

Basic Python		
S.No	Topic	Subtopic
1	Introduction	1.1 Introduction to Python
		1.2 Origin of Python
		1.3 Key features in Python
		1.4 Areas of applications of Python
		1.5 Installation & Environment setup
		1.6 "Hello World" program
		1.7 Python programming syntaxes
2	Variables & Operators	2.1 Variable properties
		2.2 Scope of variable
		2.3 Operators in Python
		2.4 Operator precedence
		2.5 Basic I/O using <i>input()</i> and <i>print()</i>
3	Control-flow statement	3.1 Decision making using <i>if-elif-else</i>
		3.2 Iteration using <i>for</i> loop and <i>range()</i>
		3.3 <i>while</i> loop and <i>while-else</i>
		3.4 <i>break</i> , <i>continue</i> and <i>pass</i>
4	Numbers	4.1 integers and float
		4.2 complex numbers
		4.3 random numbers
		4.4 <i>math</i> library
5	Strings	5.1 String overview & properties
		5.2 String formatting
		5.3 String functions
6	Lists	6.1 List overview & properties
		6.2 List functions
		6.3 List comprehension
7	Tuples	7.1 Tuple overview & properties
		7.2 Tuple functions
8	Dictionary	8.1 Dictionary overview & properties
		8.2 Dictionary functions
		8.3 Dictionary comprehensions
9	Sets	9.1 Set overview & properties
		9.2 Set functions
10	Functions & Modules	10.1 Functions overview
		10.2 <i>lambda</i> function
		10.3 Recursive functions
		10.4 <i>map</i> , <i>filter</i> and <i>reduce</i>
		10.5 Modules and packages

Advance Python		
11	Exception Handling	11.1 Exceptions & Errors
		11.2 Handling exceptions using <i>try-except-else-finally</i>
		11.3 Causing exceptions using raise-assert
		11.4 Making your own exception object
		11.5 Python exception hierarchy
		11.5 Logging & Debugging
12	File I/O	12.1 Working with files open(), close(), closed, with, access modes
		12.2 Working with file cursor seek(), tell()
		12.3 Reading text files readable(), read(), readline(), readlines()
		12.4 Writing into text files writable(), write(), writelines()
		12.5 Working with CSV files reader(), writer(), writerow(), writerows()
13	XML & JSON parsing	13.1 Working with xml files
		13.2 etree parsing
		13.3 XML to CSV convertor
		13.4 Web scraping of <i>sitemap.xml</i>
		13.5 Working with JSON files
14	Database Connectivity	14.1 Understanding database i/o
		14.2 Working with sqlite3 database
		14.3 Working with MySQL database
		14.4 Working with MongoDB
15	Regular expression	15.1 Pattern matching
		15.2 Understanding meta characters
		15.3 re functions and flags
16	Standard libraries	16.1 Operating System Interface
		16.2 Command Line Arguments
		16.3 Error Output Redirection and Program Termination
		16.4 Dates and Times
		16.5 Data Compression
		16.6 Performance Measurement
		16.7 Output Formatting
17	Classes & objects	17.1 Object oriented terminologies
		17.2 Python Scopes and Namespaces
		17.3 Static and Instance Variables
		17.4 Inheritance and Overriding
		17.5 Overloading

		17.6 Data abstraction and hiding
		17.7 Iterators, Generators & Decorators
		17.8 Multi-threading and multi-processing
		17.9 Locks in thread
18	Socket Programming	18.1 Overview of socket programming
		18.2 Client-server architecture
		18.3 Message sharing using client-server
19	E-mail Automation	19.1 Working with SMTP object
		19.2 Email sending automation
20	GUI Programming	20.1 Using tkinter library for app GUI
		20.2 App 1: Twitter Analysis
		20.3 App 2: Django architecture
		20.4 App 3: Game
		20.5 App 4: recognition coding
Django		
1	Django overview	- Django is a framework - Frameworks vs Packages - What can it do? - What are the alternatives?
2	Getting Started with Django	- Sites and apps - Shared configuration - Minimal Django layout - Built-in flexibility
3	Using cookiecutter	- Why cookiecutter? - cookiecutter features - Creating a project - Creating an app - Customizing cookiecutter
4	Creating models with Django ORM	- Defining models - Related objects - SQL Migration - App configuration - Accessing models
5	Login for Nothing and Admin for Free	- Setting up the admin user - Running the admin site - Tweaking the admin interface - Changing the admin index page
6	Basic Views (AKA Controllers)	- What is a view - HttpResponse - URL route configuration

		• Shortcut: get_object_ar_0
7	Working with templates	• About templates • Django template syntax • Statsc files • Loading templates • The uld tag • Shortcut: template.renderD
8	Querying the Models	• QuerySets • Field lookups • Chaining filters • Slicing QuerySets • Related fields • Q objects
9	Enhancing Model	• Custom methods • Complex relationships • Overriding standard methods
10	Forms	• Forms overview • GET and POST • The Form class • Processing the form • Widgets • Validation • Forms in templates • Beyond the basics
11	Debugging	• Printing to the console • Django debug page • Reading Django logs • Django debug toolbar
12	Class-based views	• About generic views • Types of generic views • Default generic views • Class-based generic views • List and detail views
13	User Authentication	• Authentication vs Authorization • Configuring users • Permissions • Groups
14	Session Management	• Enabling sessions • Types of session backends • Session cookies • Accessing sessions from views

15	Migrations	• Managed vs unmanaged tables
		• Handling schema changes
		• Using makemigrations and migrate
		• Tweaking migrations
		• Migrating data
16	Advanced configuration	• Django settings architecture
		• Typical configuration
		• Using 12 factor config
		• Security considerations
17	Serving Data with a RESTful Interface	• The Django REST framework
		• Serialization
		• Requests and Responses
		• Function-based views
18	Static File Management	• Class-based views
		• Types of static files
		• Configuring access
		• Namespaces
		• Templates
19	Automated Testing	• Deploying from outside Django
		• Why create tests?
		• When to create tests
		• Using Django's test framework
		• Using the test client
		• Running tests
20	Using the Cache	• Checking code coverage
		• Types of caches
		• Setting up the cache
		• Per-site and per-view caching
		• Low-level API
21	Creating Reusable Apps	• Cache security
		• Paracaching
		• Choosing an appropriate name
		• Deploying
22	Documentation with Sphinx	• Using setup tools
		• Description of Sphinx
		• Restructured Text
		• Configuring Sphinx
		• Creating doc files
		• Using autoDoc
		• Generating docs



Winner Academy of Excellence

ADDITIONAL TOPIC:

"Invoke SOAP/REST web services over http/https with SSL for Django"

- (a) This is in preparation for their initiative towards integration with 3rd party facilities
(e.g. Payment Gateways, SMS gateways, weather services)
- (b) Can the Trainer also help include possible similar case scenarios and exercises, as relating to the sub-section (a) above?