

COMPLETE SYLLABUS

By K.S. Chandre

Technological Stack	
Roles	Python Developer ML Data Scientist Data Engineer Data Analyst
Programming Lang	Python [Basic Adv]
Web Framework	Django Flask FastAPI
Database	SQL MongoDB PostgreSQL MSSQL
DevOps Tools	GIT Docker Kubernetes Ansible Jenkins[CICD]
Cloud Technology	AWS: - S3, EC2, Step Function, lambda's glue, redshift, EMR, CloudWatch, RDS
	Azure: - Logic App Data Factory ML Studio Resource Group
Machine Learning	Pandas NumPy Seaborn Matplotlib Scikit Learn EDA Statistics
ML Algorithms	Linear Reg Logistic Reg Decision Tree KNN K-means Random Forest XGBoost
Big Data Technology	PySpark Hadoop
Visualization Tools	PowerBI Tableau
Deep Learning	Deep Learning and AI
API/Rest Framework	Django Rest Framework [API creation, consumption], Flask API
API Testing Tool	Postman
Front End	Html CSS PHP Java Script
Model Deployment	Nginx Gunicorn
OS	Unix EC2 Amazon Linux
Projects	Real Time Project - 3
Mandatory Basic Skills	VPN, Jira Task Management tools, Slack, Teams, Confluence, real time industry scenario [meeting, client interaction]

Python [Basic | Advanced]

Sr. No	Major module	Course Content
1	Python Introduction- Part 1	<ul style="list-style-type: none"> • Introduction class- (Glass breaking, Team building approach, Industry scenarios, Market conditions, Supply-demand opportunities) • Syllabus declaration and time management • What is Language?, Types of languages, Introduction to Translators, Compiler, Interpreter, Debugger etc
2	Python Introduction- Part 2	<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
3	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
4	Language Initials	Python Identifiers(Rules and Regulations) Basic Data types in Python(Sequencial, non-sequencial, ordered, non-ordered)
Basic Python		
5	List	What is list?
		Properties of List.
		Types of List.
		List support indexing(forward and Reverse)
		Ordered and unordered nature.
		Mutable and immutable nature.
		Operations/ Methods/ Functions of List- Part 1
		Operations/ Methods/ Functions of List- Part 2
6	Tuple	What is a Tuple?
		Properties of tuple.
		Types of tuple.
		Operations/ Methods/ Functions of tuple
7	Set	<ul style="list-style-type: none"> • What is set? • Different ways of creating set • Difference between list and set • Accessing elements of set • Python Set Methods • Python Set Operations • Union of sets
8	Dictionary	<ul style="list-style-type: none"> • What is dictionary? • Difference between list, set and dictionary • How to create a dictionary? • Accessing values of dictionary • Copying dictionary • Updating Dictionary • Reading keys from Dictionary • Reading values from Dictionary • Reading items from Dictionary • Delete Keys from the dictionary • Sorting the Dictionary • Python Dictionary Functions and methods

9	String	<ul style="list-style-type: none"> • What is String? Properties of string. • Processing elements using indexing • Processing elements using Iterators • Manipulation of String using Indexing and Slicing • String operators • Methods of String object
10	Type Casting	String Formatting and Type casting
11	Operators	<ul style="list-style-type: none"> • Arithmetic Operators • Comparison Operators • Python Assignment Operators • Logical Operators • Bitwise Operators • Shift operators • Membership Operators • Identity Operators • Ternary Operator • Operator precedence • Difference between “is” vs “==”
12	Control statements	<ul style="list-style-type: none"> • Conditional control statements • If • If-else • If-elif-else ladder • Nested-if-else • Loop control statements • for • while • Nested for loops • Branching statements • Break • Continue • Pass • Case studies- Pattern making (letters and Diagrams)
13	Functions	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
		With arguments and with return values
		No arguments and with return values
		Classification of Functions- On the basis of parameters or Arguments
		Positional argument type functions
		Default argument functions
		variable length arguments function
		Keyword arguments function(*arg)
		Variable length keyword arguments functions(**kwargs)
		zip() in Python
	Variables	What is variable?
		Global variable and local variable?
14	Anonymous functions	Lambda functions
		· map ()
		· filter ()
		· reduce ()
15	Code optimization	List comprehension, tuple ,dict
17	Function Aliasing	What is Function Alising?
		use of Function Alising.
18	Decorators	what is a decorator?

		Uses of a decorator?
		Syntax of writing a decorator?
		Closures – inner function outside
		Generators
		Iterators
19	Extras	Doubt solving session.
Advanced Python		
20	Object oriented programming	Procedural v/s Object oriented programming
		Principles of OOP – Encapsulation, Abstraction (Data Hiding)
		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
		Creating object properties using setattr, getattr functions\
		Inner classes
		Introduction
		Writing inner class
		Accessing class level members of inner class
		Accessing object level members of inner class
		Local inner classes
		Complex inner classes
		Encapsulation (Data Binding)
		Access modifiers – private (____), protected (____), public
		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
		Concrete Methods in Abstract Base Classes
		Difference between Abstraction & Encapsulation ---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
21	Python Modules	Importance of modular programming
		What is module
		Types of Modules – Pre defined; User defined.
		User defined modules creation
		Functions based modules

		Class based modules
		Connecting modules
		Import module
		From ... import
		Module alias / Renaming module
		Built In properties of module
		Math module, random module
22	Packages	Organizing python project into packages
		Types of packages – pre defined, user defined.
		Package v/s Folder
		.py file
		Importing package
		PIP
		Introduction to PIP
		Installing PIP
		Installing Python packages
		Un installing Python packages
23	Exception Handling & Types of Errors	What is Exception?
		Why exception handling?
		Syntax error v/s Runtime error
		Exception codes – AttributeError, ValueError, IndexError, TypeError...
		o Handling exception – try except block
		o Try with multi except
		o Handling multiple exceptions with single except block
		Finally block
		o Try-except-finally
		o Try with finally
		o Case study of finally block
		Raise keyword
		o Custom exceptions / User defined exceptions
		o Need to Custom exceptions
		Case studies
24	File &Directory handling	Introduction to files
		Opening file
		File modes
		Reading data from file
		Writing data into file
		Appending data into file
		Line count in File
		CSV module
		Creating CSV file
		Reading from CSV file
		Writing into CSV file
25	OS module	Shell script commands
		Various OS operations in Python
		Python file system shell methods
		Creating files and directories
		Removing files and directories
		Shutdown and Restart system
		Renaming files and directories
		Executing system commands

26	Multi-threading & Multi Processing	Introduction
		Multi-tasking v/s multi-threading
		Threading module
		Creating thread – inheriting Thread class, Using callable object
		Life cycle of thread
		Single threaded application
		Multi-threaded application
		Can we call run () directly?
		Need to start () method
		Sleep ()
		Join ()
27	Object Serialization pickle module	XML parsing
		JSON parsing
28	Python Logging	Logging Levels
		implement Logging
		Configure Log File in over writing Mode
		Timestamp in the Log Messages
		Python Program Exceptions to the Log File
		Requirement of Our Own Customized Logger
		Features of Customized Logger
29	Assertion in Python	· Types of assertion
		· Simple, Augmented type
		· Use of assertion in real time
30	Garbage collection	Introduction
		Importance of Manual garbage collection
		Self reference objects garbage collection
		‘gc’ module
		Collect() method
		Threshold function
		Case studies
31	Database Connection SQL	Introduction to DBMS applications
		File system v/s DBMS
		Communicating with MySQL
		Python – MySQL connector
		connector module
		connect () method
		Oracle Database
		Install cx_Oracle
		Cursor Object methods
		execute () method
		execute Many () method
		fetchone()
		fetchmany()
		fetchall()
		Static queries v/s Dynamic queries
		Transaction management
		Case studies
****End of Python Syllabus****		
Monkey patching? ---interview		

Data science with Python

Numpy

- Arrays
- Basic Operations in Numpy
- Indexing
- Array Processing

Pandas

- Series
- Data Frames
- Indexing and slicing
- Groupby
- Concatenating
- Merging Joining
- Missing Values
- Operations
- Data Input and Output
- Pivot
- Cross tab

Data Visualization

- **Introduction to Matplotlib**
 - Line plots
 - Histograms
 - Box and Violin Plots
 - Scatterplot
 - Heatmaps
 - Subplots

Visualization with Seaborn

2. Understanding Text using Python

12. Regular Expressions

- Literals and Meta Characters
- How to Regular Expressions using Pandas?
- Inbuilt Methods
- Pattern Matching

Projects

- **Data Mining**

This project starts completely from scratch which involves collection of Raw Data from different sources and converting the unstructured data to a structured format to apply Machine Learning and NLP models. This project covers the main four steps of Data Science Life Cycle which involves

1. Data Collection
2. Data Mining
3. Data Pre-processing
4. Data Visualization.

Ex: Text, CSV, TSV, Excel Files, Matrices, Images

Statistics - DESCRIPTIVE & INFERENCE

Basic Statistics Terminology

- What is Statistics?
- How Statistics is used in Data Science
- What is Probability?
- Population and Sample
- Sampling Techniques
- Convenience Sampling
- Simple Random Sampling

- Systematic Random Sampling
- Stratified Sampling
- Cluster Sampling
- **Variables**
 - Dependent and Independent Variables
 - Qualitative and Quantitative Data
 - Categorical Data
 - Nominal
 - Ordinal
 - Numerical Data
 - Interval
 - Ratio
 - Discrete and Continuous Data

Central Tendencies

- Mean, Median and Mode
- Standard Deviation and Variance
- Box Plot and Distribution

Basics of Probability

- Probability vs Statistics
- Terminology
- Probability Rules
- Probability Types
 - Marginal Probability
 - Joint Probability
 - Union Probability
 - Conditional Probability

Probability Theory

- Conditional Probability
- Bayes Theorem
- Confusion Matrix
- Z-Score
- Histogram

Probability Distribution

- Expectation
- Variance of Distribution
- Skewness
- Kurtosis
- Discrete Probability Distribution
 - Bernoulli
 - Binomial
 - Geometric
 - Poison
- Continuous Probability Distribution
 - Exponential
 - Normal Distribution
 - Gaussian Distribution
 - t-Distribution
 - **Confidence Interval**
 - Standard Error
 - Margin of Error
- **Statistical Testing**
 - Hypothesis Testing
 - Chi-square test
 - t-test
 - ANOVA

3.MACHINE LEARNING – SUPERVISED LEARNING

1. INTRODUCTION

- What is Machine Learning?

- Difference between Supervised Learning and Unsupervised Learning?
 - Difference between Regression and Classification Models?
- 2. Linear and Multiple Regression**
 - Relationship between variables: Regression (Linear, Multivariate Linear Regression) in prediction.
 - Hands on Linear and Multiple Regression using a use case.
 - Understanding the summary output of Linear Regression
 - Residual Analysis
 - Identifying significant features, feature reduction using AIC, multi-collinearity check, observing influential points, etc.
 - Hypothesis testing of Regression Model
 - Confidence intervals of Slope
 - R-square and goodness of fit
 - Influential Observation – Leverage
 - Polynomial Regression
 - Categorical Variable in Regression
 - 3. Logistic Regression**
 - **Logistic Regression Intuition**
 - Understanding Logit Function.
 - Hands-on Python Session on Logistic Regression using business case.
 - Measuring the Evaluation Metrics – Confusion Metrics, Accuracy, Precision, recall and ROC Curve.
 - 4. Naive Bayes Classifier**
 - Review probability distributions, Joint and conditional probabilities
 - Model Assumptions, Probability estimation
 - Required data processing
 - Feature Selection
 - Classifier
 - 5. Principal Component Analysis (PCA)**
 - Introduction to dimensionality reduction and it's necessity
 - Background: Eigen values, Eigen vectors, Orthogonality
 - Principal components analysis (PCA)
 - Feature Extraction
 - Advantage and application of Dimensionality reduction.
 - 6. Time Series (Forecasting)**
 - Trend analysis
 - Cyclical and Seasonal analysis
 - Smoothing; Moving averages; Auto-correlation; ARIMA
 - Application of Time Series in financial markets
 - 7. Decision Tree (Rule – Based)**
 - Decision nodes and leaf nodes
 - Variable Selection, Parent and child nodes branching
 - Stopping Criterion
 - Tree pruning and Depth of a tree
 - Overfitting
 - Metrics for decision trees-Gini impurity, Information Gain, Variance Reduction
 - Regression using decision tree
 - Interpretation of a decision tree using If-else
 - Pros and cons of a decision tree
 - Accuracy estimation using cross-validation
 - 8. K-Nearest Neighbor (Distance Based Learning)**
 - What is KNN and why do we use it?
 - KNN-algorithm and regression
 - Curse of dimensionality and brief introduction to dimension reduction
 - KNN-outlier treatment and anomaly detection
 - Cross-Validation
 - Pros and cons of KNN
 - 9. Support Vector Machine (Distance Based Learning)**
 - Linear learning machines and Kernel space, making kernels and working in feature space
 - Hands on example of SVM classification and regression problems using a business case in Python.
 - 10. Ensemble Methods**
 - Introduction to Ensemble
 - Bias and Tradeoff

- Bagging & boosting and its impact on bias and variance
- Random forest
- Gradient Boosting
- XGBoost

Case Studies:

- Predictive Analytics
- Banking Use cases – Customer Service prediction,
- Health care Use cases – Heart Disease, Diabetics
- Insurance Use cases
- Telecom Churn Prediction
- Bike Sharing
- Air Quality

4. MACHINE LEARNING – UNSUPERVISED LEARNING

1. Clustering

- Different clustering methods
- review of several distance measures
- Iterative distance-based clustering
- Dealing with continuous, categorical values in K-Means
- Constructing a hierarchical cluster, and density-based clustering.
- Test for stability check of clusters
- Hands-on implementation of each of these methods in Python

2. Recommendation Systems

- Association Rules:
 - How to combine clustering and classification;
 - A mathematical model for association analysis
 - Apriori: Constructs large item sets with mini sup by iterations
 - Metrics of rules-Lift, Support, Confidence, Conviction
- Recommendation Rules:
 - Collaborative Filters
 - Content based Learning

5. Natural Language Processing (NLP) – Text Mining

1. INTRODUCTION

- What is Text Mining?
- Libraries
 - NLTK
 - Spacy
 - TextBlob
- Structured and Unstructured Data
 - Extracting Unstructured text from files and websites

2. Text Preprocessing

- Regular Expressions for Pattern Matching
- Text Normalization
- Text Tokenization
 - Sentence Tokenization
 - Word Tokenization
- Text Segmentation
 - Stemming
 - Lemmatization

3. Natural Language Understanding (NLP Statistical)

- Automatic Tagging
- N-grams Tagging
- Transformation based Tagging
- Bag of Words
- POS Tagging
- TF – IDF
- Cosine Similarity

- Thinking about the math behind text; Properties of words; Vector Space Model
- Named Entity Recognition
- Relation Extraction
- 4. Matrix Factorization**
- Singular Value Decomposition
- 5. Text Indexing**
 - Inverted Indexes
 - Boolean query processing
 - Handling phrase queries, proximity queries
 - Latent Semantic Analysis
- 6. Text Classification**
- Case Studies:**
 - Text Mining
 - Sentiment Analysis
 - Spam Detection
 - Dialogue Prediction

6. Artificial Intelligence

1. Introduction to Neural Networks

- Introduction to Neural Network
- Introduction to Perceptron
- Activation Functions
- Cost Functions
- Gradient Descent
- Stochastic Gradient Descent
- Back propagation

2. Deep Frameworks

- Installing Tensorflow and Keras
- Tensorflow and Keras Basic Syntax
- Tensorflow Graphs
- Variables and Placeholder
- Saving and Restoring Models
- Tensorboard

3. Artificial Neural Network with Tensorflow

- Neural Network for Regression
- Neural Network for Classification
- Evaluating the ANN
- Improving and tuning the ANN

4. Convolution Neural Networks

- Convolution Operation
- ReLU Layer
- Pooling
- Flattening
- Full Connection
- Softmax and Cross Entropy

5. Building Convolution Neural Network in Python

- Introduction to Computer Vision
 - OpenCV library in Python
- Getting Started with Images/Videos
 - Operations on Images
- Image Processing in OpenCV
 - Geometric Transformation of Images
 - Rotation
 - Affine Transformation
 - Perspective Transformation
 - Imaging Thresholding
 - Contours
 - Edge Detections
 - Morphological Transformation
 - Harris Corner Detection

- Reshaping Images
- Normalizing Images
- Building Convolutional Network with Tensorflow
- Training CNN for Image Classification

Case Studies:

- Image Classification
- #### **6. Keras (Backend Tensorflow)**
- Keras vs Tensorflow
 - Introduction to Keras
 - Building Artificial Neural Network with Keras
 - Building Convolution Neural Network with Keras
- #### **7. Natural Processing Language (Sequential Process)**
- The Idea behind Recurrent Neural Networks
 - Vanishing Gradient Problem
 - LSTM (Long Short-Term Memory)
 - GRU (Gated Recurrent Unit)

Projects

- **Face Recognition**

Face Recognition project gives details of the person and can recognize the gender and names. This project involves in

Collection of images

1. Preprocessing the data
2. Applying the Model (Machine Learning or Deep Learning)
3. Training and Testing using the model

Ex: Security Unlock, Gender Recognition, Identity Recognition

- **Chatbot**

Virtual Assistants are now a common requirement for an Organization. But, to make the assistant more effective we are now into the chatbots which involves Natural Language Process, Deep Learning and Artificial Intelligence. This interactive chatbots are designed to serve as an intellectual responsive process.

Ex: Alexa, Siri, Google Assistant

7. Deployment

- Creating pickle and frozen files
- Cloud Deploying Machine Learning and Deep Learning model for production

Django Syllabus (Includes Project)

1. INTRODUCTION TO FRONT END

- Html
- CSS
- JavaScript
- Bootstrap

2. INTRODUCTION TO DJANGO

- Features of Django
- Django web server
- Understanding Django environment
- A simple 'Hello world' application

3. DISPLAYING HYPERLINKS - PROJECT

- Django architecture
- MVC and MTV
- Starting a project
- Django apps
- Activating our first app
- A view that displays a hyperlink
- Mapping the views to URLs
- Running our first app
- Improving the views using templates
- The improved templates
- Template inheritance

- Sending data from url to view
- Sending data from view to template

4. CREATING A WEBSITE - PROJECT

- Starting a project
- Creating an app inside the project
- Activating the app
- Creating model for our site
- Converting the model into a table
- Examples for Fields in Models
- Basic data access using Django shell
- Saving objects into database
- Retrieving objects from database
- Modifying objects of database
- Sorting objects
- Filtering objects
- Deleting objects
- Making changes in the data model

5. CREATING ADMINISTRATION PANEL

- Using the admin interface
- Customizing the admin interface
- Adding users
- Data access and modification using admin panel
- Giving permissions to users

6. CREATING FIRST PAGE OF OUR SITE

- The Django template system
- Template Inheritance
- Improving the website
- Adding background colour for web pages
- Adding banner to the web site
- Adding background image in the web pages
- Storing and displaying images
- Adding users to our site
- Uploading the information by user
- Changing the administration interface title

7. DJANGO FORMS CREATION

- Forms basics
- Creating Contact, Us form
- Form field examples

8. DJANGO's EMAIL FUNCTIONALITY

- Configuring email settings
- Sending emails with Django

9. DJANGO TEMPLATE LANGUAGE

- Django template tags
- If/else
- If/elif/else
- For

- Comments
- Filters
- Using templates to display data in the site

10. INTEGRATING BOOTSTRAP INTO DJANGO

- Introduction to Bootstrap
- Creating tables
- Creating grids
- Creating carousels

11. SESSIONS AND COOKIES

- Difference between session and cookie
- Creating sessions and cookies in Django

12. USING OTHER DATABASES IN DJANGO

- Using SQLite
- Configuring MySQL database
- Working with MySQL in Django
- Configuring Oracle database
- Working with Oracle in Django

13. DJANGO RESTful API

- CRUD operations

14. LIVE PROJECT IMPLEMENTATION

- Project life cycle
- Creating a functional website in Django

SQL Syllabus

SQL Server Training Course Objective

- Learn Database models
- Overview of SQL Server Management Studio and Transact-SQL language
- Master writing simple and complex queries that retrieve data from the database
- Calculate information across result sets using aggregate queries (sum, min, max, avg, etc.)
- Insert, update, and delete data
- Retrieve data from tables
- Joins
- Sub-queries
- Working with Data Types
- Procedure and Functions
- Understand the different Views
- Working with Triggers
- Design a database
- Maintain databases, tables, and sequences with SQL statements
- Create and manage views
- Ensure the integrity of multiple, related database updates by using transactions
- Retrieve data using cursors
- Manage binary data using BLOBs

Introduction To DBMS

- File Management System and Its Drawbacks
- Database Management System (DBMS) and Data Models

- Physical Data Models
- Logical Data Models
 - Hierarchical Data Model (HDBMS)
 - Network Data Model (NDBMS)
 - Relational Data Model (RDBMS)
 - Object Data Model (ODBMS)
 - Object Relational Data Model (ORDBMS)

Conceptual Data Models

- Entity – Relationship (E-R) Model

Introduction To SQL Server

- Advantages and Drawbacks Of SQL Server Compared To Oracle And DB2
 - Connecting To Server
 - Server Type
 - Server Name
 - Authentication Modes
 - Sql Server Authentication Mode
 - Windows Authentication Mode
 - Login and Password
 - Sql Server Management Studio and Tools In Management Studio
 - Object Explorer
 - Object Explorer Details
 - Query Editor

TSQL (Transact-Structured Query Language)

Introduction To TSQL

- History and Features of TSQL
- Types Of TSQL Commands
 - Data Definition Language (DDL)
 - Data Manipulation Language (DML)
 - Data Query Language (DQL)
 - Data Control Language (DCL)
 - Transaction Control Language (TCL)
 - Database
 - Creating Database
 - Altering Database
 - Deleting Database
 - Constrains
 - Procedural Integrity Constraints
 - Declarative Integrity Constraints
 - Not Null, Unique, Default and Check constraints
 - Primary Key and Referential Integrity or foreign key constraints
 - Data Types In TSQL
 - Table
 - Creating Table
 - Altering Table
 - Deleting Table

Data Manipulation Language

- Insert
 - Identity
 - Creating A Table From Another Table
 - Inserting Rows From One Table To Another
 - Update
 - Computed Columns
 - Delete
 - Truncate
 - Differences Between Delete and Truncate

Data Query Language (DQL)

- Select
- Where clause

- Order By Clause
- Distinct Keyword
- Isnull() function
- Column aliases
- Predicates
 - Between ... And
 - In
 - Like
 - Is Null

Built In Functions

- Scalar Functions
 - Numeric Functions
 - Character Functions
 - Conversion Functions
 - Date Functions
 - Aggregate Functions
 - Convenient Aggregate Functions
 - Statistical Aggregate Functions
 - Group By and Having Clauses
 - Super Aggregates
 - Over(partition by ...) Clause
 - Ranking Functions
 - Common Table Expressions (CTE)

Top n Clause

Set Operators

- Union
- Intersect
- Except

Joins

- Inner Join
 - Equi Join
 - Natural Join
 - Non-Equi Join
 - Self Join
 - Outer Join
 - Left Outer Join
 - Right Outer Join
 - Full Outer Join
 - Cross Join

Sub Queries

- Single Row Sub Queries
- Multi Row Sub Queries
 - Any or Some
 - ALL
 - Nested Sub Queries
 - Co-Related Sub Queries
 - Exists and Not Exists

Indexes

- Clustered Index
- NonClustered Index
- Create , Alter and Drop Indexes
- Using Indexes

Security

- Login Creation
 - SQL Server Authenticated Login
 - Windows Authenticated Login
 - User Creation

- Granting Permissions
- Revoking Permissions
- Roles

Views

- Purpose Of Views
- Creating , Altering and Dropping Indexes
- Simple and Complex Views
- Encryption and Schema Binding Options in creating views

Transaction Management

- Introduction
- Begin Transaction
- Commit Transaction
- Rollback Transaction
- Save Transaction
- Role Of Log File In Transaction Management
- Implicit Transactions

TSQL Programming

- Drawbacks Of TSQL that leads to TSQL Programming
- Introduction To TSQL Programming
- Control statements In TSQL Programming
 - Conditional Control Statements
 - If
 - Case

Looping Control Statements

- While

Cursors

- Working With Cursors
- Types Of Cursors
 - Forward_Only and Scroll Cursors
 - Static, Dynamic and Keyset Cursors
 - Local and Global Cursors

Stored Sub Programs

- Advantages Of Stored Sub Programs compared to Independent SQL Statements
- Stored Procedures
 - Creating , Altering and Dropping
 - Optional Parameters
 - Input and Output Parameters
 - Permissions on Stored Procedures

User Defined Functions

- Creating, Altering and Dropping
- Types Of User Defined Functions
 - Scalar Functions
 - Table Valued Functions
 - Inline Table Valued Functions
 - Multi Statement Table Valued Functions
- Permissions On User Defined Functions

Triggers

- Purpose of Triggers
- Differences Between Stored Procedures and User Defined Functions and Triggers
- Creating, Altering and Dropping Triggers
- Magic Tables
- Instead Of Triggers

Exception Handling

- Implementing Exception Handling
- Adding and removing User Defined Error Messages To And From SQL Server Error Messages List
- Raising Exceptions Manual

CLR Integration

- What is CLR Integration and The Steps For Implementing It
- A Simple Example With CLR Integration

Working With XML Data Type
Backup and Restore Of Database
Attach and Detach of Database
Normalization

NoSQL Database -MongoDB:

OVERVIEW OF MongoDB

- Introduction of MongoDB
- No SQL Database
- Advantage over RDBMS
- MongoDB Data Types
- Install MongoDB
- MongoDB Data Modeling

MongoDB Operators

- Query & Projection Operator
- MongoDB Update Operator
- Aggregation Pipeline Stages

- MongoDB limit()
- MongoDB sort()
- Query Modifiers

Database Commands

- Aggregation Commands
- Geospatial Command
- Query and Write Operation Commands
- Query Plan Cache Commands
- Authentication Commands
- User Management Commands
- Role Management Commands
- Replication Command
- Sharding Commands
- Session Commands

Database

- Create Database
- Drop Database

Collection

- Create Collection
- Drop Collection

CRUD : Documents

- Inset Documents
- Update Documents
- Delete Documents
- Query Documents
- SQL to MongoDB Mapping
- MongoDB text search
- Partial Updates & Document Limits
- Removing Documents
- Multi Update
- Upsert
- Wire Protocol
- Bulk() Operations and Methods
- Common Commands
- db.runCommand()
- db.isMaster()
- db.serverStatus()
- db.currentOp() & db.killOp()
- collection.stats() & collection.drop()
- MongoDB Shell

MongoDB Shell

- Shell Collection Methods
- Cursor Method
- MongoDB Database Commands
- Query Plan Cache Methods
- User Management Method
- Role Management Method
- MongoDB Replication Methods

MongoDB Cloud

- MongoDB Stitch
- MongoDB Atlas
- MongoDB Cloud Manager

- MongoDB Ops Manager

MongoDB Tools

- MongoDB Compass
- MongoDB BI connector

Connectivity

- Java MongoDB
- PHP MongoDB
- Python MongoDB

GIT Syllabus:

Introduction

- What is a Version Control System (VCS)?
- Distributed vs Non-distributed VCS
- What is Git and where did it come from?
- Alternatives to Git
- Cloud-based solutions (Github, Gitlab, BitBucket etc)

Installation and Configuration

- Obtaining Git
- Installing Git
- Common configuration options
- GUI tools

Key Terminology

- Clone
- Working Tree
- Checkout
- Staging area
- Add
- Commit
- Push
- Pull
- Stash

Git - Local Repository Actions

- Creating a repository (git init)
- Checking status (git status)
- Adding files to a repository (git add)
- Committing files (git commit)
- Removing staged files (git reset)
- Removing committed files (git rm)
- Checking logs (git log)

Git - Remote Repository Actions

- Creating a remote repository (git init)
- Cloning repositories (git clone)
- Updating the remote repository from the local (git push)
- Updating the local repository from the remote (git pull)

Tagging in Git

- What are Git Tags?
- Listing tags
- Lightweight tags
- Displaying tag details (tag show)
- Annotated tags
- Checking out tags
- Pushing tags
- Pulling tags

Branching in Git

- What is a branch
- A note about `andlt;HEADandgt;`
- Listing branches
- Create new branch
- Checkout branch
- Pushing branches
- Pulling branches

Merging in Git

- Fetching Changes (git fetch)
- Rebasing (git rebase)
- Git Pull

Git Workflows

- Different ways of using Git

- Centralised
- Feature Branch
- Gitflow Workflow
- Forking Workflow

Git - Stashing Changes

- What is Stashing?
- Using Stash
- Creating a branch from a Stash

Advanced Repository Actions

- Removing untracked files (git clean)
- Remove staged changes (git reset)
- Revert a commit (git revert)
- Checkout a previous commit (git checkout)

Advanced Branching and Merging

- Deleting a Branch
- Fast forward merge
- Three way merge
- Resolving merge conflicts
- Cherry-Picking (git cherry-pick)

Advanced Git Configuration

- Aliases
- Submodules
- Patches
- Hooks

Docker Syllabus:

Module 1: Docker World – An introduction

- 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
- Customizing container on your own
- Running Container with Docker – commands
- Port forwarding with docker container
- Exercise: Installation of docker and Image Setup
- Exercise: Creating own Images
- Exercise: Creating own Images
- Exercise: Exposing Container Ports to the Host and test it

Module 2:The Dockerfile, Builds and Network Configuration

- 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

Module 3: Docker Commands and Structures

- Inspect Container Processes
- Previous Container Management
- Controlling Port Exposure on Containers
- Naming Our Containers
- Docker Events
- Managing and Removing Base Images
- Saving and Loading Docker Images
- Image History
- Taking Control of Our Tags
- Pushing to Docker Hub
- Exercise: Base Image Maintenance and Clean-up
- Exercise: Advanced Container Creation at the Command Line
- Exercise: Create a Dockerized Basic Web Server
- Continuous Integration for Docker

Module 4: Docker-Compose

- Networking Overview
- The Default Network
- Isolating Containers
- Aliases & Container Names
- Links
- How Updates Affect Networking
- Using External Networks
- Configuring Compose
- Bringing an Environment Up
- Changing a Running Environment

- Introspecting On An Environment
- Taking an Environment Down

AWS Services for Data Science and Data Engineering /Data Analyst

1.S3
2.EC2
3.Lambda
4.Step function
5.IAM
6.SNS
7.SQS
8.Sagemaker
9.ECR
10.SES
11.RDS
12.Redshift
13.AmazonDynamoDB
14.AWS Glue
15.AWS kinesis
16.AWS Athena
17.Quicksight
18.AWS Cloudwatch
19.ECS
20.Git/Cloud Commit

I) PYSPARK INTRODUCTION

- What is Apache Spark?
- Why Pyspark?
- Need for pyspark
- spark Python Vs Scala
- pyspark features
- Real-life usage of PySpark
- PySpark Web/Application
- PySpark - SparkSession
- PySpark – SparkContext
- PySpark – RDD
- PySpark – Parallelize
- PySpark – repartition() vs coalesce()
- PySpark – Broadcast Variables
- PySpark – Accumulator

II) PYSPARK - RDD COMPUTATION

- Operations on a RDD
- Direct Acyclic Graph (DAG)
- RDD Actions and Transformations
- RDD computation
- Steps in RDD computation
- RDD persistence
- Persistence features

II) PERSISTENCE Options:

- 1) MEMORY_ONLY
- 2) MEMORY_SER_ONLY
- 3) DISK_ONLY
- 4) DISK_SER_ONLY
- 5) MEMORY_AND_DISK_ONLY

III) PYSPARK - CORE COMPUTING

- Fault Tolerance model in spark
- Different ways of creating a RDD
- Word Count Example
- Creating spark objects(RDDs) from Scala Objects(lists).
- Increasing the no of partitons
- Aggregations Over Structured Data:
- reduceByKey()

IV) GROUPINGS AND AGGREGATIONS

- i) Single Grouping and Single Aggregation
- ii) Single Grouping and multiple Aggregation
- iii) multi Grouping and Single Aggregation
- iv) Multi Grouping and Multi Aggregation
- Differences b/w reduceByKey() and groupByKey()
- Process of groupByKey
- Process of reduceByKey
- Reduce() function
- Various Transformations
- Various Built-in Functions

V) Various Actions and Transformations:

- countByKey()
- countByValue()
- sortByKey()
- zip()
- Union()
- Distinct()
- Various count aggregation
- Joins
- -inner join
- -outer join
- Cartesian()
- Cogroup()
- Other actions and transformations

VI) PySpark SQL - DataFrame

- Introduction
- Making data Structured
- Case Classes
- ways to extract case class objects
- 1) using function
- 2) using map with multiple expressions
- 3) using map with single expression
- Sql Context
- Data Frames API
- DataSet API
- RDD vs DataFrame vs DataSet
- PySpark – Create a DataFrame
- PySpark – Create an empty DataFrame
- PySpark – Convert RDD to DataFrame
- PySpark – Convert DataFrame to Pandas
- PySpark – show()
- PySpark – StructType & StructField
- PySpark – Row Class
- PySpark – Column Class
- PySpark – select()
- PySpark – collect()
- PySpark – withColumn()
- PySpark – withColumnRenamed()
- PySpark – where() & filter()
- PySpark – drop() & dropDuplicates()
- PySpark – orderBy() and sort()
- PySpark – groupBy()
- PySpark – join()
- PySpark – union() & unionAll()
- PySpark – unionByName()
- PySpark – UDF (User Defined Function)
- PySpark – map()
- PySpark – flatMap()
- pyspark – foreach()
- PySpark – sample() vs sampleBy()
- PySpark – fillna() & fill()
- PySpark – pivot() (Row to Column)
- PySpark – partitionBy()
- PySpark – ArrayType Column (Array)
- PySpark – MapType (Map/Dict)

VII) PySpark SQL Functions

- PySpark – Aggregate Functions
- PySpark – Window Functions
- PySpark – Date and Timestamp Functions
- PySpark – JSON Functions
- PySpark – Read & Write JSON file

VIII) PySpark Built-In Functions

- PySpark – when()
- PySpark – expr()
- PySpark – lit()
- PySpark – split()
- PySpark – concat_ws()
- PySpark – substring()
- PySpark – translate()
- PySpark – regexp_replace()
- PySpark – overlay()
- PySpark – to_timestamp()
- PySpark – to_date()
- PySpark – date_format()
- PySpark – datediff()
- PySpark – months_between()
- PySpark – explode()
- PySpark – array_contains()
- PySpark – array()
- PySpark – collect_list()
- PySpark – collect_set()
- PySpark – create_map()
- PySpark – map_keys()
- PySpark – map_values()
- PySpark – struct()
- PySpark – countDistinct()
- PySpark – sum(), avg()
- PySpark – row_number()
- PySpark – rank()
- PySpark – dense_rank()
- PySpark – percent_rank()
- PySpark – typedLit()
- PySpark – from_json()
- PySpark – to_json()
- PySpark – json_tuple()
- PySpark – get_json_object()
- PySpark – schema_of_json()
- Working Examples

IX) Pyspark External Sources

- Working with sql statements
- Spark and Hive Integration
- Spark and mysql Integration
- Working with CSV
- Working with JSON
- Transformations and actions on dataframes
- Narrow, wide transformations
- Addition of new columns, dropping of columns ,renaming columns
- Addition of new rows, dropping rows
- Handling nulls

- Joins
- Window function
- Writing data back to External sources
- Creation of tables from Dataframes (Internal tables, Temporary tables)

X) DEPLOYMENT MODES

- Local Mode
- Cluster Modes (Standalone, YARN)

XI) PYSPARK APPLICATION

- Stages and Tasks
- Driver and Executor
- Building spark applications/pipelines
- Deploying spark apps to cluster and tuning
- Performance tuning

PySpark Streaming Concepts

Integration with Kafka

PySpark-mllib

Linux Commands:

Module 1 – Understanding Linux Concepts

- What is Linux?
- Everyday use of Linux
- Unix vs. Linux
- Quiz, Homework and Handouts

Module 2 – Download, Install and Configure

- What is Oracle Virtual Box?
- Downloading and Installing Oracle Virtual Box
- Creating virtual machine
- Linux Distributions
- Different way to install Linux
- Downloading and Installing Linux (CentOS)
- Redhat Linux installation (Optional)
- Linux Desktop (GUI)
- Virtual Machine Management
- Linux vs. Windows
- Who Uses Linux?

Module 3 – System Access and File System

- Accessing Linux system
- Download and install Putty
- New Network Commands (ifconfig and ip)
- Connect Linux VM via Putty
- Important Things to Remember in Linux
- Introduction to File System
- File system structure description

- File system navigation commands
- File System Paths
- Directory listing overview
- Creating Files and Directories
- Linux File Types
- Finding Files and Directories (find, locate)
- Difference between find and locate command
- Changing Password
- Wildcard (*, \$, ^)
- Soft and Hard Links (ln)
- How to open image file through GUI

Module 4 – Linux Fundamentals

- Commands Syntax
- File Permissions (chmod)
- File Ownership (chown, chgrp)
- Getting Help (man, whatis etc.)
- TAB completion and up arrow keys
- Adding text to file
- Standard output to a file (tee command)
- Pipes (|)
- File Maintenance Commands
- File Display Commands
- Filters / Text Processing Commands (cut, sort, grep, awk, uniq, wc)
- Compare Files (diff, cmp)
- Compress and un-compress files/directories (tar, gzip, gunzip)
- Truncate file size (truncate)
- Combining and Splitting Files (cat and split)
- Linux vs. Windows Commands
- Quiz, Homework and Handouts

Module 5 – Linux System Administration

- Linux File Editors (vi text editor)
- “sed” command
- User account management
- Switch users and Sudo access
- Monitor users
- Talking to users (users, wall, write)
- Linux Directory Service - Account Authentication
- System utility commands (date, uptime, hostname, which, cal, bc etc.)
- Processes and schedules (systemctl, ps, top, kill, crontab and at)
- System Monitoring Commands (top, df, dmesg, iostat 1, netstat, free etc.)
- OS Maintenance Commands (shutdown, reboot, halt, init etc.)
- System logs monitor (/var/log)
- Changing System Hostname (hostnamectl)
- Finding System Information (uname, cat /etc/redhat-release, cat /etc/*rel*, dmidecode)
- System Architecture (arch)
- Terminal control keys
- Terminal Commands (clear, exit, script)
- Recover root Password (single user mode)
- SOS Report
- Environment variables
- Quiz, Homework and Handouts

Module 6 – Shell Scripting

- Linux Kernel

- What is a Shell?
- Types of Shells
- Shell scripting
- Basic Shell scripts
- If-then scripts
- For loop scripts
- Do-while scripts
- Case statement scripts
- Aliases
- Command history

Module 7 – Networking, Servers and System Updates

- Enabling internet in Linux VM
- Network Components
- Network files and commands (ping, ifconfig, netstat, tcpdump, networking config files)
- NIC Information (ethtool)
- NIC or port bonding
- Download files with URLs (wget)
- curl and ping commands
- File transfer commands (ftp, scp etc.)
- System updates and repositories (rpm and yum)
- System Upgrade/Patch Management
- Create Local Repository from CD/DVD
- Advance Package Management
- Rollback Patches and Updates
- SSH and Telnet
- DNS
- Hostname and IP Lookup (nslookup and dig)
- NTP
- chronyd
- Sendmail
- Apache Web Server (http)
- Central Logger (rsyslogd)
- Securing Linux Machine (OS Hardening)
- OpenLDAP Installation
- Tracing Network Traffic (traceroute)
- Quiz, Homework and Handouts

Module 8 – Disk Management and Run Levels

- System run levels
- Linux Boot Process
- Message of the Day
- Customize Message of the Day
- Storage
- Disk partition (df, fdisk, etc.)
- Add Disk and Create Standard Partition
- Logical Volume Management (LVM)
- LVM Configuration during Installation
- Add Disk and Create LVM Partition
- Extend disk using LVM
- Adding swap space
- RAID
- File System Check (fsck and xfs_repair)
- System Backup (dd Command)
- Network File System (NFS)
- Difference Between CentOS/Redhat 5, 6 and 7
- Quiz, Homework and Handouts

