Data Science | 30 Days of Machine Learning | Day - 3

Educator Name: Nishant Dhote
Support Team: **+91-7880-113-112**

 # Machine Learning Development Life Cycle (MLDLC/MLDC):

 # Data science life cycle (DSLC):

 # Tools used in Machine Learning? Installing: Anaconda | Jupiter Notebook (IDEs)

Optional Tools: Spyder | PyCharm | Noteable | Google Colab | Kaggle Notebooks | Microsoft Azure Notebooks | Apache Zeplin | Count.co and Many More

#How to import dataset and download data files

#How we create virtual environment

#Data Gathering

#Working with CSV Files

#Working with JSON/SQL

#Fetching data from an API

#Fetching data using web scraping

#Framing a Machine Learning Problem

 # Machine Learning Development Life Cycle (MLDLC/MLDC):

1. Frame the problem
2. Gathering data
3. Data pre processing
4. Exploratory data analysis
5. Feature engineering & selection
6. Model training, evaluation and selection
7. Model deployment
8. Testing
9. Optimize

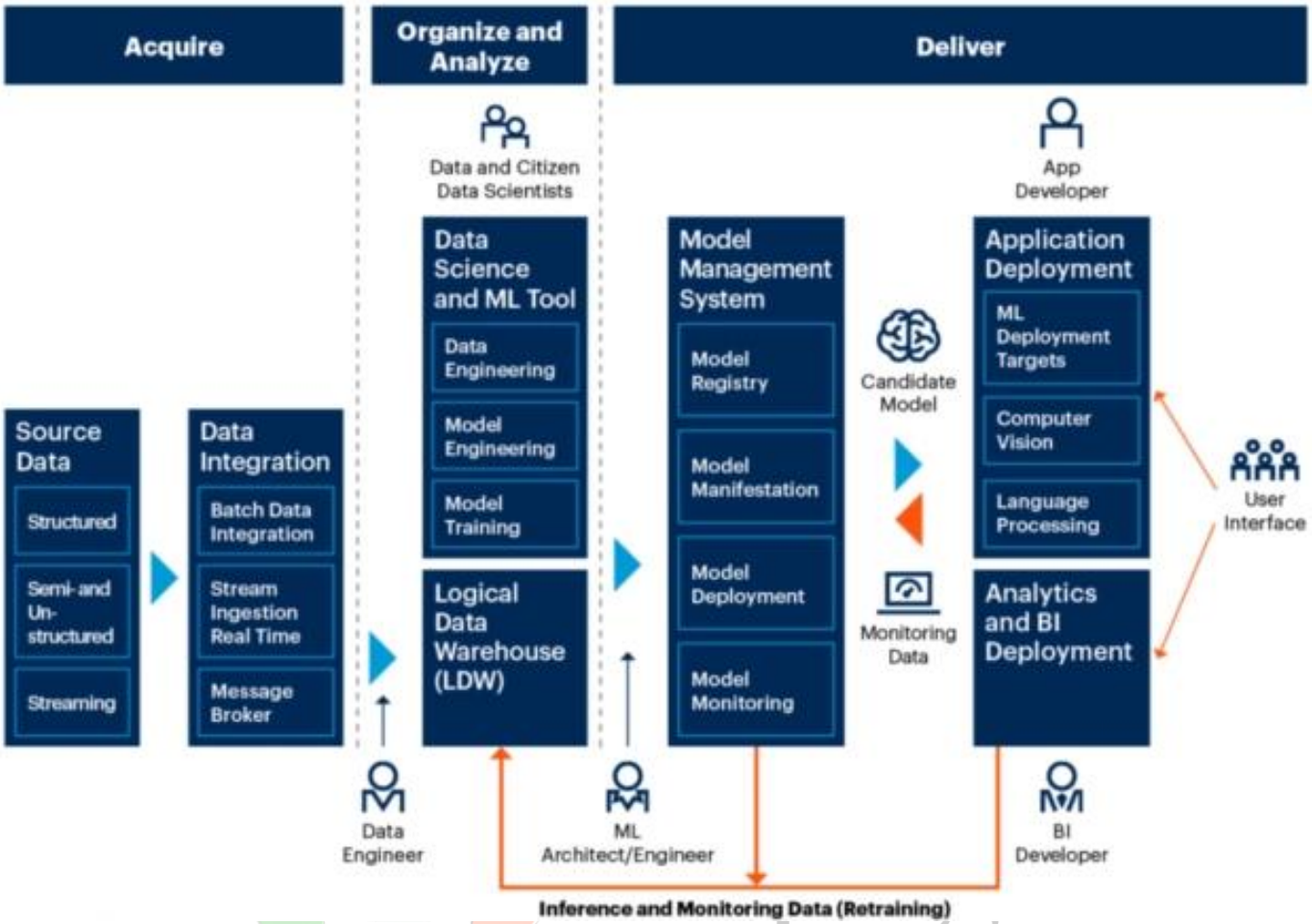
   Note: Some books mention in 10-12 steps in MLDLC.

What is SDLC?

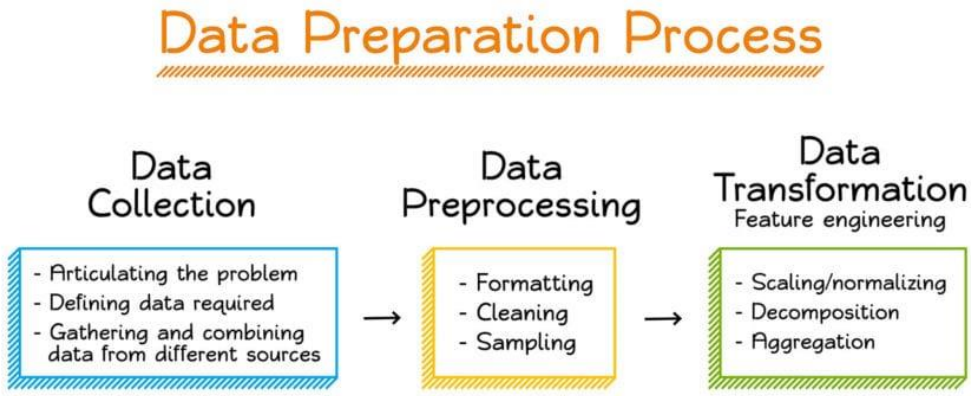
The software development lifecycle (SDLC) is the cost-effective and time-efficient process that development teams use to design and build high-quality software. The goal of SDLC is to minimize project risks through forward planning so that software meets customer expectations during production and beyond. This methodology outlines a series of steps that divide the software development process into tasks you can assign, complete, and measure.

Read Blog: Click Here

Architecture to Operationalize MLDLC

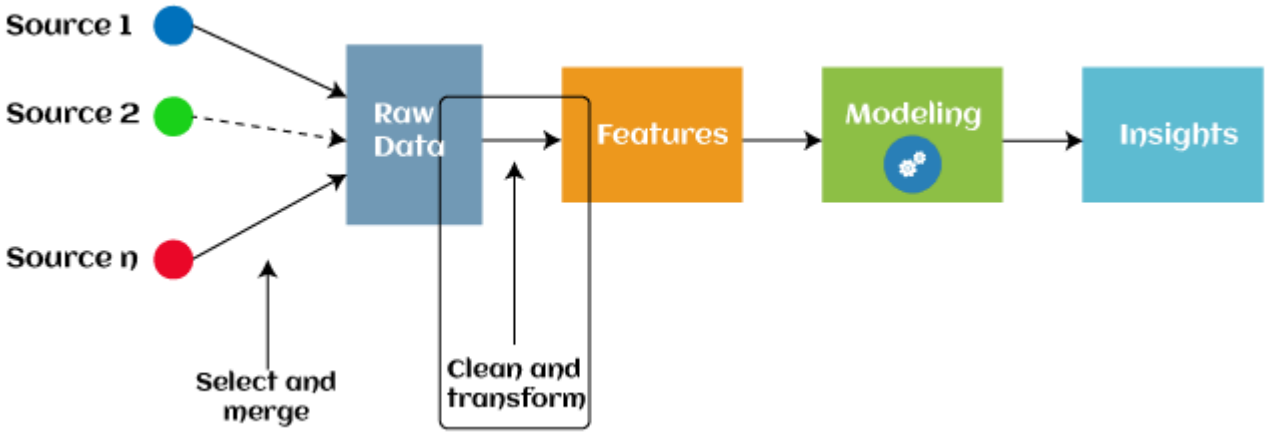


1. **Frame the problem:** The first stage of MLDLC is all about "What do we want?" Project planning is a vital role in the software delivery lifecycle since this is the part where the team estimates the cost and defines the requirements of the new software.
2. **Gathering data:** The second step of MLDLC is gathering maximum information from the client requirements for the product. Discuss each detail and specification of the product with the customer. Data collection means pooling data by scraping, capturing, and loading it from multiple sources, including offline and online sources. High volumes of data collection or data creation can be the hardest part of a machine learning project, especially at scale.
3. **Data pre-processing:** Data pre-processing in Machine Learning refers to the technique of preparing (cleaning and organizing) the raw data to make it suitable for a building and training Machine Learning models**.**

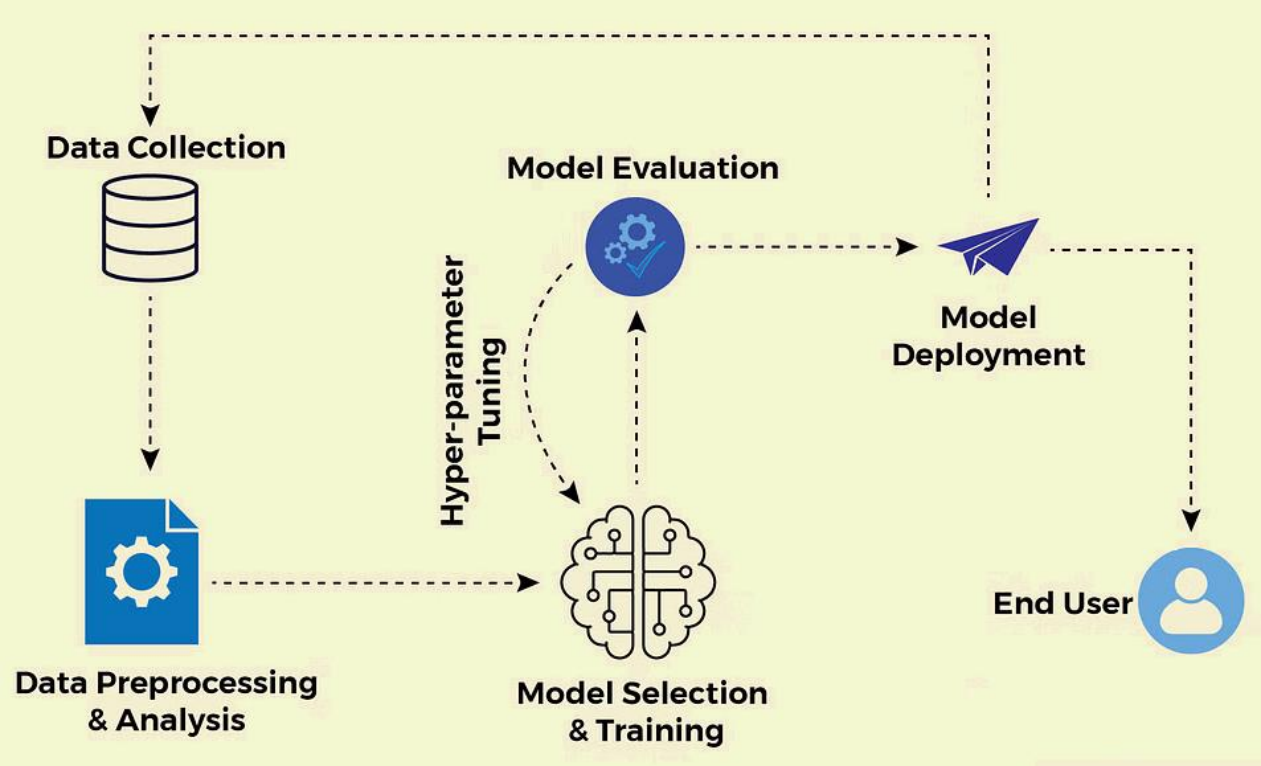


4. **Exploratory Data Analysis (EDA):** Exploratory data analysis (EDA) is used by data scientists to analyse and investigate data sets and summarize their main characteristics, often employing data visualization methods.

5. **Feature Engineering & Selection:** Feature engineering in Machine learning consists of mainly 5 processes: Feature Creation, Feature Transformation, Feature Extraction, Feature Selection, and Feature Scaling. It is an iterative process that requires experimentation and testing to find the best combination of features for a given problem.

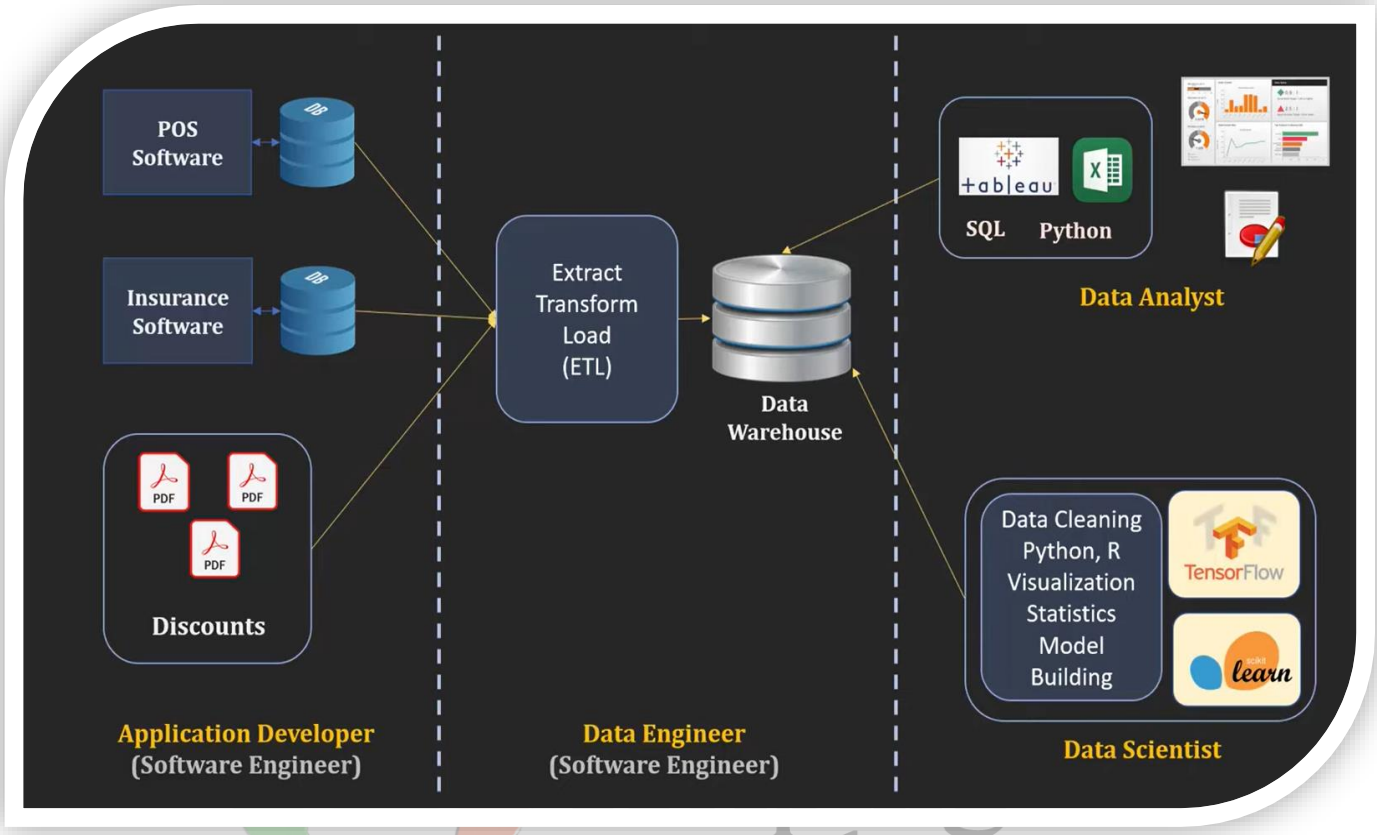


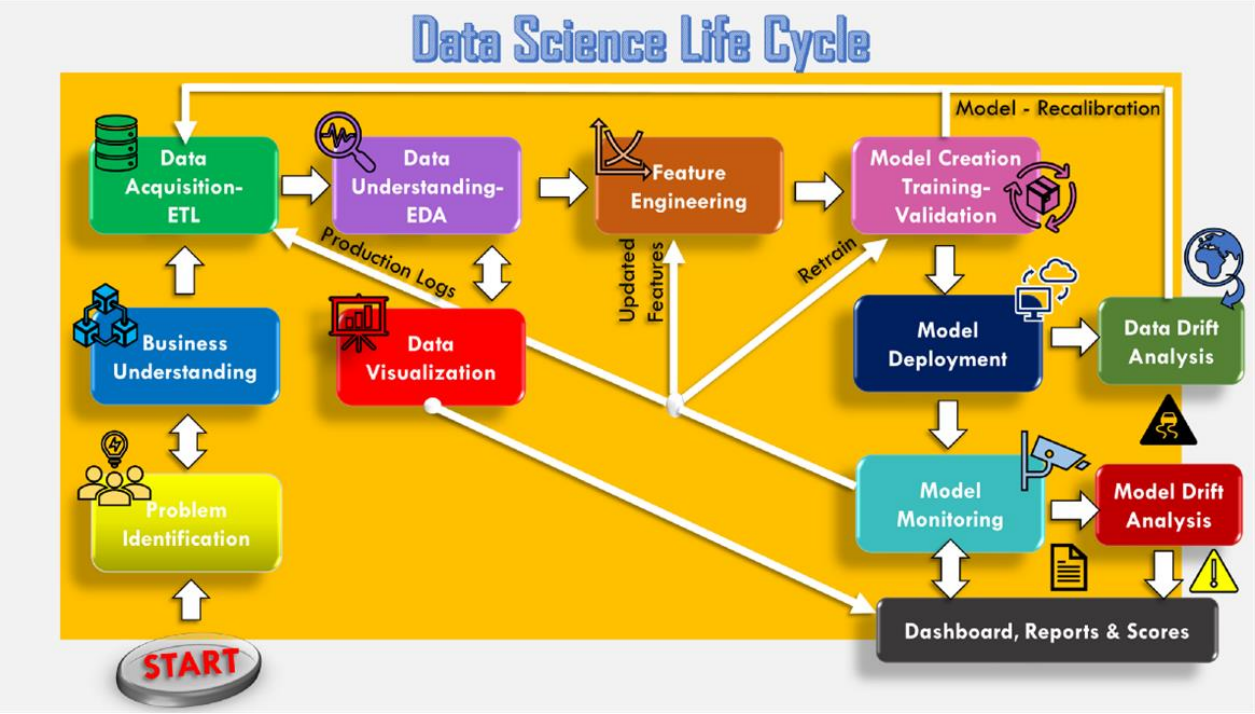
6. **Model training, evaluation and selection:** Model evaluation is the process that uses some metrics which help us to analyse the performance of the model. As we all know that model development is a multi-step process and a check should be kept on how well the model generalizes future predictions. Therefore, evaluating a model plays a vital role so that we can judge the performance of our model.

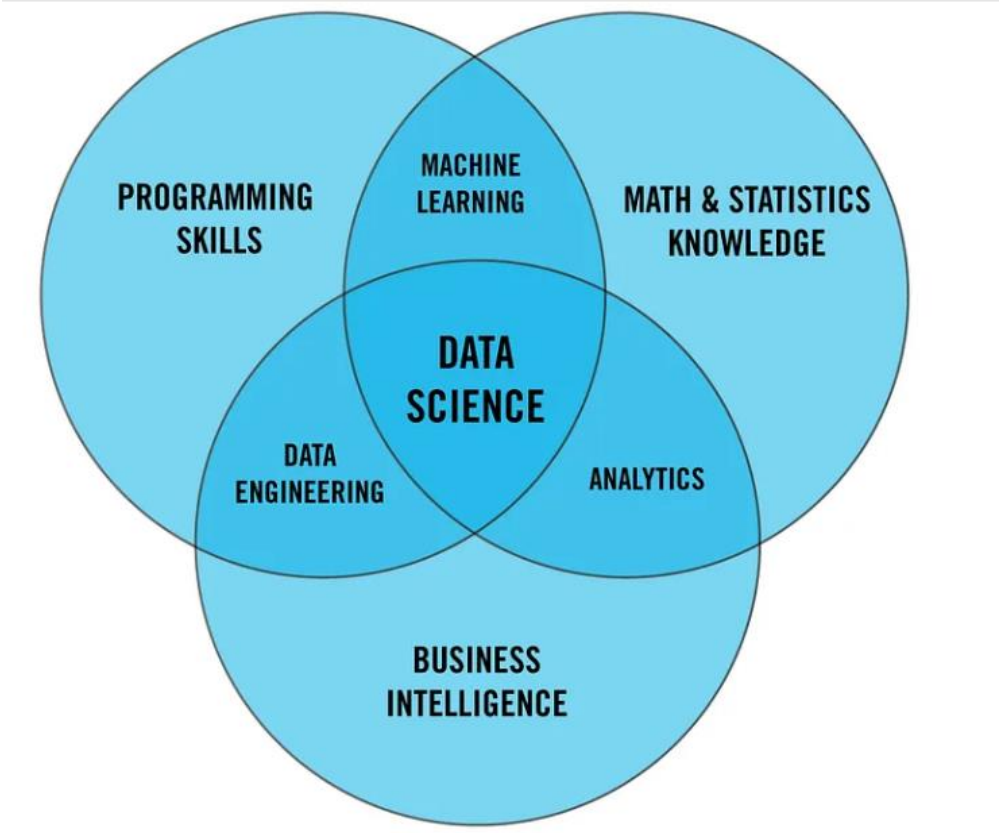


7. **Model deployment:** Deploying a machine learning model, known as model deployment, simply means to integrate a machine learning model and integrate it into an existing production environment.
8. **Testing:** In machine learning, model testing is referred to as the process where the performance of a fully trained model is evaluated on a testing set.
9. **Optimize:** Machine learning optimisation is the process of iteratively improving the accuracy of a machine learning model, lowering the degree of error. Machine learning models learn to generalise and make predictions about new live data based on insight learned from training data.

Data Scientist Work?
- DATA ANALYST Vs DATA SCIENTIST Vs DATA ENGINEER




Data Science Life Cycle
Model - Recalibration
Data Acquisition- ETL
Data Understanding- EDA
Feature Engineering
Model Creation Training- Validation
Production Logs
Updated Features
Retrain
Business Understanding
Data Visualization
Model Deployment
Data Drift Analysis
Problem Identification
Model Monitoring
Model Drift Analysis
Dashboard, Reports & Scores
START


PROGRAMMING SKILLS
MACHINE LEARNING
MATH & STATISTICS KNOWLEDGE
DATA SCIENCE
DATA ENGINEERING
ANALYTICS
BUSINESS INTELLIGENCE

 # Tools used in Machine Learning?

Installing:

Anaconda: https://www.anaconda.com/download

Jupiter Notebook (IDEs): https://jupyter.org/

Discuss Notebook Interface and Navigations.

Installation Video Link : https://youtu.be/AS1a5K8zejk?si=ilKogijLjqdjhThi

#How to import dataset and download data files

Step 1 - Upload dataset in the system folder.

Step 2 – Run Code

```
# pandas
import pandas as pd
```

Step 3 – Run → pd.read_csv('hotel_bookings.csv')

If you want to Download Notebook so click on File→ Download us

#How we create virtual environment

What Is a Virtual Environment?

The main purpose of Python virtual environments is to create an isolated environment for Python projects. This means that each project can have its own dependencies, regardless of what dependencies every other project has.

The great thing about this is that there are no limits to the number of environments you can have since they're just directories containing a few scripts. Plus, they're easily created.

Required Dataset Links: https://github.com/TheiScale/30_Days_Machine_Learning

#Data Gathering

#Working with CSV Files

1. Importing pandas

```
import pandas as pd
```

2. Opening a local csv file

```
df = pd.read_csv('aug_train.csv')
```

3. Opening a csv file from an URL

URL Link: https://raw.githubusercontent.com/cs109/2014_data/master/countries.csv

```
import requests
from io import StringIO
url = "https://raw.githubusercontent.com/cs109/2014_data/master/countries.csv"
headers = {"User-Agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.14; rv:66.0) Gecko/20100101 Firefox/66.0"}
req = requests.get(url, headers=headers)
data = StringIO(req.text)
pd.read_csv(data)
```

4. Sep Parameter

Step 1 : pd.read_csv('movie_titles_metadata.tsv')

Step 2 :

```
pd.read_csv('movie_titles_metadata.tsv',sep='\t')
```

Step 3 :

```
pd.read_csv('movie_titles_metadata.tsv',sep='\t',names=['sno','name','release_year','rating','votes','genres'])
```

5. Index_col parameter

Step 1 : pd.read_csv('aug_train.csv')

Step 2: pd.read_csv('aug_train.csv',index_col='enrollee_id')

6. Header parameter

Step 1 : pd.read_csv('test.csv')

Step 2 : pd.read_csv('test.csv',header=1)

7. Use_col parameter

pd.read_csv('aug_train.csv',usecols=['enrollee_id','gender','education_level'])

8. Squeeze parameters

pd.read_csv('aug_train.csv',usecols=['gender'],squeeze=True)

9. Skiprows/nrows Parameter

Step 1 : pd.read_csv('aug_train.csv',skiprows=[0,1])

pd.read_csv('aug_train.csv',nrows=100)

10. Encoding parameter

Step 1 : pd.read_csv('zomato.csv')

pd.read_csv('zomato.csv',encoding='latin-1')

11. Skip bad lines

Step 1 : pd.read_csv('zomato.csv', address=';', encoding="latin-1")

pd.read_csv('BX-Books.csv', address=';', encoding="latin-1",error_bad_lines=False)

12.dtypes parameter

Step 1 : pd.read_csv('aug_train.csv').info()

pd.read_csv('aug_train.csv',dtype={'target':int}).info()

13. Handling Dates

Step 1 : pd.read_csv('IPL Matches 2008-2020.csv').info()

pd.read_csv('IPL Matches 2008-2020.csv',parse_dates=['date']).info()

14. Convertors

pd.read_csv('IPL Matches 2008-2020.csv',converters={'team1':rename})

```
def rename(name):
    if name == "Royal Challengers Bangalore":
        return "RCB"
    else:
        return name
```

IN: rename("Royal Challengers Bangalore")

OUT: 'RCB'

15. na_values parameter

Step 1 : pd.read_csv('aug_train.csv')

pd.read_csv('aug_train.csv',na_values=['Male',])

16. Loading a huge dataset in chunks

Step 1 : pd.read_csv('aug_train.csv')

```
dfs = pd.read_csv('aug_train.csv',chunksize=5000)
for chunks in dfs:
    print(chunk.shape)
```

#Working with JSON/SQL

JSON : JavaScript Object Notation.

API → JSON → JAVA | Python

What is JSON used for?

JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications (e.g., sending some data from the server to the client, so it can be displayed on a web page, or vice versa)

Code : Jupiter Notebook

import pandas as pd

Woking with JSON

pd.read_json('train.json')

URL Data with JSON

pd.read_json('https://api.exchangerate-api.com/v4/latest/INR')

Working with SQL (Structured Query Language)

SQL is used to communicate with a database. According to ANSI (American National Standards Institute), it is the standard language for relational database management systems. SQL statements are used to perform tasks such as update data on a database, or retrieve data from a database.

Xampp download link: https://www.apachefriends.org/index.html

XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself.

SQL Series Xampp Install : https://youtu.be/ZcDm262cam4?si=tkU5tEs5G3RqAm0o

SQL dataset : https://github.com/TheiScale/30_Days_Machine_Learning

Pandas read_json documentation:
https://pandas.pydata.org/docs/reference/api/pandas.read_json.html

Pandas read_sql_query documentation:
https://pandas.pydata.org/docs/reference/api/pandas.read_sql_query.html#pandas.read_sql_query

Code Jupiter Notebook:

Working with SQL

```
!pip install mysql.connector

--------------------------

import mysql.connector

--------------------------

conn =
mysql.connector.connect(host='localhost',user='root',password='',database='world')

--------------------

pd.read_sql_query("SELECT * FROM city",conn)

----or----

df = pd.read_sql_query("SELECT * FROM countrylanguage",conn)

df
```

#Framing a Machine Learning Problem

#Data Gathering

#Fetching data from an API

#Fetching data using web scraping

Data Story Telling: Curious Data Minds

Case Study Link: https://data-flair.training/blogs/data-science-in-agriculture/

https://intellias.com/how-to-encourage-farmers-to-use-big-data-analytics-in-agriculture/