[REGEX]*

Software Services. Learn To Create



Data Science
Specialization Program

contact@regexsoftware.com | www.regexsoftware.com

ABOUT THE PROGRAM:

At **REGex Software**, we take pride in being a leading Data Science Training Company, offering top-notch education and career support. With increasing demand for data scientists, this field offers some of the most lucrative career opportunities.

Join the best Data Science course and take the first step towards a successful career. Limited seats available – enroll today!

Our course stands out among the top Data Science programs, offering expert-led training in **Python**, **Machine Learning, Deep Learning, Generative AI, Agentic AI, Prompt Engineering, Statistics, SQL, Data Visualization**, and **Power BI**. Learners also gain hands-on experience through real-world projects, case studies, and industry-standard tools, helping them build practical skills and gain a competitive edge in today's data-driven job market.



MODE:
Physical (Jaipur And Ahmedabad) or Online (Google Meet)

DURATION:
10 Months
+ 6 Months Support

PARTICIPANTS:
18 – 20 per Batch

What you will Learn ?

- Python
- Machine Learning
- Deep Learning
- Generative AI
- Agentic AI
- Prompt Engineering
- Statistics
- SQL
- Data Visualization
- Power BI

BUILD YOUR FUTURE WITH

DATA SCIENCE

Learn. Apply. Get Placed.

Study Material

- E-Books
- Lecture Recordings
- Interview Questions
- Hands-on Exercises & Projects
- Notes & Practice Sheets
- Real-time Projects & Case Studies
- Doubt Support
- Slack / Community Access

Output

- Jobs in tech and non-tech
- Become expert in multiple technology domains like Python, Statistics, Machine Learning, Deep Learning, NLP, GenAI, Data Visualization, and more
- End-to-end live projects
- Portfolio building with real datasets
- Get guidance for interviews
- Gain in-demand skills for top MNCs & product companies





DATA SCIENCE

TURNING DATA INTO SMART DECISIONS

Package Offered So Far

	IT Candidates	Non-IT Candidates
Minimum Package	4 LPA	3 LPA
Average Package	4 – 7 LPA	3.5 – 6 LPA
Overall Highest Package	39 LPA	14.5 LPA

Industry Expert Trainers | Real-world Projects | Interview Preparation | Resume & Portfolio Building | Placement Assurance

Extra Sessions: Resume Building, Git & GitHub, Docker, AWS Basics, Job Interview Preparation, and many more for all students.

Our Students Placed / Partnership

amazon | Microsoft | TCS TATA CONSULTANCY SERVICES | Google | Deloitte. | Adobe | JPMorgan Chase & Co.

ORACLE | cisco | intel | wipro | PhonePe | zomato | Capgemini | Mindtree

PLACEMENT PROCESS:



At **REGex Software,** we are committed to providing a structured and results-driven training approach to ensure your career success.

Training & Performance Analysis:

- Your training will begin from day 1 of your joining, focusing on hands-on learning and practical implementation.
- Our team will analyze your performance based on assignments, projects and weekly assessments from the **second week onwards** and we will provide weekly feedback to help you improve.

Mandatory Criteria for Placement Opportunities:

To be eligible for placement opportunities, you must meet the following criteria:

- ✅ **80%** attendance in live training sessions.
- ✅ **80%** completion and timely submission of assignments & projects.
- ✅ **80%** attendance in assessments, including tests, mock interviews, HR interviews and group discussions.

Resume Preparation & Placement Process:

- Between **5.5 to 6 months,** our team will provide guidance on resume building and evaluate your resumes accordingly.
- After completing **75–80%** of the program, you will receive details about the placement opportunities based on your performance and company requirements.
- Placement opportunities will be provided **continuously via email, calls and WhatsApp groups**, depending on your performance.

Placement Assurance & Refund Policy *(Applicable only for Indian Students Only):*

- This is a **Placement Assured Program,** with an additional **6-month post-program assistance.**
- IT Graduates who passed out in **2026** (Regular B.Tech, BCA, MCA, M.Tech, Integrated) are assurred a minimum salary package of **4LPA** upon placement. However, for IT Graduates who passed out in **2025**, having gaps in their academics, as well as for Non-IT Graduates (graduates other than regular B.Tech, BCA, M.tech, MCA programs), the minimum guaranteed package will be **3LPA.**
- In the event that you have attended & completed at least 80% of the program, submitted and finished at least 80% of the assignments, Tests, Mock Interviews & HR Interview and still do not secure a placement then REGEX will refund your fees with a **9% Annual Interest** rate. Furthermore, **Refunds are applicable only within the first 3 days of the demo period** and solely in cases where a specific concern is raised regarding the quality of the learning experience provided. You will receive an official confirmation mail from our team on the third day at 7:30 PM, confirming the completion of your demo period. Requests for a refund of the registration amount must be submitted prior to the issuance of this email. No refund requests will be entertained after this time and Even if you discontinue the program prematurely, you are still obligated to pay the full fee to REGex.
- **Our Commitment to Your Success:**

At REGex Software, Placement Assurance = Skills + Opportunities

We equip you with industry-relevant skills and provide continuous job opportunities based on your performance. However, it is the **student's responsibility** to crack interviews and enhance their skills based on feedback.

For additional support, we offer the flexibility to rejoin previous batches to reinforce concepts and improve understanding.

We are dedicated to your career success! 🚀

COURSE CONTENT:

WEEK 1

Linux

- Introduction to Linux OS
- Basic Commands of Linux OS
- Vi editor
- Tar Archive
- User Management and permission

GitHub

- GitHub - Version control system
- Git vs GitHub
- Git commands
- Git Branches
- Branching and merging
- Git push vs pull commands

WEEK 2

C Logical

- Mathematics for programming
- Data Types
- Variables
- Variable Scope (Local & Global)
- Constants

WEEK 3

C Logical

- Operators
- Decision Making (if, if-else, switch)
- Loops (while, do-while, for)
- Basic I/O Functions

COURSE CONTENT:

WEEK 4

C Logical

- Loops Revision (While, Do-While, For)
- Leetcode questions
- practice questions
- practice

WEEK 5

C Logical

- Nested loops
- Advance patterns

WEEK 6

Python

- Introduction to Python
- Data types
- Immutable vs Mutable data types
- String formatting & slicing
- Conditional Statements
- **Python MCQ Test - 50 Questions**

WEEK 7

Python

- Introduction to looping statement
- for loop vs While loop
- String Data type
- String manipulation
- Sequential Data type
- List Vs Tuple
- Problem solving with List datatype

COURSE CONTENT:

WEEK 8

Python

- Dictionary & Sets
- In-built functions related to the sequential datatypes
- List comprehension, set and dictionary comprehension
- Problem solving with datatype
- Introduction to functions
- Inbuilt vs user-defined functions
- Type of parameter passing in functions
- Functions vs Lambda functions
- Exception Handling
- File Handling
- Generators & Iterators

https://leetcode.com/problems/two-sum/

SQL Mastery - Beginners to Advanced

- Introduction to E-R diagram
- Understanding entity and relationship
- Working with symbols
- Type of relationship and data models
- Defining Database
- Type of databases
- DBMS vs RDBMS
- Installing MySQL database
- Type of SQL statements
- Select statement

https://youtu.be/Oi0mYCAU358?si=j4CvMt7En35gDSNM

https://youtu.be/nBEAvLHYwPg?si=oeu0PIWkirf3B4op

https://youtu.be/GwcTHHPRh_M?si=XH0J4vSHbyj2U1f

https://youtu.be/WzcqLbEDWV4?si

WEEK 9

Python

- OOPS concepts
- Introduction to oops concepts
- Creating your first class & objects
- Inheritance, polymorphism, abstraction, encapsulation
- Use cases of __init__ and __main__
- Advanced topics of Python
- **Hacker rank 5 star**

https://leetcode.com/problems/two-sum/

https://www.geeksforgeeks.org/python-extract-elements-with-frequency-greater-than-k/

SQL Mastery - Beginners to Advanced

- Operators in SQL
- SQL querying statement
- String, date, number functions
- Working with aggregation functions
- Group by statement
- Where vs having clause
- Sorting data in SQL
- Advanced sorting and aggregate function

https://dev.mysql.com/doc/refman/8.4/en/string-functions.html

https://youtu.be/qaYP6VcJ9TE?si=790pcbH6FUDR_9wa

https://youtu.be/CbXlqJ_33Aw?si=ryOcIOwQOj

COURSE CONTENT:

WEEK 10

Advanced Python for Data Science NumPy

- **Arrays** – Fast, memory-efficient numerical data structures that form the foundation of all data science and ML computations.
- **Indexing & Slicing** – Quickly access and extract specific elements, rows, or columns from an array.
- **Reshape & Flatten** – Change the dimensions of data (1D ↔ 2D ↔ 3D) to match what a model or function expects.
- **Broadcasting** – Perform fast mathematical operations between arrays of different shapes without writing explicit loops.
- **Mathematical Operations** – Vectorized calculations (sum, mean, dot product) that run significantly faster than plain Python loops.

SQL Mastery - Beginners to Advanced

- Constraints in SQL
- Type of keys
- Setting up the first constraint to your tables
- Normalization, types of dependencies and anomalies
- 1NF, 2NF, 3NF and BCNF
- Conversion from 1 NF to 2NF, 3NF and BCNF
- DDL statement - Create, Drop Alter Queries

https://dev.mysql.com/doc/refman/8.4/en/flow-control-functions.html https://learn.micro

https://www.freecodecamp.org/news/database-normalization-1nf-2nf-3nf-table-examples/

WEEK 11

Project Using Pandas, Data Visualization

- **Series** – A labeled, one-dimensional data structure used to handle a single column of data.
- **Data Frame** – An Excel-like 2D tabular structure used to store, explore, and manipulate real-world datasets.
- **Filtering & Selection** – Apply conditions to pull out specific rows or columns (e.g., only high-salary employees).
- **Merge, Join** – Combine multiple tables/datasets based on a common column, similar to SQL joins.
- **Group By & Apply Functions** – Group data into categories and calculate aggregated insights like average, sum, or count.

SQL Mastery - Beginners to Advanced

- DML operations - insert, update and delete
- TCL commands - commit, rollback & savepoint
- DCL - Grant & revoke

★ ★ ★

- Milestone - HackerRank SQL certificate

COURSE CONTENT:

WEEK 12

Exploratory Data Analysis (EDA)

- **Univariate Analysis** – Study the distribution, spread, and pattern of a single column using histograms and boxplots.
- **Bivariate Analysis** – Examine the relationship between two variables using scatter plots and correlation.
- **Multivariate Analysis** – Analyze multiple variables together to uncover deeper patterns and interactions.
- **Real Dataset EDA** – Apply end-to-end analysis on actual messy real-world data to generate business-ready insights.

SQL Mastery - Beginners to Advanced

- Analytical functions
- Windows functions (Partition by & Group up)
- HackerRank - Gold Badge

WEEK 13

Data Analytics

- **Business Problem Understanding** – Translate a company's real-world problem into a clear, solvable data question.
- **Data Cleaning** – Fix missing values, duplicates, and inconsistent formats to make data analysis-ready.
- **EDA** – Discover patterns, trends, and anomalies from the cleaned data.
- **Insights Generation** – Convert raw numbers into actionable business recommendations.
- **Project Review** – Present the complete analytics workflow, just like it happens in a real job.

Project:
App Reviews Sentiment Analysis – In this project, we extract the sentiment (positive, negative, neutral) from app reviews and analyze the results using the Seaborn data visualization library

SQL Mastery - Beginners to Advanced

- Introduction to Indexes
- Clustered vs non-clustered index
- Composite vs Column index
- Revision of SQL
- Views in SQL
- Type of views
- Complex views and it's usage

COURSE CONTENT:

WEEK 14

Data Analytics

- **Project Planning** – Decide what data is needed, which tools to use, and how the timeline should look.
- **Dataset Analysis** – Understand the structure, quality, and gaps in a raw dataset.
- **Reporting** – Present findings as dashboards/reports that are easy for non-technical stakeholders to understand.
- **Insights Generation** – Extract decision-making insights from business metrics like stock and inventory data.
- **Project Review** – Walkthrough and feedback on the complete analytics pipeline.
- **Data Visualization** – Use plots and charts to communicate findings and highlight patterns in inventory data.

Project: Inventory Management Analysis – In this project, we analyze stock levels, sales patterns, and inventory data to generate insights that help optimize restocking and reduce wastage

SQL Mastery - Beginners to Advanced

- PL/SQL
- Creating the first procedures
- Conditional statements
- Loops
- What is cursor
- Types of cursors
- Procedure vs functions
- DML Triggers and Audit trigger
- Error handling and cursors
- Project - Library management system
- BookStore Analysis
- Advanced topic of SQL
- Wrapping up your SQ

WEEK 15

Feature Engineering

- **Standardization** – Rescale data to have mean = 0 and standard deviation = 1 so the model trains without bias.
- **Normalization** – Bring values into a fixed range (0–1), especially useful when features have very different scales.
- **Standardization vs Normalization** – Decide which technique to use based on the algorithm and data distribution.
- **Scaling Techniques** – Different scalers (MinMax, Robust, etc.) chosen based on outliers and data distribution.

WEEK 16

Feature Engineering

- **Label Encoding** – Convert categorical text values into numbers (e.g., "Yes/No" → 1/0).
- **Ordinal Encoding** – Assign numbers to ranked categories (Low/Medium/High) based on their order.
- **One Hot Encoding** – Split categories into multiple binary columns so the model doesn't assume a false ordering.
- **Get Dummies** – A quick Pandas method to easily implement One Hot Encoding.

COURSE CONTENT:

WEEK 17

Feature Engineering

- **ColumnTransformer** – Apply different preprocessing steps (scaling, encoding) to different columns simultaneously.
- **Pipeline** – Combine all preprocessing and modeling steps into a single, reusable workflow.
- **Outlier Handling** – Detect and treat extreme or incorrect values so they don't distort model accuracy.
- **End-to-End Preprocessing Pipeline** – Build a complete automated flow that takes raw data to model-ready data.
- **Handling Missing Data (Imputation)** – Fill missing values using mean/median/mode or advanced imputers so the dataset remains complete and unbiased.
- **Feature Selection** – Choose the most relevant features to improve model performance and reduce overfitting.

WEEK 18

Supervised Machine Learning

- **Linear Regression** – The most fundamental algorithm for predicting continuous values like price or salary.
- **Model Building** – The standard ML workflow of splitting, training, and evaluating a model.
- **Interview Questions** – Conceptual and coding questions commonly asked in real placement interviews.
- **Case Studies** – Apply regression to real business scenarios to build practical understanding.
- **Assumptions of Linear Regression** – Understand linearity, homoscedasticity, and multicollinearity checks that validate a regression model.
- **Multiple Linear Regression** – Predict an outcome using more than one independent variable.

WEEK 19

Supervised Machine Learning

- **Ridge Regression** – Uses an L2 penalty to reduce overfitting, especially when features are correlated.
- **Lasso Regression** – Uses an L1 penalty to shrink unimportant features to zero, also performing feature selection.
- **Regularization Techniques** – Prevent a model from becoming too complex so it generalizes well to new data.
- **Elastic Net** – Combines L1 and L2 penalties to balance feature selection and coefficient shrinkage.
- **Bias-Variance Tradeoff** – Understand how model complexity affects underfitting vs overfitting.
- **Cross-Validation** – Evaluate model stability and generalization by testing across multiple data splits.

WEEK 20

Supervised Machine Learning

- **Logistic Regression** – The core classification algorithm used to predict binary outcomes like Yes/No.
- **Classification Modeling** – The complete process of building models that predict categories.
- **Evaluation Metrics** – Accuracy, Precision, Recall, F1-score, ROC-AUC — used to judge a model's real-world performance.
- **Confusion Matrix** – A table that summarizes correct and incorrect predictions across classes.
- **Multiclass Classification** – Extend logistic regression to handle more than two output categories.
- **Hyperparameter Tuning** – Optimize model settings (like regularization strength) using GridSearch/RandomSearch.

COURSE CONTENT:

WEEK 21

Supervised Machine Learning

- **Decision Tree Regressor** – Predicts continuous values using an if-else style tree structure.
- **Decision Tree Classifier** – Uses the same tree logic to classify categories, making it an easily interpretable model.

WEEK 22

Supervised Machine Learning

- **Random Forest Regressor** – Averages predictions from multiple decision trees for more stable regression results.
- **Random Forest Classifier** – Combines the votes of multiple trees to reduce overfitting and improve accuracy.
- **Feature Importance** – Identify which variables contribute most to a tree-based model's predictions.
- **Hyperparameter Tuning (Trees)** – Control depth, splits, and leaf size to prevent overfitting in tree-based models.

WEEK 23

Supervised Machine Learning

- **Naive Bayes** – A fast, probability-based algorithm especially popular for text classification like spam detection.
- **K-Nearest Neighbors (KNN)** – Makes predictions by looking at the closest, most similar data points — simple and intuitive.

WEEK 24

Supervised Machine Learning

- **Support Vector Machine (SVM)** – Draws the best possible boundary (margin) between classes.
- **Kernel Trick** – Projects non-linear data into a higher dimension to make it linearly separable.
- **End-to-End ML Workflow** – The complete production-style process from raw data to a deployed model.
- **Model Deployment Preparation** – Making a trained model ready for use inside a real application.

Project: COVID Prediction System – In this project, we build a machine learning model to predict COVID risk/outcome based on patient data, export it using Pickle, and deploy it as a REST API using Flask, tested via Postman

COURSE CONTENT:

WEEK 25

Unsupervised Machine Learning

- **KMeans Clustering** – Groups data into a fixed number of clusters based on similarity.
- **DBSCAN Clustering** – Forms clusters based on data density, capable of handling irregular shapes and detecting outliers/noise.
- **Elbow Method** – Techniques used to determine the optimal number of clusters for a dataset.
- **Distance Metrics** – Understand Euclidean, and Cosine distance measures used across clustering algorithms.
- **Cluster Visualization** – Use dimensionality reduction to visualize high-dimensional clusters in 2D/3D.

WEEK 26

Unsupervised Machine Learning

- **Principal Component Analysis (PCA)** – Reduces the number of dimensions in high-dimensional data without losing important information.
- **Apriori Algorithm** – An association rule mining technique (market basket analysis) that finds "customers who buy X also buy Y" patterns.
- **Eigenvalues & Eigenvectors** – The mathematical foundation behind how PCA identifies the directions of maximum variance.
- **Dimensionality Reduction Use Cases** – Apply PCA for visualization, noise reduction, and speeding up downstream ML models.
- **Market Basket Analysis** – A real-world retail application of association rule mining to recommend frequently bought-together products.

WEEK 27

Ensemble Learning & Optimization

- **Bagging** – Trains multiple models in parallel and averages their results to reduce variance (e.g., Random Forest).
- **Boosting** – Trains models sequentially, where each new model corrects the errors of the previous one.
- **Batch Gradient Descent** – Updates model weights using the entire dataset at once.
- **Mini Batch GD** – Updates weights using small batches of data — a balance between speed and stability.
- **SGD (Stochastic Gradient Descent)** – Updates weights using one sample at a time, making it efficient for large datasets.

Project: Image Classification using MobileNetV2 – In this project, we classify images into categories using the MobileNetV2 deep learning model through transfer learning.

WEEK 28

Deep Learning Fundamentals

- **ML vs DL** – Understand the difference between traditional ML and Deep Learning, and when to use which.
- **Neural Networks** – Layered structures inspired by the human brain that learn complex patterns.
- **Activation Functions** – Add non-linearity to a network so it can learn complex relationships (ReLU, Sigmoid, etc.).
- **Optimizers** – Algorithms (Adam, RMSprop, etc.) that efficiently update a model's weights during training.
- **Loss Functions** – Measure the difference between predicted and actual values to guide model training.
- **Overfitting in Deep Learning** – Understand techniques like Dropout and Early Stopping that prevent a network from memorizing training data.

COURSE CONTENT:

WEEK 29

Artificial Neural Networks (ANN)

- **ANN Architecture** – Design the structure of input, hidden, and output layers.
- **ANN Implementation** – Build and train an actual neural network in code.
- **Forward Propagation** – Pass input data through the layers to generate an output prediction.
- **Backpropagation** – Send the error backward through the network to adjust weights and improve the model.
- **Model Evaluation** – Validate the accuracy and performance of the trained ANN.

Project: Employee Attrition Prediction System – In this project, we build an Artificial Neural Network to predict whether an employee is likely to leave the company, based on historical HR data.

WEEK 30

Convolutional Neural Networks (CNN)

- **CNN Basics** – A specialized deep learning architecture that automatically extracts features from images.
- **Kernel, Stride, Pooling** – Techniques for detecting important image features and efficiently reducing data size.
- **Feature Maps** – The visual patterns (edges, shapes, textures) extracted at each layer of the network.
- **Transfer Learning (MobileNetV2)** – Fine-tune a powerful pre-trained model on your own data — saving both time and computing resources.
- **Data Augmentation** – Artificially expand training data using rotations, flips, and crops to improve generalization.
- **CNN Architectures (VGG, ResNet)** – Popular pre-built CNN architectures used as benchmarks and backbones for image tasks.

Project: Image Classification using MobileNetV2 – In this project, we classify images into categories using the MobileNetV2 deep learning model through transfer learning.

WEEK 31

Sequence Models

- **RNN (Recurrent Neural Network)** – Processes sequential data like text or time-series while retaining memory of previous inputs.
- **LSTM** – Uses special gates to remember long-term dependencies, solving a key limitation of basic RNNs.
- **GRU** – Performs a similar function to LSTM but with a simpler and faster architecture.
- **Sequential Data Processing** – Handle time-dependent data (stock prices, sentences) in the correct order.
- **Bidirectional RNNs** – Process sequences in both forward and backward directions to capture richer context.
- **Attention in Sequence Models** – Allow the model to focus on the most relevant parts of a sequence when making predictions.

WEEK 32

Natural Language Processing (NLP)

- **Text Cleaning** – Remove noise like punctuation, stopwords, and special characters from raw text.
- **Stemming, Lemmatization** – Convert words to their root form (running → run) so the model interprets them consistently.
- **TF-IDF** – Numerically represent how important a word is within a document.
- **Word Embeddings** – Convert words into meaning-aware numerical vectors.
- **NER, POS Tagging** – Identify names/entities in text and tag the grammatical role of each word (noun, verb, etc.).

Project: Employee Information Extraction from PDFs – In this project, we extract structured employee details (name, designation, skills, etc.) from PDF documents using NLP techniques like text cleaning and Named Entity Recognition (NER).

COURSE CONTENT:

WEEK 33

NLP Project & Generative AI

- **Regex & NER-based Extraction** – Extract structured information using pattern-matching and entity recognition.
- **Transformers** – The attention-based architecture that serves as the foundation for all modern LLMs.
- **LLMs** – Large Language Models that understand and generate human-like text.
- **LangChain (Data Loaders, Agents, Chains)** – A framework for building LLM-based applications — loading data, chaining multi-step logic, and creating autonomous agents.
- **RAG & Advanced RAG (Chunking, Embeddings, Vector DB, BM25, Re-Ranker)** – A retrieval technique that lets an LLM give accurate, hallucination-free answers using your own custom data.
- **LangGraph** – Design complex, multi-step AI agent workflows using a graph-based structure.
- **Tokenization** – Break text into tokens (words/sub-words) — the basic units an LLM reads and generates.
- **Attention Mechanism** – Lets a model weigh which parts of the input matter most when generating each output token.
- **Prompt Engineering** – Design effective inputs (zero-shot, few-shot, chain-of-thought) to get accurate, reliable LLM outputs.
- **Generative AI Landscape** – Overview of open-source vs closed-source models and how to pick the right one for a use case.

Project: Business Strategy Analyzer using Lang Graph – In this project, we build an AI agent workflow using Lang Graph and LLMs that analyzes business data/documents and generates strategic insights and recommendations.

WEEK 34

Walkthrough on Microsoft Excel

- Introduction to Excel
- HLookup vs VLOOKUP
- Basic Functions
- Advance Function
- Pivots and visualization

WEEK 35

Data Analysis & Reporting with Power BI

- What is Power BI?
- Power BI Components (Desktop, Service, Mobile)
- Workflow in Power BI
- Installation & Setup
- Cardinality in relationship
- Importing Excel, CSV, JSON, SQL Server, Web APIs
- Direct Query vs Import Mode
- Data source credentials

COURSE CONTENT:

WEEK 36

Data Analysis & Reporting with Power BI

- Relationships between tables
- Cardinality, cross-filtering
- Creating a data model
- Star vs Snowflake schema
- Applied Steps
- Column transformation (split, merge, fill, etc.)
- Filtering and removing rows

WEEK 37

Data Analysis & Reporting with Power BI

- Calculated Columns vs Measures
- Basic DAX functions: SUM, COUNT, DISTINCTCOUNT IF, SWITCH, RELATED, CALCULATE
- Syntax and rules
- Bar, Line, Pie, Card, KPI
- Maps, Matrix, Tables
- Custom visuals from marketplace
- Visual, Page, Report filters
- Slicers for Date, Category
- Syncing slicers across pages
- Projects on Power BI
- HR attrition, or Profit & Loss statement
- Time-series analysis with DAX

WEEK 38

Revision and solving real-life use-cases

WEEK 39

External Mock interviews and preparation

JOIN OUR COMMUNITY:

JOIN YOUTUBE

JOIN INSTAGRAM

JOIN LINKEDIN

JOIN FACEBOOK

JOIN TELEGRAM