LEARN 2 EARN LABS

Training Syllabus

[3 & 6 Months Duration]

Data Analytics using Python

Join us for Better Career and Job Guarantee

[Live Online & Offline Classes Available]

LEARN-2-EARN LABS TRAINING INSTITUTE, AGRA

Website: www.LearntoearnLabs.com | Contact : +91-9548868337

Data Analytics Using Python Training Program

In today's data-driven world, businesses and organizations rely on data analytics to make informed decisions, gain insights, and optimize processes. The **Data Analytics Using Python Training Program** is designed to provide you with hands-on experience in data handling, manipulation, visualization, statistical analysis, and machine learning, making you a skilled data analyst ready for industry challenges.

Our training program is offered in two durations:

- **3-Months Program** (Covers Core Data Analytics Concepts – 9 Modules).
- **6-Months Program** (Covers Advanced Data Analytics & Machine Learning – 15 Modules).

Whether you're a beginner, an IT professional, or a career switcher, this program will equip you with industry-relevant data analytics skills using Python.

About the Program

This comprehensive training program is structured to take you from beginner to expert level in data analytics using Python. You will learn:

- Python programming essentials for data analytics
- Data manipulation & analysis using NumPy & Pandas
- Data visualization using Matplotlib, Seaborn, and Plotly
- Handling large datasets and file processing
- Statistical analysis & Exploratory Data Analysis (EDA)
- Machine learning techniques (Regression, Classification, Clustering)
- Model evaluation & optimization techniques

By the end of this program, you will be ready to analyze real-world datasets, build interactive dashboards, and implement predictive analytics models using Python.

Who is a Data Analyst?

A Data Analyst is a professional who collects, processes, and analyzes data to help businesses make strategic decisions. Data analysts use statistical techniques, machine learning, and visualization tools to identify trends, detect patterns, and generate insights from structured and unstructured data.

Data Analysts play a crucial role in industries like finance, healthcare, e-commerce, banking, social media, and marketing, making it one of the **most in-demand careers** today.

What Makes This Program Unique?

- **Structured Learning Path** – Covers foundational to advanced data analytics concepts.
- **Hands-On Projects** – Learn through real-world datasets and live projects.
- **Industry-Relevant Curriculum** – Aligned with current industry standards in data analytics.
- **Comprehensive Coverage** – From basic Python programming to advanced machine learning.
- **Practical Application Focus** – Learn to handle, clean, visualize, and analyze data.

This program is designed to **help you transition into the field of data analytics** with practical hands-on experience in data science libraries and tools.

Career Options After Completion

After completing this training program, you will be eligible for multiple job roles, including:

- Data Analyst
- Business Analyst
- Data Scientist (Entry-Level)
- Market Research Analyst
- Machine Learning Engineer (For 6-Month Program)
- Financial Data Analyst
- Healthcare Data Analyst

With the rapid growth in data-driven decision-making, data analytics professionals are in high demand across industries such as banking, healthcare, retail, finance, and e-commerce.

Who Can Join This Program?

- **Students** – Looking to start a career in data analytics & data science.
- **Working Professionals** – Wanting to upskill or switch careers into data analytics.
- **Business Analysts** – Looking to enhance their data analysis and visualization skills.
- **Developers & IT Professionals** – Seeking data-driven decision-making skills.
- **Entrepreneurs** – Interested in leveraging data analytics for business growth.

No prior experience in Python or data analytics is required! Beginners are welcome as we cover everything from scratch.

Advantages of the Program

- **Complete Data Analytics Training** – Covers Python, Pandas, NumPy, Matplotlib, Seaborn, Plotly, Machine Learning.
- **Real-World Projects** – Hands-on experience with business datasets.
- **Career-Focused Curriculum** – Gain in-demand skills for high-paying job roles.
- **Hands-On Learning** – Work on live projects & case studies.
- **Industry-Recognized Training** – Learn the tools used by data professionals worldwide.
- **Flexible Learning Duration** – Choose between a 3-months (basic) or 6-months (advanced) program.

This program prepares you for the job market by ensuring practical learning & real-world implementation.

How Much Dedication is Required?

This program is designed to be intensive yet flexible, allowing you to balance learning with your existing schedule.

- **For the 3-months training program** – You should dedicate 10-12 hours per week.
- **For the 6-months training program** – You should dedicate 16-20 hours per week, focusing on advanced analytics and machine learning.

Regular hands-on practice with Python, data manipulation, and visualization tools is essential to grasp concepts effectively. Completing assignments, exercises, and projects will reinforce learning and help you gain practical experience. Consistency and dedication to problem-solving will ensure that you develop the critical thinking skills needed to become a proficient data analyst.

Future Scope

Data analytics is the future of business decision-making. By completing this course, you will:

- Have strong Python programming and data analysis skills.
- Be able to perform statistical analysis and EDA on datasets.
- Create interactive dashboards and reports.
- Build machine learning models (for the 6-months program).
- Be industry-ready for data analyst and machine learning engineer roles.

Data analytics is a **booming industry** with **high-paying job opportunities** worldwide. The demand for skilled data analysts is growing rapidly across industries like finance, healthcare, e-commerce, and marketing.

Practical Experience to Boost Your Career

This program includes practical assignments and projects to help you gain practical experience:

- **Case Studies** – Work on business datasets from finance, healthcare, and e-commerce.
- **Capstone Project** – Apply everything learned to a real-world data problem.
- **Portfolio Building** – Create a GitHub repository with your projects.
- **Resume Preparation & Career Guidance** – Get ready for data analyst job interviews.

During the training, you will learn how to clean, manipulate, visualize, and analyze data just like a professional data analyst. Each module includes practical exercises that reinforce your understanding through industry-relevant problem-solving. By the end of the program, you will have **strong analytical skills** and a **portfolio showcasing your expertise** to potential employers.

Transform Your Career Today

Data Analytics is one of the hottest career fields, and Python is the most in-demand programming language for data analysis.

This program will equip you with the skills to enter the world of data analytics and machine learning, helping you land your dream job or advance in your current role. Whether you are looking to start a career as a Data Analyst, transition from another field, or enhance your problem-solving and analytical abilities, this training will give you a competitive edge.

With the growing reliance on data-driven decision-making, professionals with data analytics expertise are highly valued across industries worldwide. Now is the perfect time to **take action and shape your future** in the rapidly evolving field of data analytics.

Your Journey to Success Starts Here!

Take the first step toward mastering data analytics. Whether you choose the 3-months or 6-months program, this training will help you **become a data analytics expert**.

Data is shaping the future, and those who master it will lead the way! By joining this program, you are not just learning a skill; you are investing in a high-growth career with endless opportunities. Every dataset tells a story—be the one who uncovers the insights and drives impactful decisions!

Enroll now and start your journey in data analytics with Python! 🚀

Training Content (Syllabus)

Module 1 - Python Basics : Introduction to Python, Why use Python for Data Analytics?, Applications of Python in Data Science, AI, and Machine Learning, Key Features of Python, Python 2 vs. Python 3 Differences; Setting Up Python and Jupyter Notebook - Installing Python (Anaconda, Miniconda, or Standalone Python), Introduction to Jupyter Notebook and JupyterLab, Running Python Scripts and Jupyter Cells, Python Virtual Environments and Package Management (pip, conda); Python Syntax, Variables, and Data Types - Understanding Variables and Constants, Primitive Data Types: Integers, Floats, Strings, Booleans; Type Conversion (Casting); Dynamic vs. Static Typing; Operators and Expressions - Arithmetic Operators (+, -, *, /, %, //, **), Comparison Operators (==, !=, >, <, >=, <=), Logical Operators (and, or, not), Bitwise Operators, Assignment Operators, Operator Precedence and Associativity.

Module 2 – Programming Concepts using Python: Conditional Statements (if, elif, else), Nested Conditionals, Short-Circuit Evaluation; Loops in Python - For Loop (Iterating over Sequences), While Loop (Conditional Execution), Loop Control Statements: break, continue, pass; Nested Loops; Using enumerate() and zip() in Loops; Functions and Lambda Functions - Defining and Calling Functions, Function Arguments (Positional, Keyword, Default, and Arbitrary Arguments), Returning Values, Scope of Variables (Local, Global, and Nonlocal), Anonymous (Lambda) Functions and Use Cases, Map, Filter, and Reduce Functions; Exception Handling in Python - Understanding Errors and Exceptions, Try-Except Blocks, Handling Multiple Exceptions, Using Finally and Else Blocks, Raising Custom Exceptions.

Module 3 - Python Data Structures : Lists and List Comprehensions - Introduction to Lists, Indexing, Slicing, and Mutability, Adding and Removing Elements (append(), insert(), remove(), pop()), Sorting and Reversing Lists (sort(), sorted(), reverse()), Copying Lists (copy(), list(), [:]), List Comprehensions (Filtering and Transformations); Tuples - Understanding Tuples and Their Immutability, Tuple Operations (Indexing, Slicing, Concatenation), Packing and Unpacking Tuples, Named Tuples (collections.namedtuple); Dictionaries and Dictionary Comprehensions - Understanding Key-Value Pairs, Creating and Accessing Dictionaries, Dictionary Methods (get(), keys(), values(), items(), pop(), update()), Iterating Over Dictionaries, Nested Dictionaries, Dictionary Comprehensions; Sets and Frozen Sets - Understanding Sets and Their Uniqueness Property, Creating and Modifying Sets (add(), update(), remove(), discard()), Set Operations (Union, Intersection, Difference, Symmetric Difference), Frozen Sets (Immutable Sets), Use Cases of Sets in Data Analytics.

Module 4 - NumPy for Data Handling : Introduction to NumPy Arrays, Why is it Important?, Installing NumPy (pip install numpy), NumPy vs. Python Lists (Performance

Comparison), Understanding NumPy Arrays (ndarray) and Their Benefits, Creating NumPy Arrays (array(), arange(), linspace(), zeros(), ones(), eye()); Creating and Manipulating NumPy Arrays - Generating Random Numbers (random module), Reshaping Arrays (reshape(), ravel(), flatten()), Concatenating, Stacking, and Splitting Arrays (hstack(), vstack(), split()), Copying and Cloning Arrays (copy(), deepcopy()), Broadcasting in NumPy; Indexing, Slicing, and Iterating Over Arrays - Accessing Elements in 1D, 2D, and 3D Arrays ([], .item()), Slicing and Dicing Arrays ([start:stop:step]), Boolean Indexing and Fancy Indexing, Iterating Over Arrays (nditer(), enumerate()); Mathematical and Statistical Functions in NumPy - Basic Arithmetic Operations (+, -, *, /, %, **), Aggregation Functions (sum(), min(), max(), mean(), median(), std(), var()), Trigonometric Functions (sin(), cos(), tan()), Exponential and Logarithmic Functions (exp(), log(), log10()), Linear Algebra Operations (dot(), matmul(), det(), inv(), eig()); Handling Missing Values with NumPy - Identifying Missing Data (isnan()), Replacing Missing Values (nan_to_num()), Masking and Filtering Data.

Module 5 - Pandas for Data Manipulation : What is Pandas, and Why Use It?, Installing Pandas (pip install pandas), Understanding Series and DataFrame Objects, Creating DataFrames from Dictionaries, Lists, and NumPy Arrays; Loading Data from CSV, Excel, and Databases - Reading CSV Files (read_csv()), Reading Excel Files (read_excel()), Connecting to Databases (read_sql()), Writing Data to CSV, Excel, and Databases (to_csv(), to_excel(), to_sql()); Data Cleaning, Handling Missing and Duplicate Values - Identifying Missing Data (isnull(), notnull()), Filling Missing Values (fillna(), interpolate(), ffill(), bfill()), Dropping Missing Data (dropna()), Removing Duplicate Rows (drop_duplicates()); Data Transformation: Filtering, Sorting, and Grouping - Filtering Data (loc[], iloc[], query()), Sorting Data (sort_values(), sort_index()), Applying Functions to Columns (apply(), map(), lambda functions), Grouping Data (groupby()); Merging and Joining Datasets - Concatenating DataFrames (concat()), Merging DataFrames (merge()), Joining DataFrames (join()), Handling Overlapping Column Names; Data Aggregation and Pivot Tables - Aggregation Functions (mean(), sum(), count(), std()), Pivot Tables (pivot_table()), Cross-tabulation (crosstab()); Working with DateTime Data - Parsing Dates (pd.to_datetime()), Extracting Date Components (.dt.year, .dt.month, .dt.day), Date Arithmetic and Time Series Analysis.

Module 6 - Working with Files : Introduction to File Handling, Types of Files in Python (Text Files vs. Binary Files), File Paths and Directory Navigation (os module), Opening and Closing Files (open(), close()), Modes of File Handling - Read Mode ('r'), Write Mode ('w'), Append Mode ('a'), Read and Write Mode ('r+', 'w+', 'a+'), Binary Mode ('rb', 'wb', 'ab'); Using the with Statement for File Handling, Reading and Writing Text Files - Reading Files Line by Line (read(), readline(), readlines()), Writing to Files (write(), writelines()), Appending Data to Files ('a' mode), File Cursor Positioning (seek(), tell()), Working with Multi-Line Text Files, Error Handling in File Operations (try-except-

finally); Introduction to CSV Files and Their Structure, Reading CSV Files Using csv.reader(), Reading CSV Files into Lists and Dictionaries, Writing Data to CSV Files Using csv.writer(), Appending Data to Existing CSV Files, Working with CSV Files Using pandas (read_csv(), to_csv()), Handling Missing Values in CSV Files (fillna(), dropna()); Introduction to JSON Format and Its Usage, Reading JSON Files Using json.load(), Writing JSON Data to a File (json.dump()), Converting JSON Strings to Python Objects (json.loads()), Converting Python Objects to JSON (json.dumps()), Working with Nested JSON Data, Reading JSON Files Using Pandas (pd.read_json()); Handling Large Files in Python - Using readline() and readlines() for Memory-Efficient File Processing, Using Iterators to Process Large Files (for line in file), Using mmap for Fast File I/O; Working with Memory-Efficient Data Processing - Processing Large CSV Files Using pandas.read_csv() with chunksize, Optimizing Performance with dtypes and converters in Pandas, Filtering Data in Large CSV Files Without Loading the Entire Dataset; Using Generators for Handling Large Data - Introduction to Generators (yield Keyword) and Lazy Iteration, Creating a Generator to Process Large Files Line-by-Line, Practical Implementation: Processing Multi-Gigabyte Data Without Crashing Memory; Comparing Performance: List vs. Generator in Large File Processing.

Module 7 - Matplotlib for Basic Visualization : Overview of Matplotlib and its Role in Data Visualization, Installing Matplotlib (pip install matplotlib), Understanding the pyplot API vs. Object-Oriented API, Matplotlib Figure Anatomy (Figures, Axes, Ticks, Grid), Creating and Displaying Plots (plt.show()), Embedding Matplotlib Charts in Jupyter Notebooks; Line Charts, Bar Charts, Histograms, and Scatter Plots - Creating Line Charts (plot()) for Time Series Data, Customizing Line Styles (Dashed, Dotted, Markers), Creating Bar Charts (bar(), barh()) for Categorical Data, Grouped and Stacked Bar Charts, Creating Histograms (hist()) for Data Distributions, Adjusting Bins and Density Curves (normed, kde), Creating Scatter Plots (scatter()) for Correlation Analysis; Adding Trendlines and Regression Curves, Customizing Plots (Titles, Labels, Legends, Annotations) - Adding Titles (title()) and Axis Labels (xlabel(), ylabel()), Formatting Ticks (xticks(), yticks()), Adjusting Line Width, Opacity, and Markers, Positioning and Styling Legends (legend()), Annotating Data Points (annotate(), text()); Using Gridlines and Background Colors for Better Readability; Multi-Plot Figures and Subplots - Creating Multiple Plots (subplots()), Using Different Layouts (gridspec, subplot2grid), Adjusting Figure Size and Aspect Ratio, Sharing Axes and Removing Unnecessary Ticks; Saving and Exporting Charts - Saving Charts in Different Formats (savefig()), Adjusting DPI for High-Quality Exports, Embedding Charts in PDFs and Reports.

Module 8 - Advanced Visualization with Seaborn : Introduction to Seaborn, Why Use It?, Installing Seaborn (pip install seaborn), Seaborn vs. Matplotlib: Key Differences, Understanding Built-in Seaborn Datasets (sns.load_dataset()), Customizing Themes (darkgrid, whitegrid, ticks); Pair Plots and Distribution Plots - Creating Pair Plots

(pairplot()) for Multi-Variable Relationships, Using Hue Parameter for Categorical Coloring, Creating Distribution Plots (histplot(), kdeplot()), Comparing Distributions with Multiple KDE Curves; Box Plots and Violin Plots - Creating Box Plots (boxplot()) for Outlier Detection, Grouping Data Using Box Plots (hue, palette), Creating Violin Plots (violinplot()) for Density Estimation, Combining Box and Violin Plots for In-Depth Insights; Heatmaps for Correlation Analysis - Understanding Correlation Matrices (df.corr()), Creating Heatmaps (heatmap()) for Feature Relationships, Customizing Heatmaps (Annotations, Color Maps, Scaling), Handling Large Matrices with Clustering Heatmaps.

Module 9 - Interactive Data Visualization : Introduction to Plotly, Why Use It?, Installing Plotly (pip install plotly), Understanding Plotly Express vs. Graph Objects, Creating Interactive Line Charts and Bar Charts; Creating Interactive Charts and Dashboards - Creating Interactive Scatter Plots (scatter()), Creating Interactive Pie Charts (pie()), Creating Interactive Histograms and Box Plots, Using Hover Effects, Sliders, and Dropdown Menus, Building Multi-Page Dashboards Using Dash; Visualizing Geospatial Data with Folium - Introduction to Geospatial Data Visualization, Installing Folium (pip install folium), Creating Interactive Maps with Folium, Adding Markers, Popups, and Tooltips, Plotting Choropleth Maps for Geospatial Data Analysis, Integrating Folium with Pandas for Data Mapping.

Module 10 - Statistical Concepts for Data Analytics : Introduction to Statistics, Why is it important for Data Analytics?, Descriptive vs. Inferential Statistics, Applications of Statistics in Real-World Data Science; Descriptive Statistics (Summarizing Data) - Measures of Central Tendency (Mean (np.mean()), Median (np.median()), Mode (scipy.stats.mode())), Measures of Dispersion (Variance (np.var()), Standard Deviation (np.std()), Range), Quartiles and Interquartile Range (IQR) (np.percentile()), Identifying Skewness & Kurtosis, Visualizing Distributions: Histograms, Box Plots, Density Plots; Probability and Probability Distributions - Understanding Probability Theory (Basic Probability Rules), Types of Probability: Conditional Probability (P(A|B)), Bayes' Theorem (P(A|B) = P(B|A) * P(A) / P(B)), Discrete vs. Continuous Probability Distributions; Common Probability Distributions: Binomial Distribution (np.random.binomial()), Poisson Distribution (np.random.poisson()), Uniform Distribution (np.random.uniform()), Normal Distribution (np.random.normal()).

Note : Available only with six months duration training of Data Analytics.

Module 11 - Hypothesis Testing, Correlation and Covariance : What is Hypothesis Testing (Statistical Significance)? Null & Alternative Hypotheses (H0, H1), Understanding p-values & significance levels (α = 0.05, 0.01, 0.1), One-Tailed vs. Two-Tailed Tests; Types of Hypothesis Tests in Python: t-test (scipy.stats.ttest_ind()) (Comparing two sample means), Chi-Square Test (scipy.stats.chi2_contingency()) (Categorical variable independence), ANOVA (scipy.stats.f_oneway()) (Comparing

multiple group means); Correlation and Covariance (Feature Relationships) - Understanding Covariance (np.cov()), Pearson Correlation Coefficient (np.corrcoef()), Spearman Rank Correlation (scipy.stats.spearmanr()), Kendall Rank Correlation (scipy.stats.kendalltau()), Heatmaps for Correlation Analysis (seaborn.heatmap()).
Note : Available only with six months duration training of Data Analytics.

Module 12 - Exploratory Data Analysis (EDA) : Introduction to Exploratory Data Analysis, Identifying Outliers and Anomalies, Detecting Outliers Using: Z-Score Method (scipy.stats.zscore()), IQR Method (Box Plot Analysis), Visualization (sns.boxplot(), sns.violinplot()); Handling Outliers: Winsorization (winsorize()), Log Transformations, Capping & Floor Techniques; Feature Engineering and Data Preprocessing - Handling Missing Data (df.isnull().sum(), fillna(), dropna()), Creating New Features (Feature Transformation & Combination), Encoding Categorical Data (OneHotEncoder(), LabelEncoder()); Feature Selection Techniques: Variance Threshold (sklearn.feature_selection.VarianceThreshold()), Correlation-Based Selection (SelectKBest()); Data Scaling and Normalization - Why is Feature Scaling Important?, Min-Max Scaling (MinMaxScaler()), Standardization (StandardScaler()), Log Transformations (np.log()), Power Transformations (yeo-johnson, box-cox).
Note : Available only with six months duration training of Data Analytics.

Module 13 - Introduction to Machine Learning : Definition of Machine Learning, How Machines Learn from Data, The Lifecycle of a Machine Learning Model, Understanding Features, Labels, and Target Variables, Difference Between AI, ML, and Deep Learning, When to Use Traditional ML vs. Deep Learning?, Applications of Machine Learning in the Real World, Identify how ML models are used in Google, Netflix, Amazon, and Facebook; Types of Machine Learning, What is Supervised Learning?, Characteristics of Labeled Data, Types of Supervised Learning – Regression (Predicting Continuous Values), Classification (Predicting Categorical Values); Common Supervised Learning Algorithms - Linear Regression (sklearn.linear_model.LinearRegression), Logistic Regression (sklearn.linear_model.LogisticRegression), Decision Trees (sklearn.tree.DecisionTreeClassifier), Random Forest (sklearn.ensemble.RandomForestClassifier); Real-World Applications of Supervised Learning - Spam Email Classification, Loan Approval Prediction, Disease Diagnosis in Healthcare; What is Unsupervised Learning?, Characteristics of Unlabeled Data, Types of Unsupervised Learning – Clustering (Grouping Similar Data Points), Dimensionality Reduction (Reducing Complexity (e.g., PCA, t-SNE)), Common Unsupervised Learning Algorithms - K-Means Clustering (sklearn.cluster.KMeans), Hierarchical Clustering (scipy.cluster.hierarchy), Principal Component Analysis (PCA) (sklearn.decomposition.PCA); Real-World Applications of Unsupervised Learning: Customer Segmentation in Marketing, Anomaly Detection in Cybersecurity, Topic

Modeling in Natural Language Processing; Supervised vs. Unsupervised Learning, Challenges and Limitations of Each Approach; Implementing a Basic ML Model Using Scikit-Learn - Introduction to Scikit-Learn (sklearn), Installing Required Libraries: pip install numpy pandas matplotlib seaborn scikit-learn, Understanding Dataset Structure (Features & Labels), Splitting Data into Training and Testing Sets (train_test_split), Training an ML Model (model.fit()), Making Predictions (model.predict()), Evaluating Model Performance (accuracy_score, confusion_matrix).

Note : Available only with six months duration training of Data Analytics.

Module 14 – Regression & Classification : Introduction to Regression for Predictive Analytics, Types of Regression and Their Use Cases, Linear Regression - Understanding the Line of Best Fit, Simple Linear Regression (sklearn.linear_model.LinearRegression), Evaluating Regression Models (R^2 Score, MSE, RMSE), Visualizing Regression Line (seaborn.regplot); Multiple Regression - Extending Linear Regression to Multiple Features, Feature Selection & Engineering (SelectKBest, Recursive Feature Elimination), Handling Multicollinearity (VIF Analysis); Polynomial Regression - Introduction to Non-Linear Relationships, Implementing Polynomial Regression (PolynomialFeatures), Choosing the Right Polynomial Degree; Introduction to Classification, Understanding Binary vs. Multi-Class Classification, How Classification Works in ML; Logistic Regression - Difference Between Logistic Regression and Linear Regression, Sigmoid Function and Decision Boundaries, Implementing Logistic Regression (sklearn.linear_model.LogisticRegression), Handling Class Imbalance (SMOTE, Class Weights); Decision Trees and Random Forest - Understanding Decision Trees, Gini Index vs. Entropy in Splitting, Implementing Decision Trees (sklearn.tree.DecisionTreeClassifier), Overfitting in Decision Trees and Pruning Techniques, Introduction to Random Forest Classifier, Implementing Ensemble Learning (Bagging, Boosting); K-Nearest Neighbors (KNN) - Understanding Instance-Based Learning, Implementing KNN (sklearn.neighbors.KNeighborsClassifier), Choosing the Best Value for K, Distance Metrics in KNN (Euclidean, Manhattan, Minkowski).

Note : Available only with six months duration training of Data Analytics.

Module 15 – Clustering, Model Evaluation and Optimization : Introduction to Clustering, Why is it Important?; K-Means Clustering - Understanding Centroids and Inertia, Implementing K-Means (sklearn.cluster.KMeans), Choosing the Right Number of Clusters (Elbow Method, Silhouette Score), Applications: Customer Segmentation, Market Analysis; Hierarchical Clustering - Understanding Agglomerative and Divisive Clustering, Implementing Hierarchical Clustering (scipy.cluster.hierarchy), Dendrograms and How to Interpret Them, Applications: Document Clustering, Fraud Detection; Model Evaluation and Optimization, Train-Test Split - Splitting Data into Training and Testing Sets (train_test_split), Understanding Bias-Variance Tradeoff,

Avoiding Data Leakage; Cross-Validation - Understanding K-Fold Cross-Validation, Implementing Stratified K-Fold Cross-Validation (sklearn.model_selection.StratifiedKFold), Leave-One-Out Cross-Validation (LOO-CV); Evaluating Model Performance - Confusion Matrix (sklearn.metrics.confusion_matrix), Accuracy Score, Precision, Recall, F1-Score, ROC Curve & AUC Score (roc_auc_score), Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE).

Note : Available only with six months duration training of Data Analytics.

LEARN 2 EARN
LABS

How to Join the Data Analytics using Python Training Program

At **Learn2Earn Labs Training Institute**, we believe in nurturing talent and shaping the future of aspiring data analysts. To ensure the best outcomes for our students, we welcome applications from individuals who meet the following criteria:

Eligibility Requirements

- **Educational Qualification**: A degree in a relevant domain (e.g., Computer Science & Engineering, Business Administration, Mathematics, Statistics or any other related degree programs).
- **Passion for Learning**: Candidates must demonstrate a strong zeal to become a successful data analyst.
- **Growth-Oriented Mindset**: We are looking for individuals who are eager to learn, adapt to new challenges, and continuously improve their skills.
- **Hard Work and Dedication**: This program is rigorous and requires commitment. Candidates must be ready to put in the effort to achieve their career goals.

How to Apply

1. Visit our website page of Data Analytics using Python Training Program at **https://learntoearnlabs.com/data-analytics-course-using-python/** and read the complete details about the training program.
2. Under the **Apply Now** section, fill out the form with your details.
3. Based on your submitted details, a Learn2Earn Labs representative will contact you to further process your application or query.
4. Complete any additional steps, such as interviews, assessments, or telephonic discussions, as requested by the institute representative via email or WhatsApp.

What Happens Next?

- Once your details are reviewed, eligible candidates will be contacted for the next steps, which may include an introductory session or discussion.
- After successful enrollment, you will receive all the program details, including the schedule, enrollment ID, batch ID, terms and conditions, etc., through an enrollment confirmation letter.

Take the first step toward transforming your career! If you meet the eligibility criteria and are passionate about becoming a professional data analyst, **this program is designed for you**. 😊

Join us today and pave the way to a successful career in data analytics!

LEARN 2 EARN LABS

Best Training Institute for Graduates

Live Online & Offline Classes Available

Join us for Better Career and Package Guarantee

Top Edge Training Programs

- Full Stack Web Development
- Mobile App Development
- Front End Development
- Back End Development
- Java Full Stack
- Data Science & ML
- Digital Marketing

Guaranteed Package (In Writing)*

- **3-5 Lakhs** (With 6 Months Training Programs)
- **5-8 Lakhs** (With 12 Months Training Programs)
- **8 Lakhs+** (With 2 Years Training Programs)

* Terms & Conditions Applied

Amenities

- Digital Notes
- Live Training Sessions
- Project Assistance
- Interview Preparation
- Working Experience
- Job Recommendations
- Professional Development
- Digital Resume & Portfolio

LEARN-2-EARN LABS TRAINING INSTITUTE

Anna Icon Complex, near Kargil Petrol Pump, Sikandra-Bodla Road, Sikandra, Agra
Website : www.LearntoearnLabs.com
Call : 91-9548868337

LEARN 2 EARN LABS

Guiding Careers with Visionary Support

Institute Director(s)

Mr. Mohit Singh
M.Tech, B.Tech (C.S.E)

Mr. Mohit Singh is a professional full-stack trainer, project consultant and startup mentor. He is holding expertise in Java, Application Design, MERN Stack, DevOps, Design Thinking and User Experience Design.
He has trained thousands of students & hundreds of employed professionals. He completed his trainings in Google, Gurugram and short term projects in IIT Delhi, IIT BHU & IIT Jodhpur.
He is also recognized as Mentor with startup India (MAARG), Punjab Startup, startup Uttarakhand, Mumbai State Innovation Society, Atal Innovation Mission, etc. in the area of education & utility services.

https://www.linkedin.com/in/mohit9pages/

Dr. Shubhendra Gupta
Phd, B.Ed, M.Sc (Physics)

Dr. Shubhendra Gupta is an experienced digital marketer, Business Consultant and startup mentor with a demonstrated history of working in the education and services industry.
He use to train students & working professionals for getting better job opportunities and train business owners in generating profits or leads. His areas of interest are Digital Marketing, Business Development, Data Analysis, Strategic Planning, Market Research & Reality, User Testing, Website design, etc.
He is also recognized as Mentor with Startup Hubs & Innovation Labs in the area of education, brand building & business consultation.

https://www.linkedin.com/in/dmshubhendra/

Institute Vision

To be an institute that provides a transformative learning to produce highly skilled & competent professionals and to create leaders and innovators for society and industry.

LEARN-2-EARN LABS TRAINING INSTITUTE

F-4, First Floor, Anna Icon Complex,
near Kargil Petrol Pump, Sikandra-Bodla Road,
Sikandra, Agra, Uttar Pradesh, India

Website : www.LearntoearnLabs.com

Call : 91-9548868337