Data Analyst Syllabus

For Apprenticeship Training Program

```python
# Practical Code Example 3: Types of Data Analytics
import pandas as pd

# Load data
data = pd.read_csv('customer_data.csv')

# Descriptive Analytics
summary_statistics = data.describe()

# Diagnostic Analytics
sales_drop_reason = data[data['sales'] < data['sales

# Predictive Analytics (using scikit-learn)
from sklearn.linear_model import LinearRegression

X = data[['advertising_budget']]
y = data['sales']

model = LinearRegression()
model.fit(X, y)
future_sales = model.predict([[5000]])

# Prescriptive Analytics
optimized_pricing_strategy = data['price']
```

DATA ANALYTICS

pandas NumPy Power BI Microsoft Office EXCEL

A **Data Analyst** is a professional responsible for interpreting and transforming raw data into meaningful insights that support decision-making within organizations. By leveraging statistical techniques, analytical tools, and domain knowledge, data analysts help identify trends, patterns, and anomalies in data that might otherwise go unnoticed. The role of a data analyst typically begins with data collection from various sources such as databases, surveys, or third-party APIs. Once the data is gathered, it undergoes a cleaning and preprocessing phase to ensure consistency, accuracy, and reliability. This cleaned data is then analyzed using a combination of descriptive statistics, exploratory data analysis (EDA), and sometimes predictive modeling.

To accomplish this, data analysts utilize a wide range of tools and technologies. Excel remains a foundational tool for quick calculations and visual summaries, while Structured Query Language (SQL) is used for extracting and managing data from relational databases. For deeper analysis and automation, programming languages like Python and R are employed—often in conjunction with libraries such as Pandas, NumPy, and Matplotlib for data manipulation and visualization. Tools like Tableau, Power BI, and Looker allow analysts to present findings through interactive dashboards and reports that can be understood by non-technical stakeholders.

Data analysts play a critical role across industries including finance, healthcare, marketing, logistics, and technology, where data-driven strategies are key to competitive advantage. Whether it's optimizing business operations, tracking customer behavior, or forecasting future trends, data analysts enable organizations to act on evidence rather than intuition. With the increasing importance of big data and real-time analytics, the role continues to evolve, often overlapping with data science, business intelligence, and machine learning domains.

In today's data-driven world, the demand for skilled data analysts is rapidly growing as organizations recognize the value of making informed decisions based on concrete evidence rather than assumptions. A data analyst not only needs technical expertise but also strong business acumen and communication skills to translate complex data findings into actionable recommendations. They often collaborate closely with stakeholders, such as marketing teams, finance departments, or product managers, to understand business goals and tailor their analysis accordingly. In addition, data analysts are expected to maintain data integrity, ensure compliance with data privacy standards, and stay current with evolving technologies and methodologies.
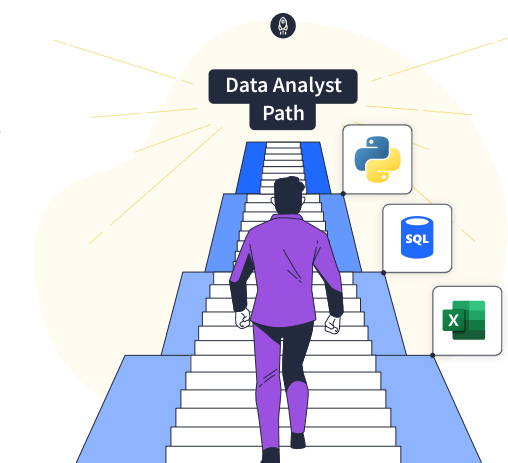
Data Analyst Path

INTRODUCTION TO DATA ANALYTICS

- Introduction to the World of Business and Data
- Relevant Terms Explained
- Data Analyst Compared to Other Data Jobs
- Data Analyst Job Description
- Why Python

SETTING UP THE ENVIRONMENT

- Programming Explained in a Few Minutes
- Jupyter - Introduction
- Jupyter - Installing Anaconda
- Jupyter - Intro to Using Jupyter
- Jupyter - Working with Notebook Files
- Jupyter - Using Shortcuts
- Jupyter - Handling Error Messages
- Jupyter - Restarting the Kernel

PYTHON BASICS

- Python Variables
- Types of Data - Numbers and Boolean Values
- Types of Data - Strings
- Introduction to Anaconda AI
- Using the Anaconda Assistant: Strings
- Basic Python Syntax - Arithmetic Operators
- Basic Python Syntax - The Double Equality Sign
- Basic Python Syntax - Reassign Values
- Basic Python Syntax - Add Comments
- Basic Python Syntax - Line Continuation
- Basic Python Syntax - Indexing Elements
- Basic Python Syntax - Indentation
- Operators - Comparison Operators
- Operators - Logical and Identity Operators
- Conditional Statements - The IF Statement
- Conditional Statements - The ELSE Statement
- Conditional Statements - The ELIF Statement
- Conditional Statements - A Note on Boolean Values
- Functions - Defining a Function in Python
- Functions - Creating a Function with a Parameter
- Functions - Another Way to Define a Function
- Functions - Using a Function in Another Function
- Functions - Combining Conditional Statements and Functions
- Functions - Creating Functions That Contain a Few Arguments
- Functions - Notable Built-in Functions in Python

- Sequences - Lists
- Sequences - Using Methods
- Sequences - List Slicing
- Sequences - Tuples
- Sequences - Dictionaries
- Iteration - For Loops
- Iteration - While Loops and Incrementing
- Iteration - While Loops and Incrementing
- Iteration - Use Conditional Statements and Loops Together
- Using the Anaconda Assistant: Several Python Tools
- Iteration - Iterating over Dictionaries
- Using the Anaconda Assistant: Dictionaries

## FUNDAMENTALS FOR CODING IN PYTHON

- Object-Oriented Programming (OOP)
- Modules, Packages, and the Python Standard Library
- Importing Modules
- Introduction to Using NumPy and pandas
- What is Software Documentation?
- The Python Documentation

MATHEMATICS FOR PYTHON

- What Is A Matrix?
- Scalars and Vectors
- Linear Algebra and Geometry
- Arrays in Python
- What Is a Tensor?
- Adding and Subtracting Matrices
- Errors When Adding Matrices
- Transpose
- Dot Product of Vectors
- Dot Product of Matrices
- Why is Linear Algebra Useful

NUMPY BASICS

- The NumPy Package and Why We Use It
- Installing/Upgrading NumPy
- Ndarray
- The NumPy Documentation
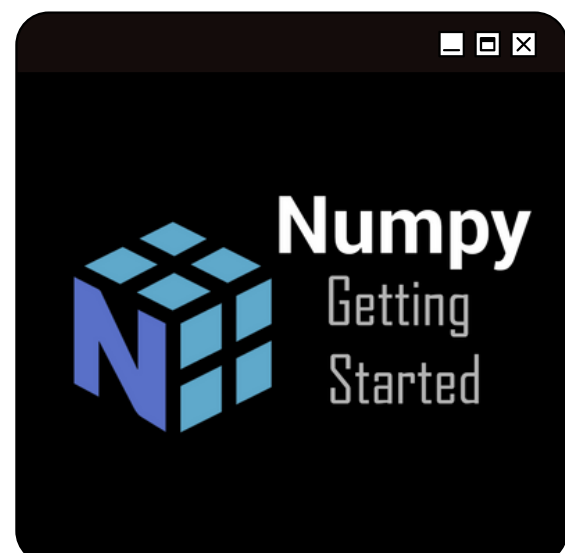
WORKING WITH TEXT FILES

- Working with Files in Python - An Introduction
- File vs File Object, Read vs Parse
- Structured vs Semi-Stuctured and Unstuctured Data
- Data Connectivity through Text Files
- Principles of Importing Data in Python
- More on Text Files (*.txt vs *.csv)
- Fixed-width Files
- Common Naming Conventions Used in Programming
- Importing Text Files in Python (open())
- Importing Text Files in Python (with open())
- Importing *.csv Files with pandas - Part I
- Importing *.csv Files with pandas - Part II
- Importing *.csv Files with pandas - Part III
- Importing Data with the "index_col" Parameter
- Importing Data with NumPy - .loadtxt() vs genfromtxt()
- Importing Data with NumPy - Partial Cleaning While Importing
- Importing *.json Files
- Prelude to Working with Excel Files in Python
- Working with Excel Data (the *.xlsx Format)
- An Important Exercise on Importing Data in Python
- Importing Data with the pandas' "Squeeze" Method
- A Note on Importing Files in Jupyter
- Saving Your Data with pandas
- Saving Your Data with NumPy - np.save()
- Saving Your Data with NumPy - np.savez()
- Saving Your Data with NumPy - np.savetxt()
- Working with Text Files - Conclusion

WORKING WITH TEXT DATA

- Working with Text Data and Argument Specifiers
- Manipulating Python Strings
- Using Various Python String Methods - Part I
- Using Various Python String Methods - Part II
- String Accessors
- Using the .format() Method
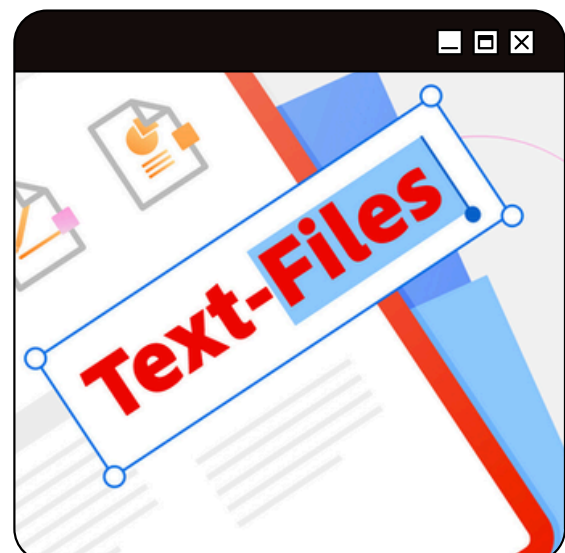
MUST-KNOW PYTHON TOOLS

- Iterating Over Range Objects
- Nested For Loops - Introduction
- Triple Nested For Loops
- List Comprehensions
- Anonymous (Lambda) Functions

DATA GATHERING/DATA COLLECTION

- What is data gathering/data collection?

API'S (POST REQUESTS ARE NOT NEEDED FOR THIS COURSE)

- Overview of APIs
- GET and POST Requests
- Data Exchange Format for APIs: JSON
- Introducing the Exchange Rates API
- Including Parameters in a GET Request
- More Functionalities of the Exchange Rates API
- Coding a Simple Currency Conversion Calculator
- iTunes API
- iTunes API: Structuring and Exporting the Data
- Pagination: GitHub API

## DATA CLEANING AND DATA PROCESSING

- Data Cleaning and Data Preprocessing

PANDAS SERIES

- .unique(), .nunique()
- Converting Series into Arrays
- .sort_values()
- Attribute and Method Chaining
- .sort_index()

PANDAS DATAFRAMES

- A Revision to pandas DataFrames
- Using the Anaconda Assistant: Importing Data with pandas
- Common Attributes for Working with DataFrames
- Data Selection in pandas DataFrames
- Data Selection - Indexing with .iloc
- Data Selection - Indexing with .loc
- A Few Comments on Using .loc and .iloc

Pandas-Data frame
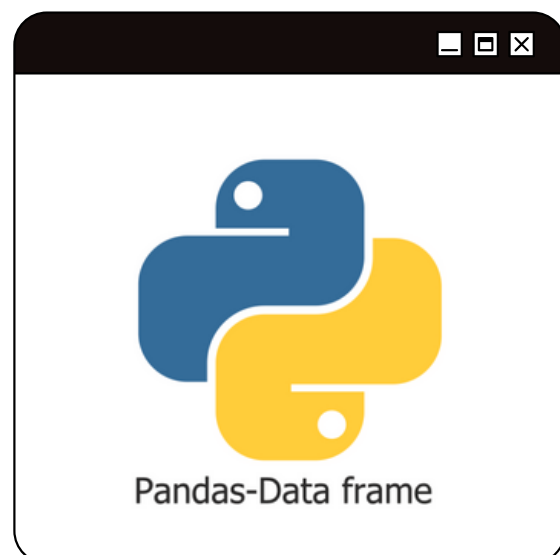
NUMPY FUNDAMENTALS

- Indexing in NumPy
- Assigning Values in NumPy
- Elementwise Properties of Arrays
- Types of Data Supported by NumPy
- Characteristics of NumPy Functions Part 1
- Characteristics of NumPy Functions Part 2

NUMPY DATATYPES

- ndarrays
- Arrays vs Lists
- Strings vs Object vs Number

WORKING WITH ARRAYS

- Basic Slicing in NumPy
- Stepwise Slicing in NumPy
- Conditional Slicing in NumPy
- Dimensions and the Squeeze Function

GENERATING DATA WITH NUMPY

- Arrays of 0s and 1s
- "_like" functions in NumPy
- A Non-Random Sequence of Numbers
- Random Generators and Seeds
- Basic Random Functions in NumPy
- Probability Distributions in NumPy
- Applications of Random Data in NumPy

STATISTICS WITH NUMPY

- Using Statistical Functions in NumPy
- Minimal and Maximal Values in NumPy
- Statistical Order Functions in NumPy
- Averages and Variance in NumPy
- Covariance and Correlation in NumPy
- Histograms in NumPy (Part 1)
- Histograms in NumPy (Part 2)
- NAN Equivalent Functions in NumPy

NUMPY - PREPROCESSING

- Checking for Missing Values in Ndarrays

Learn

Statistics
NumPy
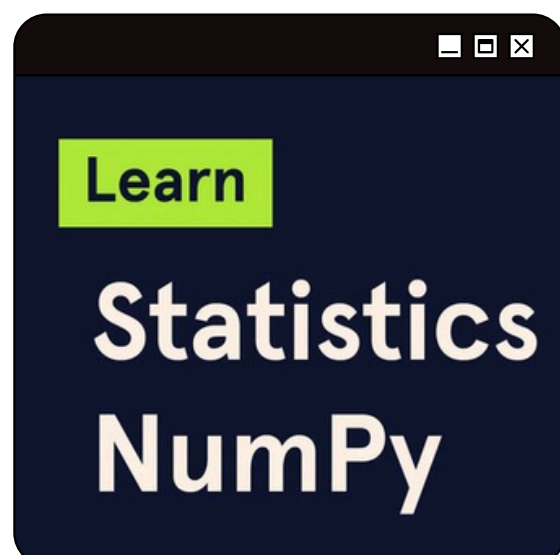
- Substituting Missing Values in Ndarrays
- Reshaping Ndarrays
- Removing Values from Ndarrays
- Sorting Ndarrays
- Argument Sort in NumPy
- Argument Where in NumPy
- Shuffling Ndarrays
- Casting Ndarrays
- Striping Values from Ndarrays
- Stacking Ndarrays
- Concatenating Ndarrays
- Finding Unique Values in Ndarrays

A LOAN DATA EXAMPLE WITH NUMPY

- Setting Up: Introduction to the Practical Example
- Setting Up: Importing the Data Set
- Setting Up: Checking for Incomplete Data
- Setting Up: Splitting the Dataset
- Setting Up: Creating Checkpoints
- Manipulating Text Data: Issue Date
- Manipulating Text Data: Loan Status and Term
- Manipulating Text Data: Grade and Sub Grade
- Manipulating Text Data: Verification Status & URL
- Manipulating Text Data: State Address
- Manipulating Text Data: Converting Strings and Creating a Checkpoint
- Manipulating Numeric Data: Substitute Filler Values
- Manipulating Numeric Data: Currency Change – The Exchange Rate
- Manipulating Numeric Data: Currency Change - From USD to EUR
- Completing the Dataset

THE "ABSENTEEISM" EXERCISE - INTRODUCTION

- An Introduction to the "Absenteeism" Exercise
- The "Absenteeism" Exercise from a Business Perspective
- The Dataset

SOLUTION TO THE "ABSENTEEISM" EXERCISE

- How to Complete the Absenteeism Exercise
- Eyeball Your Data First
- Note: Programming vs the Rest of the World

ABSENTEEISM RATE MONTHLY

Month	Number of Lost Workdays Due to Absences	Average Number of Employees During the Month	Average Number of Workdays in the Month	Absentee Rate
January	10	60	23	0.7%
February	25	59	21	2.0%
March	30	58	21	2.5%
April	25	58	22	2.0%
May	26	60	23	1.9%
June	30	58	20	2.6%
July	38	60	23	2.8%
August	40	58	22	3.1%
September	50	60	21	4.0%
October	38	60	23	2.8%
November	25	60	21	2.0%
December	55	60	22	4.2%
Annual Absenteeism Rate				2.5%

- Using a Statistical Approach to Solve Our Exercise
- Dropping the 'ID' Column
- Analysis of the 'Reason for Absence' Column
- Splitting the Reasons for Absence into Multiple Dummy Variables
- Working with Dummy Variables - A Statistical Perspective
- Grouping the Reason for Absence Columns
- Concatenating Columns in a pandas DataFrame
- Reordering Columns in a DataFrame
- Working on the 'Date' Column
- Extracting the Month Value from the 'Date' Column
- Creating the 'Day of the Week' Column
- Understanding the Meaning of 5 More Columns
- Modifying the 'Education' Column
- Final Remarks on the Absenteeism Exercise

DATA VISUALIZATION

- What Is Data Visualization and Why Is It Important?
- Why Learn Data Visualization?
- Choosing the Right Visualization – What Are Some Popular Approaches and Framewor
- Introduction into Colors and Color Theory
- Bar Chart - Introduction - General Theory and Getting to Know the Dataset
- Bar Chart - How to Create a Bar Chart Using Python
- Bar Chart – Interpreting the Bar Graph. How to Make a Good Bar Graph
- Pie Chart - Introduction - General Theory and Dataset
- Pie Chart - How to Create a Pie Chart Using Python
- Pie Chart – Interpreting the Pie Chart
- Pie Chart - Why You Should Never Create a Pie Graph
- Stacked Area Chart - Introduction - General Theory. Getting to Know the Dataset
- Stacked Area Chart - How to Create a Stacked Area Chart Using Python
- Stacked Area Chart - Interpreting the Stacked Area Graph
- Stacked Area Chart - How to Make a Good Stacked Area Chart
- Line Chart - Introduction - General Theory. Getting to Know the Dataset
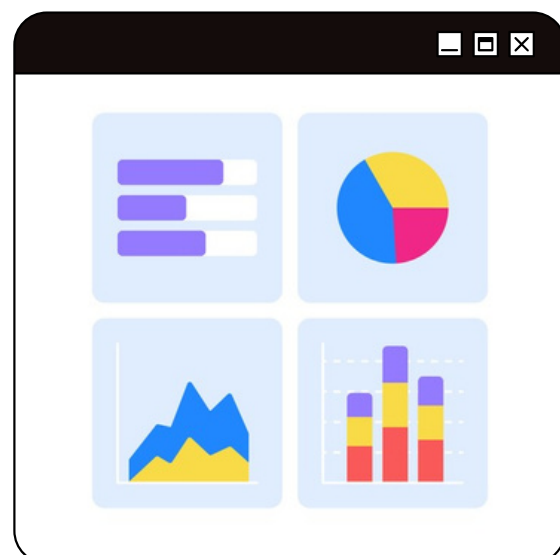
- Line Chart - How to Create a Line Chart in Python
- Line Chart - Interpretation
- Line Chart - How to Make a Good Line Chart
- Histogram - Introduction - General Theory. Getting to Know the Dataset
- Histogram - How to Create a Histogram Using Python
- Histogram – Interpreting the Histogram
- Histogram – Choosing the Number of Bins in a Histogram
- Histogram - How to Make a Good Histogram
- Scatter Plot - Introduction - General Theory. Getting to Know the Dataset
- Scatter Plot - How to Create a Scatter Plot Using Python
- Scatter Plot – Interpreting the Scatter Plot
- Scatter Plot - How to Make a Good Scatter Plot
- Regression Plot - Introduction - General Theory. Getting to Know the Dataset
- Regression Plot - How to Create a Regression Scatter Plot Using Python
- Regression Plot – Interpreting the Regression Scatter Plot
- Regression Plot - How to Make a Good Regression Plot
- Bar and Line Chart - Introduction - General Theory. Getting to Know the Dataset
- Bar and Line Chart - How to Create a Combination Bar and Line Graph Using Python
- Bar and Line Chart – Interpreting the Combination Bar and Line Graph
- Bar and Line Chart – How to Make a Good Bar and Line Graph
- Data Visualization - Exercise

POWER BI – FROM BASICS TO DASHBOARDING

Objective: To develop practical skills in data visualization, data modeling, and dashboard creation using Power BI.

Power BI Basics
- Introduction to Power BI and its ecosystem
- Understanding the Power BI Interface (Desktop & Service)
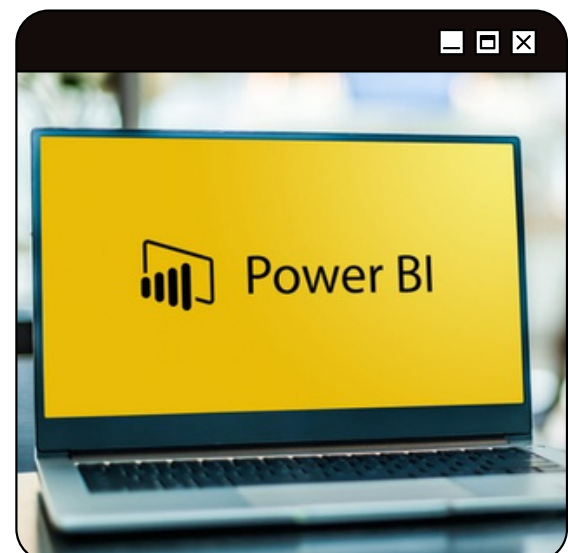
Creating and formatting a table visualisation

- Welcome to Part 1: Visualizations
- Importing from Excel, and Creating our first visualization
- Exploring Power BI Desktop - Report view
- Exploring Power BI Desktop - Data view
- Focus mode, Different visualizations and Dark mode
- Why do I need a Work email address? And how can I get one, if I don't have it?
- Saving visualization to the Desktop and to the Power BI service

Formatting our first visualization

- Opening the Format Pane - for older versions of Power BI
- Exploring the Format Pane (for all versions of Power BI)
- Formatting using On-object Interaction
- Formatting font and font size
- Formatting colors
- Stylistic options
- Position visuals
- Align visuals
- Format Painter
- Configuring summarization, both default and in a specific visualization
- Changing number and date formatting
- Custom number and date formatting
- Visual level formatting

Creating different visualizations:
Matrices and bar charts

- Matrix
- Drill down data, see data and records, and export data
- Stacked bar charts and switch theme for reports
- Bar Chart formatting, including continuous versus categorical axes
- Configure interactions between visual (Edit interactions)
- Clustered and 100% Stacked bar charts
- Line and area charts, including 8b. Configure duplicate pages
- Combo charts (Line and column charts)
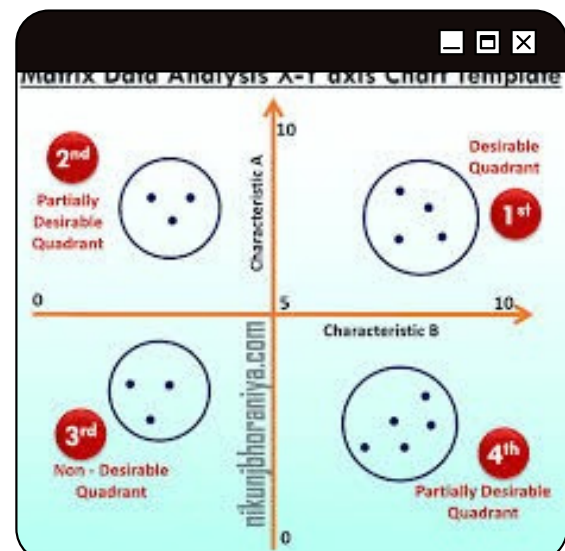
Adding more control to your visualizations
- Adding Text boxes, Images and Shapes
- Visual level, page level and report level filters - basic filters
- Advanced Filtering
- Filter Top N Items
- Slicer
- Synchronizing slicers to multiple pages
- Slicer Warning
- Sort visuals
- Configure small multiples
- Use Bookmarks for reports
- Group and layer visuals by using the Selection pane
- Drillthrough
- Buttons and Actions
- Page Navigation and Drill through actions
- Enable Natural Language Queries (Ask A Question) and Page Formatting
- Tooltip Pages
- Page and Bookmark Navigator

Other visualizations
- Ribbon charts
- Waterfall charts
- Scatter, bubble and dot charts
- Pie charts and donut charts
- Treemaps
- Funnel charts
- Import a Custom Visual - not needed from 21 October 2024

Mapping
- Enabling maps
- Maps
- Formatting maps
- Adding Data Categories
- Filled Maps, Conditional Formatting, color blindness
- Creating hierarchies
- ArcGIS Maps for Power BI

Measure performance by using KPIs, gauges and cards
- Gauges
- Cards and Multi-row cards
- More conditional formatting
- KPIs
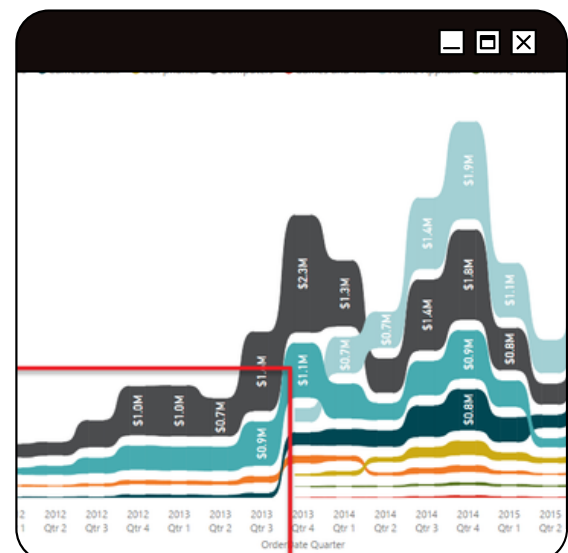
Identify patterns and trends and Other Visualization Items

- Define quick measures
- Export report data
- Create reference lines by using Analytics pane, including the Forecast feature
- Use error bars
- Identify outliers
- Use clustering
- Use Anomaly Detection
- Add a (Smart) Narrative visual
- Use groupings and binnings
- Use the AI Visual Key Influencers to explore dimensional variances
- Use the Analyze feature in Power BI
- Use the AI Visual decomposition tree visual to break down a measure
- Design and configure for accessibility
- Creating a custom theme
- Creating a paginated report
- Exploring Power BI Report Builder

Visualize and Analyze the Data

- R and Python Visualizations
- Use or create a PBIDS file

Get and Transform Data

- Get and Transform Data
- let's Get some more Data
- Exploring the Power Query Editor interface
- Introducing the M language
- Let's start look at the Home tab
- Home menu - Manage Columns
- Home menu - Reduce Rows and Use First Row as Headers

Get Data - Home

- Sort and Filter
- Split Column

Getting Multiple files

- Merge Queries and Expand Table
- Merge Queries with Group By, and different types of Joins
- Appending two queries together
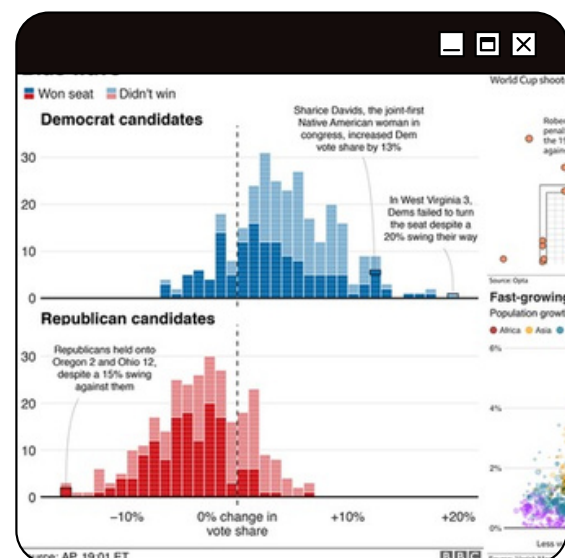
- Appending three or more queries together + resolving a problem with data types
- Combine Files (getting information from a folder)

Transform Menu
- Transform - Table and Any Column
- Pivot Column
- Unpivot
- Unpivot in conjunction with other Transform features

Transform - Text and Numbers
- Transform/Add Column - Text - Format
- Transform/Add Column - Text - Merge Columns
- Transform/Add Column - Text - Extract
- Transform/Add Column - Text - Parse
- Transform/Add Column - Number Column - Statistics and Standard
- Transform/Add Column - Other Number Column functions
- Transform/Add Column - Text and Numbers

Transform - Dates and Time
- Creating a list of dates
- Transform/Add Column - Date
- Transform/Add Column - Dates in other cultures/languages
- Transform/Add Column - Time
- Transform/Add column - Duration

Add Columns, View and Help Menus
- Column from examples - dates and numbers
- Column from examples - text
- Importing CSV/text files and Extract Tables Using Examples
- Conditional Column and Convert semi-structured data to a table
- Resolving Errors from Conditional Columns
- Index Column and Duplicate Column
- Custom Column - If Then Else
- Converting text from a different locale to a number
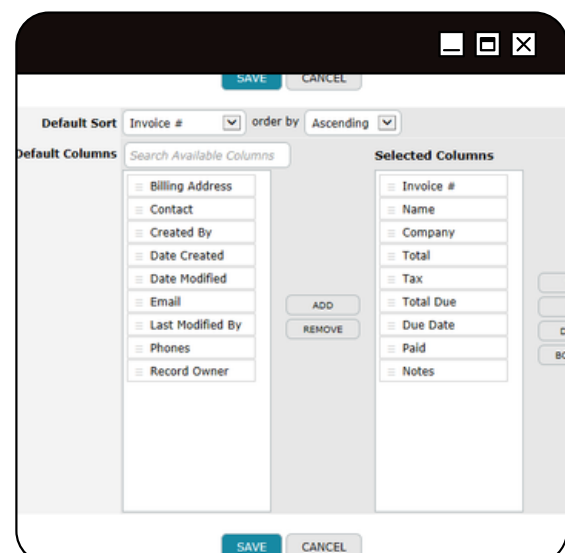
View and Help menus and advanced functionality

- Other M Functions
- View and Help menus, including Column Properties
- Profile the data
- Advanced Editor
- Functions and Parameters
- DateTimeZone date type and M functions
- Worked Practice Activity Number 16 - Dividing Annual data into Months

Get other types of data

- Installing SQL Server
- Introduction to SQL Server
- Importing database data into Power BI, and Query Folding
- Expanding multiple tables in SQL Server
- Select a storage mode
- Importing data from SQL Server Analysis Services (SSAS)
- Setting up Azure SQL Database
- Using Azure SQL Database in Power BI
- Automatic page refresh
- Use the Microsoft Dataverse
- Configure data loading

Get and Transform

- Using Big Data
- Identify query performance issues, including Query Diagnostics
- Apply AI Insights

Creating a Data Model

- Modeling and DAX functions
- Get multiple data sets, and connecting them together
- The problems with direction of relationships between data sets

- Creating a Data Model

An introduction to DAX functions, including Logical functions
- DAX functions - A useful Resource
- Calculated columns - an introduction
- Basic operators
- IF, BLANK, ISBLANK
- AND, OR and NOT
- SWITCH
- Other functions
- An introduction to DAX functions

Statistical functions (aggregation and iterator functions)
- Measures - an introduction, with standard aggregations including Countblank
- Aggregation of calculations (iterator functions)
- Other statistical functions

Mathematical functions
- Rounding functions
- Division functions - MOD and QUOTIENT
- SIGN (and use with SWITCH) and ABS
- Exponential functions
- Other mathematical functions

Text functions
- Text searching
- Text extraction and substitution
- Text conversion
- Other text functions

Information Functions
- ISERROR and LOOKUPVALUE
- Other information functions

Filter and Value Functions
- Flatten out a parent-child hierarchy
- Design a data model that uses a star schema
- RELATEDTABLE and COUNTROWS
- Context
- ALL
- FILTER
- CALCULATE
- ALLEXCEPT
- ALLSELECTED

- Other filter and value functions

Time Intelligence Functions
- Date and Time Functions
- FIRSTDATE, LASTDATE
- Start of... and End of...
- Previous... and Next...
- DATESINPERIOD
- DATESMTD, DATESQTD, DATESYTD, TOTALMTD, TOTALQTD, TOTALYTD
- Opening Balance and Closing Balance
- Semi-additive Measures
- SAMEPERIODLASTYEAR and PARALLELPERIOD
- Other Time Intelligence Functions

## Creating Date tables, calculation groups, visual calculations
- Create calculated tables
- Create a common date table
- Creating a Time Intelligence model
- Creating a Calculation Group and Items
- Expanding our Calculation Group and Items, and adding a filter
- Creating a basic Visual Calculation
- Visual Calculations: Moving Average and Running Total
- Other Visual Calculations

Other Modeling and DAX Topics
- Define role-playing dimensions
- Resolve many-to-many relationships - Joint Bank Accounts
- Resolve many-to-many relationships - Different types of granularity
- Improve cardinality levels through summarization and by changing data types
- Identify poorly performing measures, relationships, and visuals
- Using DAX Query View

An Introduction to the Power BI Service
- The Power BI Service
- Logging into Power BI Service and a quick look around
- Power BI Terminology: data sources, semantic models, reports and dashboard

INTRODUCTION TO POWER BI SERVICE
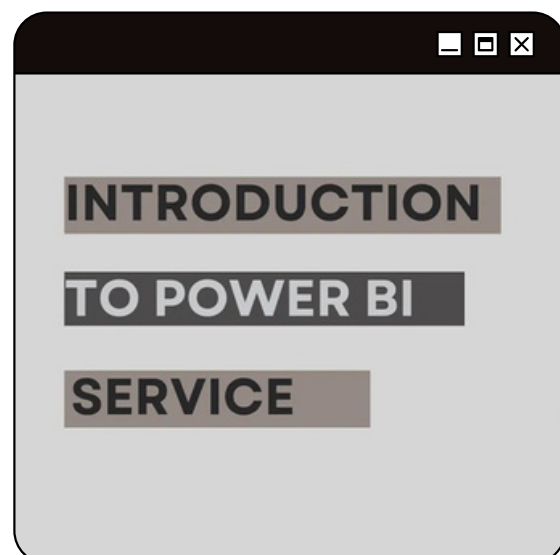
- Power BI Terminology: workspaces and apps
- Using Semantic models (Datasets) in the Power BI Service
- Using Reports in the Power BI Service
- Importing Your Data as a Semantic Model (Dataset)
- Other ways to access semantic models
- The Navigation Pane, including Add a Quick Insights and Lineage

Power BI Pro, adding users and sharing reports and visuals

- Signing up for Power BI Pro with additional users
- Creating a new report
- Sharing my new report
- Sharing reports and visuals to PowerPoint
- Using exported reports and visuals in PowerPoint

Row Level Security

- Adding role-based Row Level Security
- Adding Dynamic Row Level Security
- Testing Dynamic Row Level Security in the Power BI Service

Dashboards

- Differences between dashboards and reports
- Manage Tiles on a Dashboard and other Tiles options
- Dashboards: Options including Set Mobile View
- Configure Subscriptions
- Pin a Live Report Page to a Dashboard
- Use the Q&A Feature
- Add a Dashboard Theme
- Apply or Change Sensitivity Labels
- Configure Data Alerts

Manage semantic models

- Analyze in Excel
- Connecting to a Power BI semantic model in Excel
- Promote or Certify (Endorse) a semantic model
- Data Gateways; Providing Access to Semantic models
- Configure a Semantic Model Scheduled Refresh
- Configure Incremental Refresh Settings - Step 1
- Configure Incremental Refresh Settings - Step 2

Microsoft Excel

Analyze Data
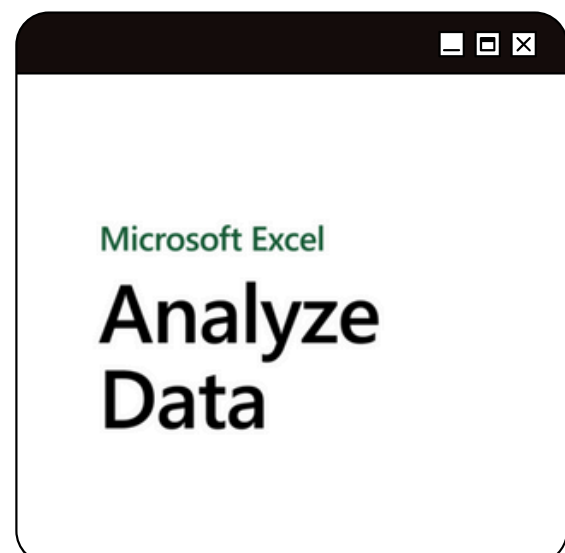
Create and manage workspaces, and publish items
- Create and Configure a Workspace
- Assign Workspace Roles
- Providing Access to Semantic Models
- Creating a Workspace App
- Updating a Workspace App
- Promote or certify Power BI report or app
- Publish, Import or Update Assets in a Workspace - Publish securely
- Publish reports on the web, so the public can see
- Personalize visuals

Scorecards and metrics
- Creating Scorecards and Metrics - not needed from 21 October 2024
- Sharing Scorecards and Metrics - not needed from 21 October 2024
- Using Scorecards and Metrics - not needed from 21 October 2024

Not needed for the PL-300 exam
- Drillthrough from another report
- Export Data
- Dataflow Scheduled Refresh and Endorsement
- Dashboard Data Classifications
- Use or create a dataflow

Integrated COVID-19 Dashboard (Built for Real-World Use, Not PL-300)
- Built with Power BI + Python using API data
- Shows case trends, vaccination progress, and map-based impact
- Uses dataflows with scheduled refresh and endorsement
- Skips exam-only features like Drillthrough, focuses on real insights
- Designed for practical monitoring, not PL-300 exam
- Built using Power BI and Python
- Allows data export for Excel/CSV
- Focused on real-world insights and decisions
- Avoids exam-only features like Drillthrough
- Shows daily/weekly case trends
- Includes map visualizations for regional impact
- Tracks vaccination progress by country/region
- Enabled scheduled data refresh (auto daily updates)
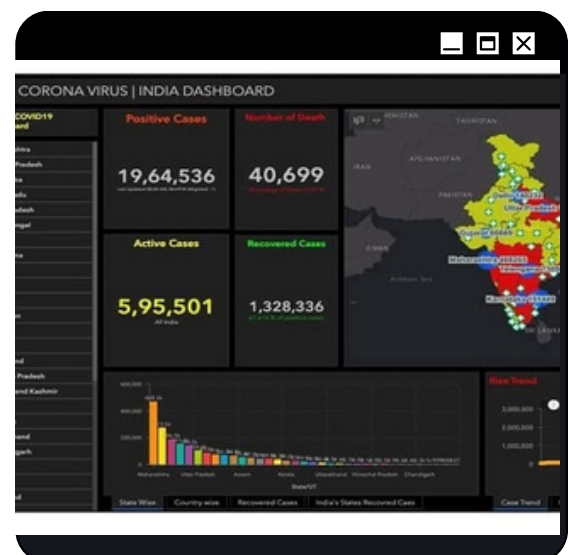

PROJECT VIEW

Microsoft Excel (Beginner to Advanced Level)

- Introduction to Excel interface, navigation, and shortcut keys
- Basic formulas: SUM(), AVERAGE(), COUNT(), ROUND(), etc.
- Data Cleaning with Excel: Removing duplicates, replacing blanks, using functions like TRIM(), CLEAN(), SUBSTITUTE()
- Lookup & Reference Functions: VLOOKUP(), HLOOKUP(), XLOOKUP(), INDEX(), MATCH(), IFERROR()
- Working with Dates and Time: TODAY(), NOW(), DATEDIF(), TEXT()
- Pivot Tables & Pivot Charts: Grouping, filtering, summarizing large data
- Conditional Formatting: Highlighting values using color codes and rules
- What-If Analysis: Goal Seek, Scenario Manager, Data Tables
- Dashboard Creation: Using charts, slicers, dynamic formulas, KPIs
- Excel + Power BI Integration: Loading Excel files into Power BI for visualisation.

SQL / MySQL for Data Analysis

- Fundamentals of databases and relational database models
- Writing Basic SQL Queries using SELECT, WHERE, ORDER BY, and LIMIT
- Filtering Techniques: LIKE, BETWEEN, IN, IS NULL, and DISTINCT
- Working with JOINs: INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL OUTER JOIN
- Grouping and Aggregating Data using GROUP BY and HAVING
- Aggregate Functions: COUNT(), SUM(), AVG(), MAX(), MIN()
- Creating and Modifying Tables: CREATE, ALTER, DROP
- Subqueries and Nested Queries for complex problems
- Indexing and Query Optimization basics
- Hands-on Practice: Use real-world datasets like Sakila, Chinook, or Northwind
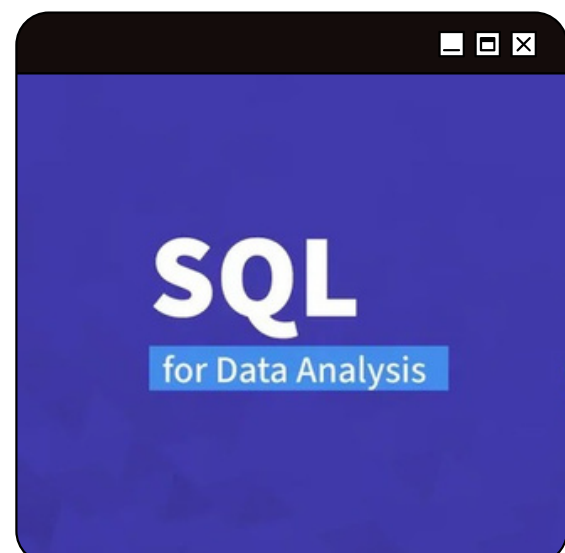
Data Science Fundamentals

- What is Data Science?
- Difference between Data Analytics, Data Science & Machine Learning
- Understanding types of Machine Learning
- Supervised vs Unsupervised
- Supervised Learning Algorithms
- Linear Regression
- Logistic Regression
- Decision Trees (Basic)
- Unsupervised Learnin
- K-Means Clustering
- Using Scikit-learn (sklearn) to implement basic ML models
- Model Evaluation Techniques
- Accuracy Score, Confusion Matrix
- Hands-on Mini Projects

Resume Building, Soft Skills & Interview Preparation

- Building a professional Data Analyst Resume with keywords and achievements
- Creating a LinkedIn profile that attracts recruiters
- Setting up a GitHub portfolio to showcase projects and notebooks
- Writing a Cover Letter that highlights skills and passion
- **Soft Skills Training:** Email writing, corporate communication, teamwork
- Common HR Interview Questions with sample answers
- Technical Interview Preparation
- Python Coding Challenges
- SQL Query Practice
- Excel Problem Solving
- Power BI Scenario-based Questions
- Conducting mock interviews and mock HR rounds
- Learn to analyze JDs to tailor resumes and prepare for company-specific interviews
- Helps in filtering fake/inappropriate job posts and preparing smartly
- Learn to solve and present business problems with logic and data
- Posting regular content on LinkedIn: Project breakdowns, learnings, achievements

DATA ANALYST APPRENTICESHIP
TRAINING SYLLABUS PROGRESS TRACKER

NOTES

HINDTECH
IT SOLUTIONS

Training
Data Analyst Apprenticeship Training Program

Duration
6 to 8 Months

Amount
₹25,000

- www.hindtechitsolutions.com
- www.hindtechlearningpoint.com

- hr@hindtechitsolutions.com
- hindtechlearningpointlko@gmail.com

- +91 7905320279
- +91 6307738600

Scan Me:

Building No 10/703, Ground Floor, near Arvindo Park Road, Sector 10, Indira Nagar, Lucknow, Uttar Pradesh 226016

Branch 1

PARMAR PLAZA, 1st Floor, Bakshi Ka Talab, Bargadi Magath, Uttar Pradesh 226201

Branch 2

Scan this QR code for details

HINDTECH
LEARNING POINT

Skill India
कौशल भारत-कुशल भारत

N·S·D·C National Skill Development Corporation
Transforming the skill landscape