**ML List**  
**Step 1: Python Basics**

**Why:** Python is the main language for ML.  
**Topics:**

* Variables, data types
* Loops, conditions
* Functions
* Lists, dictionaries, sets
* Modules and packages

**Resources:** Online Python tutorials, “Automate the Boring Stuff with Python”

**Step 2: Python Libraries for ML**

**Why:** ML heavily depends on libraries.  
**Topics:**

* **NumPy:** Arrays, vectorized operations, math functions
* **Pandas:** DataFrames, reading/writing CSV, data cleaning
* **Matplotlib & Seaborn:** Basic plotting, data visualization

**Step 3: Basic Mathematics for ML**

**Why:** ML is math-heavy, so you need these foundations.  
**Topics:**

* Linear algebra: vectors, matrices, dot product
* Statistics: mean, variance, probability, distributions
* Calculus (basics): derivatives, gradient

**Step 4: Data Preprocessing**

**Why:** ML models perform poorly without clean data.  
**Topics:**

* Handling missing values
* Encoding categorical variables
* Feature scaling (normalization, standardization)
* Train-test split

**Step 5: Introduction to ML Concepts**

**Why:** You need to understand the theory first.  
**Topics:**

* Supervised vs unsupervised learning
* Regression vs classification
* Overfitting & underfitting
* Model evaluation metrics (accuracy, precision, recall, F1-score)

**Step 6: Supervised Learning**

**Why:** The easiest way to start applying ML.  
**Topics:**

* Linear Regression
* Logistic Regression
* K-Nearest Neighbors (KNN)
* Decision Trees & Random Forests
* Support Vector Machines (SVM)

**Step 7: Unsupervised Learning**

**Why:** For clustering and dimensionality reduction.  
**Topics:**

* K-Means Clustering
* Hierarchical Clustering
* Principal Component Analysis (PCA)

**Step 8: Model Optimization**

**Why:** To improve your model’s performance.  
**Topics:**

* Cross-validation
* Hyperparameter tuning (Grid Search, Random Search)
* Regularization (L1, L2)

**Step 9: Deep Learning Basics**

**Why:** Deep learning is a core part of modern ML.  
**Topics:**

* Neural networks fundamentals
* Activation functions
* Forward and backward propagation
* Frameworks: TensorFlow / PyTorch basics

**Step 10: Advanced Deep Learning**

**Why:** For real-world applications.  
**Topics:**

* Convolutional Neural Networks (CNN) – for images
* Recurrent Neural Networks (RNN/LSTM) – for sequences
* Transformers (intro) – for NLP
* Model deployment basics

**Step 11: Real Projects**

**Why:** Experience is key.  
**Examples:**

* Predicting house prices
* Customer segmentation
* Handwritten digit recognition (MNIST)
* Sentiment analysis on text