECHNOLOGICAL STACK



1.PROGRAMING LANG

PYTHON

2.FRAMEWORK

PYTEST, ROBOT

3. UI TESTING

SELENIUM,PLAYWRIGHT

4.DEVOPS TOOLS

DOCKER,GIT,JENKINS(CI/CD),GR
ROVY SCRIPT

5.DATABASE(TESTING)

SQL/MYSQL
PYTHON SQL ALCHEMY
DATABASE TESTING

6.CLOUD TECHNOLOGY

AWS:- EC2,S3,,LAMBDA,CLOUD
WATCH

7.API TESTING

POSTMAN
REQUEST MODULE
JSON

8.FRONT END

HTML,CSS,JAVA SCRIPT

9.OS

UNIX
EC2
AMAZON LINUX

10.MANUAL TESTING

ALL TESTING CONCEPTS
SDLC/STLC
BUG LIFE CYCLE

11.PROJECT

REAL TIME(REPLICA) PROJECT
TELECOMM
AUTOMATIVE

12.MANDATORY SKILLS

JIRA,SLACK,TEAMS
CONFLUENCE
TASK MANAGEMENT
REAL TIME INDUSTRY
SENARIO[MEETING,CLIENT
INTERACTION]

SR NO	MODULES NAME	COURSE CONTENT
1	INTRODUCTION	<ul style="list-style-type: none">• What is PYTHON?• WHY PYTHON?• History of Python• Features of Python.• Why Python is General Language?/HLL?• Limitations of Python
2.	Python software Installation and Introduction	<ul style="list-style-type: none">• Python Distributions, Anaconda Navigator• Download & Python Installation Process in Windows, Unix, Linux and Mac• Online Python IDLE• Python Real-time IDEs like Spyder, Jupyter Note Book, PyCharm, Different Modes of Python
3	Language Initials	<ul style="list-style-type: none">• Python Identifiers(Rules and Regulations)• Reserved words and Keywords• Basic Data types in Python(Sequential, non-sequential, ordered, non-ordered)
4	Data Type and Data Structure	<ul style="list-style-type: none">• String(all methods ,slicing and string operation)• List, Tuple, Dictionary, Range(All Operation and Data structure work)• Set,frozenset,bytes, bytearray, None• Type Casting
5	Operators	<ul style="list-style-type: none">• Arithmetic Operators• Comparison Operators• Python Assignment Operators• Logical Operators• Membership Operators• Identity Operators• Ternary Operator• Operator precedence• Difference between “is” vs “==”

SR NO	MODULES NAME	COURSE CONTENT
6	Flow Control Control Statement	<ul style="list-style-type: none"> • Conditional control statements , If , If-else ,If-elif-else ladder ,Nested-if-else • Loop control statement, for ,while ,Nested for loops ,Branching statements • Break,Continue,Pass • Case studies- Pattern making (letters and Diagrams)
7	Functions	<p>What is Function?--Advantages of functions., Syntax and Writing function,Calling or Invoking function</p> <p>Classification of Functions- On the basis of argument and return value</p> <p>No arguments and No return values,With arguments and No return values,No arguments and with return values</p> <p>Positional argument type functions,Default argument functions.variable length arguments function, keyword argumentsfunction(*arg),Variable length keyword arguments functions(**kwargs)</p> <p>zip() in Python</p> <p>What is variables .Global and Local Variable</p> <p>Anonymous functions(Lambda, filter, map, reduce)Code Optimization(Comprehension(list,dict)</p> <p>Function Aliasing</p> <p>DecoratorGenerator</p>
8	Python Module and Package	<ul style="list-style-type: none"> • What is Module,Type Of Module- Pre Define, User Define,Function/Class Based Module,How to import modules • Module Alias,Math,random,os,sys,time,datetime • Organizing python project into packages,Types of packages – pre defined, user defined.,Package v/s Folder • .py file,Importing package,pip
9	File Handling	<ul style="list-style-type: none"> • What is Function?--Advantages of functions., Syntax and Writing function,Calling or Invoking function • Classification of Functions- On the basis of argument and return value • No arguments and No return values,With arguments and No return values,No arguments and with return values • Positional argument type functions,Default argument functions.variable length arguments function, keyword argumentsfunction(*arg),Variable length keyword arguments functions(**kwargs) • zip() in Python • What is variables .Global and Local Variable • Anonymous functions(Lambda, filter, map, reduce)Code Optimization(Comprehension(list,dict) • Function Aliasing • Decorator,Generator

SR NO	MODULES NAME	COURSE CONTENT
10	OOPs	<ul style="list-style-type: none"> • Procedural v/s Object oriented programming, Classes and Objects, How to define class in python • Types of variables – instance variables, class variables. Types of methods – instance methods, class method, static method, Object initialization, 'self' reference variable, 'cls' reference variable, Property () object theory • Principles of OOP – Encapsulation, Abstraction (Data Hiding) • Creating object properties using setattr, getattr functions, Inner classes, Class re-usability • Inheritance – single, multi-level, multiple, hierarchical and hybrid inheritance and Diamond inheritance, Method resolution order (MRO) • super () • Constructors in inheritance • Object class • Duck typing interview question • What is polymorphism • Runtime polymorphism • Overriding • i) Method overriding • ii) Constructor overriding • Method overriding in Multiple inheritance and Hybrid Inheritance • Overloading • i) Method Overloading • ii) Constructor Overloading • iii) Operator Overloading
11	Exception Handling	<ul style="list-style-type: none"> • What is Exception?, Why exception handling?, Syntax error v/s Runtime error • Exception codes – AttributeError, ValueError, IndexError, TypeError... • Handling exception – try except block, Try with multi except, Handling multiple exceptions with single except block • Finally block, Try-except-finally, Raise keyword • Case study of finally block, Custom exceptions / User defined exceptions • Need to Custom exceptions
12	OS module	<ul style="list-style-type: none"> • What is Module, Type Of Module- Pre Define, User Define, Function/Class Based Module, How to import modules • Module Alias, Math, random, os, sys, time, datetime • Organizing python project into packages, Types of packages – pre defined, user defined., Package v/s Folder • .py file, Importing package, pip

SR NO	MODULES NAME	COURSE CONTENT
1	PYTEST	<ul style="list-style-type: none">• Pytest - Introduction• Pytest - Environment Setup• Identifying Test files and Functions• Pytest - Starting With Basic Test• Pytest - File Execution• Execute a Subset of Test Suite• Substring Matching of Test Names• Pytest - Grouping the Tests• Pytest - Fixtures• Pytest - Conftest.py• Pytest - Parameterizing Tests• Pytest - Xfail/Skip Tests• Stop Test Suite after N Test Failures• Pytest - Run Tests in Parallel• Test Execution Results in XML• Pytest - Summary• Pytest - Conclusion

ROBOT FRAMEWORK

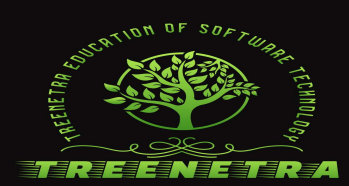


SR NO	MODULES NAME	COURSE CONTENT
1	ROBOT FW	<ul style="list-style-type: none">• Robot Framework - Home• Robot Framework - Overview• Environment Setup• Unix and Linux Installation• Introduction to Ride• First Test Case Using Ride• Writing and Executing Test Cases• Keyword and Data Driven Test Cases• Working With Browsers Using Selenium Library• Working With Textbox• Working With Radio Button• Working With Checkbox• Working With Dropdown• Working With Keywords• Working With Variables• Working With Command Line• Working With Setup And Teardown• Working with Built-In Library• Working With External Database libraries• Testing Login Page Using Robot Framework

SR NO	MODULES NAME	COURSE CONTENT
1	SELENIUM	<ul style="list-style-type: none">• Downloading WebDriver Jars and configuring in Pycharm• The architecture of Selenium Webdriver• Browser versions with WebDriver• Drivers for Firefox, IE, Chrome, Edge <p>HTML,CSS Knowledge</p> <p>Locators & Xpath :- Inspect elements from the browser , Relative & Absolute Xpath</p> <p><u>Browser Commands</u></p> <ul style="list-style-type: none">• Web Element Commands• Find Elements Commands• Check Box & Radio buttons• Drop Down & Multiple Selects• Web Table Handling• Alerts and windows• Frames• Screenshot• Scroll up and Down• Window handels• File Upload• Web Table automate <p>Synchronization :- Implicit Wait,Explicit Wait</p> <p>Action Class:-Drag & Drop,Right Click & Double Click,Mouse hover Action</p>

MODULES NAME	COURSE CONTENT	
API Request Module	<p>Introduction to REST API</p> <ul style="list-style-type: none">❑ What is WebServices?❑ Why WebServices so Popular?❑ Overview of SOAP WebServices and REST WebServices❑ What is REST API?❑ How is different from SOAP WebServices?❑ Base URL and REST Resources❑ Understanding of GET, POST, PUT, DELETE <p>Getting Started with REST API Testing</p> <ul style="list-style-type: none">❑ Understanding REST API Testing Part❑ Manual Testing on REST API using POSTMAN❑ How to Trigger New Request and Validate Response using REST API	<p>REST API Automation</p> <p>>>REQUEST MODULE</p> <ul style="list-style-type: none">❑ Overview Automation on REST Applications❑ Benefit of Automation Testing for Applications❑ Setting up Project Tools for Automation Testing❑ Understanding How to setup for Automation Testing❑ Mock Sample Testing❑ How to validate Responses <p>Depth Automating REST API</p> <ul style="list-style-type: none">❑ Automating POST request with Payload❑ Applying Advanced validation on Responses❑ Passing Previous values into subsequent Request❑ Business case for Error Responses

DEVOPS TOOLS



SR NO	MODULES NAME	COURSE CONTENT			
1	DOCKER	<ul style="list-style-type: none"> Introducing Docker, Comparing VM and Docker Docker –An Architectural overview, The Docker Hub A brief Introduction Preparing docker-machine – Installation and configuration, Start containerizing Play with docker images Dockerfile Directives, USER and RUN, RUN Order of Execution ENV, CMD vs. RUN, ENTRYPOINT, EXPOSE Docker Container Volume Management – An introduction, Docker Networking concepts List and Inspect, Create and Remove, Assign to Containers Exercise: Creating a Custom Image from a Dockerfile, Exercise: Managing Containers Exercise: Adding External Content to Containers 			
2	git	<p><u>Introduction to Git</u></p> <p>Version Control/Revision Control system, The types of VCS, The benefits of using VCS What is GIT? The difference between GIT and other VCS, Where to use Git and where not to use it Web Scale architecture</p>	<p><u>Getting Started With Git And Its Architecture</u></p> <p>Install Git on Windows/Mac/Linux/Unix Understand Git file life cycle Create Git repository - Local repository and configure it to GitHub Create a repository on GitHub and clone it, Understand basic Git commands, Git command shortcuts. Revisit the Git file lifecycle with Git terminology</p>	<p><u>Branching in Git</u></p> <p>How Git internally manages branches How to switch between branches and different commits Two way merge and three way merge Merging Strategies Merging and rebasing (using Source tree) Git tags Difference between branching and tagging</p>	<p><u>Git Methodology</u></p> <p>What is GIT workflow? Advantages of workflow Different types of workflows in Git Centralized Workflow Feature branch workflow Gitflow Workflow Forking Workflow How to use git in real time open source projects</p>
3	Jenkins	<ul style="list-style-type: none"> Introducing Continuous Integration and Jenkins Agile Development Continuous Integration History of Jenkins State of the Jenkins community 	<ul style="list-style-type: none"> Installing and Running Jenkins Running Jenkins from the jar file Installing Jenkins in a servlet container 	<ul style="list-style-type: none"> A Jenkins Job Creating a job Configure a job Run a job manually Run a job when source code is checked into version control Run a job on a regular 	<ul style="list-style-type: none"> schedule Change reporting Code coverage Static Analysis Performance reporting Style checking

SR NO	MODULES NAME	COURSE CONTENT			
1	EC2	<ul style="list-style-type: none">• Virtualization• Amazon Elastic Compute Cloud (EC2) and Its Benefits• Amazon Machine Image (AMI)• Security Groups in AWS• Authentication through Key-pair	<ul style="list-style-type: none">• Public IP vs. Elastic IP• Instance Store• Elastic Block Store (EBS), Its Features and Volume Types		
2	Database Services and Analytics	<ul style="list-style-type: none">• Amazon RDS and its benefits• Read Replica RDS• IAM Authentication• DynamoDB			
3	Networking and Monitoring Services	<ul style="list-style-type: none">• AWS CloudWatch• AWS CloudTrail• AWS Config			

MANUL TESTING / LINUX



SR NO	MODULES NAME	COURSE CONTENT			
1	MANUL TESTING	<ul style="list-style-type: none"> To understand what is testing? To understand Software development model. To Understand Architectures of software development. To learn the features of Software development models. To learn major concepts of the testing methodologies. To know different approaches to Testing. To understand of the types of testing. To plan and create test plan To execute the test plan. To create and manage test cases and defect profiles To build strategies to track testing processes in the bug tracking systems. To do document of the test report in the testing enclosure document. 			
2	LINUX	<ul style="list-style-type: none"> Understanding Linux Concepts Download, Install and Configure System Access and File System Linux Fundamentals Linux System Administration Networking, Servers and System Updates	Commands Syntax File Permissions (chmod) File Ownership (chown, chgrp) Getting Help (man, what is etc.) TAB completion and up arrow keys Adding text to file Standard output to a file (tee command)	File Maintenance Commands File Display Commands Filters / Text Processing Commands (cut, sort, grep, awk, uniq, wc)	Linux vs. Windows Commands Quiz, Homework and Handouts Compare Files (diff, cmp) Compress and un-compress files/directories (tar, gzip, gunzip) Truncate file size (truncate) Combining and Splitting Files (cat and split)
3	MANDATORY SKILS	<ul style="list-style-type: none"> JIRA SLACK TEAMS CONFLUENCE 	<ul style="list-style-type: none"> TASK MANAGEMENT REAL TIME INDUSTRY SENARIO[MEETING CLIENT INTERACTION] 		

PROJECT-TELECOM



Introduction to OSS and BSS	OSS Components	BSS Components	Integration of OSS/BSS	Service Fulfillment and Assurance
<ul style="list-style-type: none">• Overview of OSS/BSS• Evolution and History• Importance in Telecommunication Networks• Key Differences between OSS and BSS	<ul style="list-style-type: none">• Network Management Systems (NMS)• Service Management• tFault Management• Configuration Management• Performance Management• Security Management	<ul style="list-style-type: none">• Customer Relationship Management (CRM)• Order Management• tBilling and Revenue Management• Product Lifecycle Management• Customer Self-Service	<ul style="list-style-type: none">• Data Flow between OSS and BSS• Process Automation and Workflow Management• Integration Technologies (APIs, ESB)Interoperability Challenges	<ul style="list-style-type: none">• Service Fulfillment Processes• Service Assurance Processes• Activation and Provisioning• Monitoring and Reporting
Network and Service Management	Revenue Management and Billing	Customer Relationship Management (CRM)	Order and Inventory Management	Next-Generation OSS/BSS
<ul style="list-style-type: none">• End-to-End Service Management• Network Planning and Optimization• Real-Time Network Analytics• Service Level Agreements (SLAs)	<ul style="list-style-type: none">• Billing Systems and Processes• Charging Mechanisms (Prepaid, Postpaid)• Revenue Assurance• Fraud Management	<ul style="list-style-type: none">• Customer Data Management• Sales and Marketing Automation• Customer Support and Service• Customer Experience Management (CEM)	<ul style="list-style-type: none">• Order Handling and Processing• Inventory Systems• Supply Chain Management• Product Catalog Management	<ul style="list-style-type: none">• Virtualization and Cloud Computing• Software-Defined Networking (SDN)Network Function Virtualization (NFV)IoT and OSS/BSS Integration• Artificial Intelligence and Machine Learning in OSS/BSS

PROJECT-AUTOMATIVE



Introduction to Automotive IT	Vehicle Electronics and Embedded Systems	Communication Protocols in Vehicles	Connected Vehicles and Telematics	Advanced Driver Assistance Systems (ADAS)
<ul style="list-style-type: none">• Overview of Automotive IT• Evolution of Automotive Electronics and Software• Importance of IT in the Automotive Industry• Automotive IT Market Trends and Future Outlook	<ul style="list-style-type: none">• Basics of Automotive Electronics• Microcontrollers and Microprocessors in Vehicles• Embedded Systems Design and Development• Real-Time Operating Systems (RTOS)Automotive Sensors and Actuators	<ul style="list-style-type: none">• In-Vehicle Networking (CAN, LIN, Flex Ray, Ethernet)Vehicle-to-Everything (V2X) Communication• Wireless Communication Technologies (Bluetooth, Wi-Fi, 5G)Diagnostic Communication Protocols (UDS, OBD-II)Data Communication and Integration	<ul style="list-style-type: none">• Introduction to Connected Vehicles• Telematics and Infotainment Systems• Over-the-Air (OTA) Updates• Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) Communication• Data Privacy and Management in Connected Vehicles	<ul style="list-style-type: none">• Introduction to ADAS Technologies• Sensor Technologies (Radar, LiDAR, Cameras)Algorithms for ADAS (Computer Vision, Sensor Fusion)Safety and Reliability in ADAS• Testing and Validation of ADAS

THANKS
TREENETRA EDUCATION
(WWW.TREENETRA.IN)
+91-7058-475-504
+91-8296-789-108