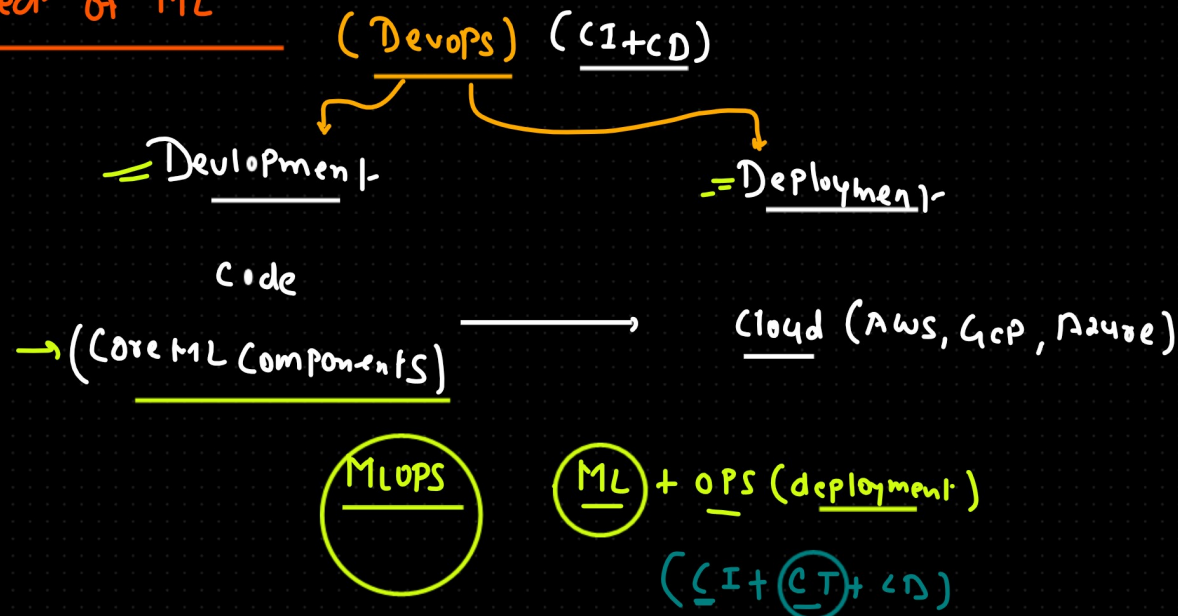① Data ingestion (Data collection)

② EDA or analysis of the data

③ FE (Preprocessing or transformation)

④ Model building

⑤ Model evalution

Data Science

↓

Data

End to End Project of ML

(Devops) (CI+CD)

=Devlopment

Code

→ (Core ML Components)

=Deployment

Cloud (AWS, GCP, Azure)

MLOPS

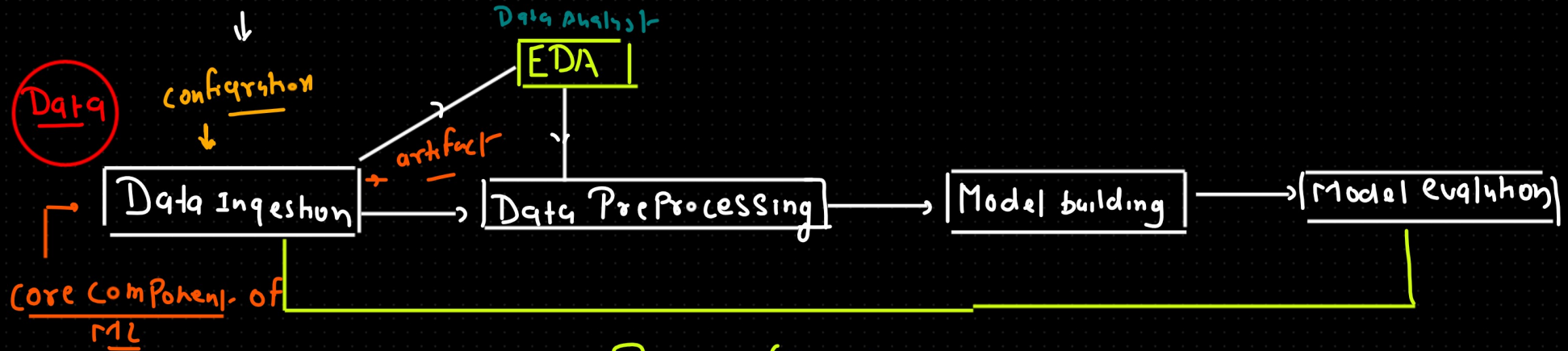ML + OPS (deployment)

(CI + CT + CD)

Data Engineers
Database → Python → .csv

Python | Flask | Exception | logging

mongoDB

Configuration | Artifact

Data Analyst
EDA

Data

configuration

artifact

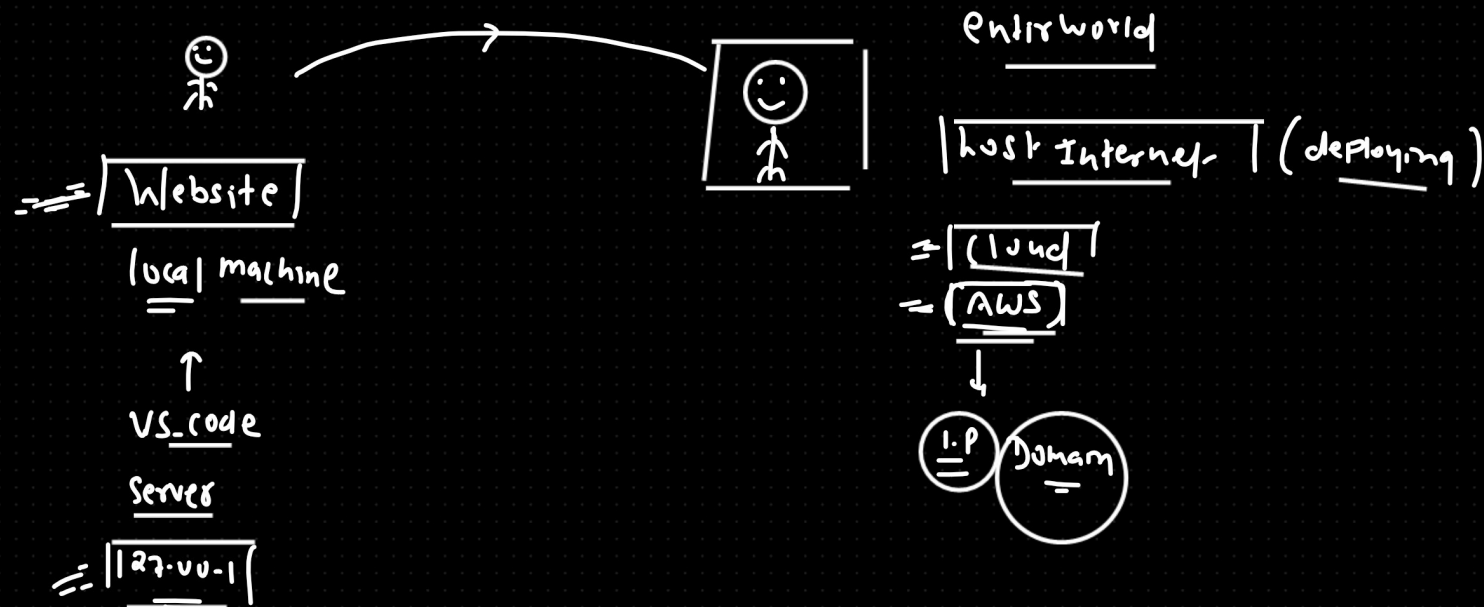Data Ingeshun → Data PreProcessing → Model building → Model Evaluation

Core Component of ML

Package (PyPi, Conda manger)

Docker

Cloud

Container

AWS

IMAGE
↑
APP
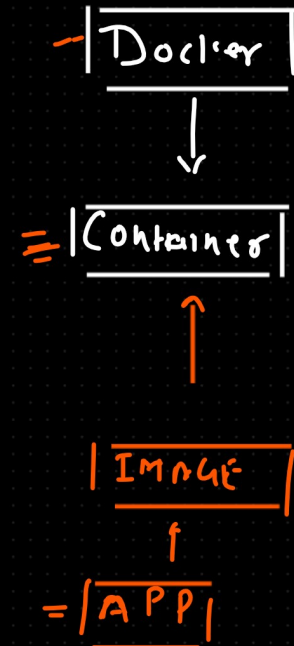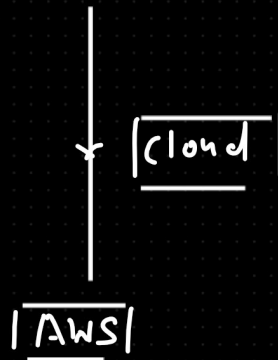
Package → folder → modules (Python file) .Py file

Config

Data ingestion

Artifact

Feature store

entirworld

host Internet (deploying)

Website

local machine

↑

VS code

server

127.00.1

Cloud

AWS
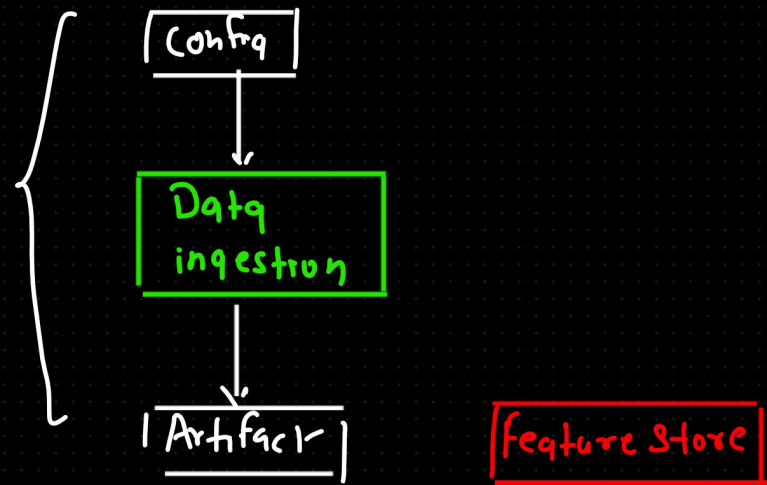
I.P Domain

| Problem statement | →1 | Wafer fault Detection | Domain expert - SME |

-1 → wafer is faulty
+1 → wafer not faulty

target $(-1/+1)$

~~Regression~~

→ classification

Binary classification

| Sensor1 | Sensor2 | Sensor3 | — — — — — — — | Sensor500 |

Silicon → | Semiconductor | → IC → | Microprocessor |

↓

| Sensor |

≡ Project-1 ⇒ | Dependencies | Pandas
Numpy =
Sklearn =
Seaborn
flask

↑

(ML)   ≡ | 3.8 |

- Project-2 | Dependencies | { Tensor flow
Pytorch
Numpy =
OPencv

↑

(DL)   = (3.9)

Virtual env → System → (env) Virtual
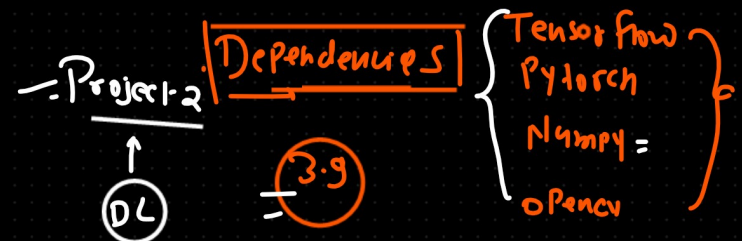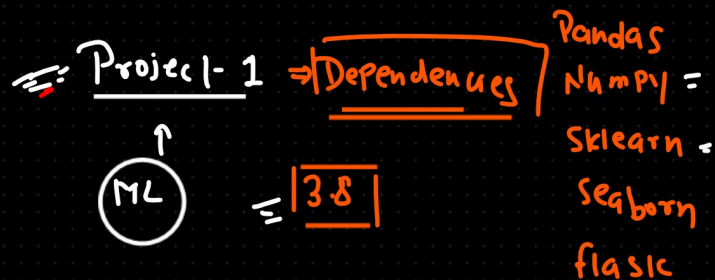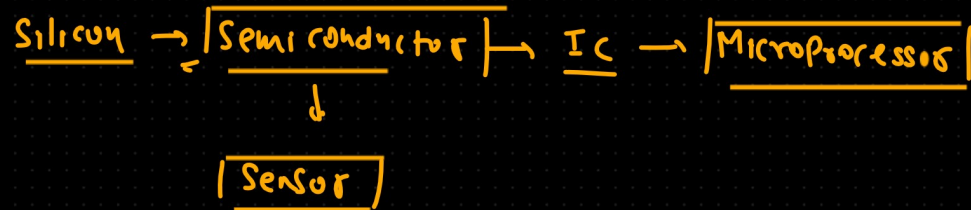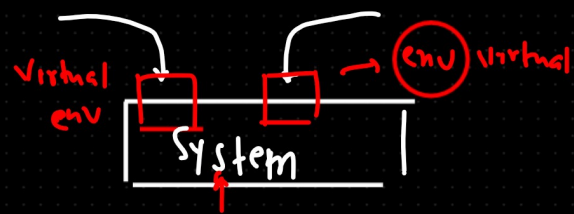
Python interpretenv

Datascience Project manager
↓
= conda

Package ⇒ folder → file → Python (module)
(library) ⇒ modules

= Src folder
! → Components

--Init--.Py ←

PyPi repo
Pandas , Numpy

Data_ingeshon.Py
Data_PreProcess.Py
Model_trainer.Py

Package

Setup.Py → findPackagey( )