UNIT-4

* Software Defined Networking :-

* It is a networking approach.
* It enables the control and managing the n/w with s/w application that means by using programming we are going to control the n/w.
* The programming to n/w centrally controlled manner through s/w applications using open "APIs". (Application programming interfaces).
* It improves the n/w performance through "N/w Virtulization".
* The s/w Applications (APIs) will control the traffic and communication between n/w devices.

SUBSCRIBE

@CodingFataka

Youtube Channel,

Follow Us on Telegram,

Instagram for Q/A.

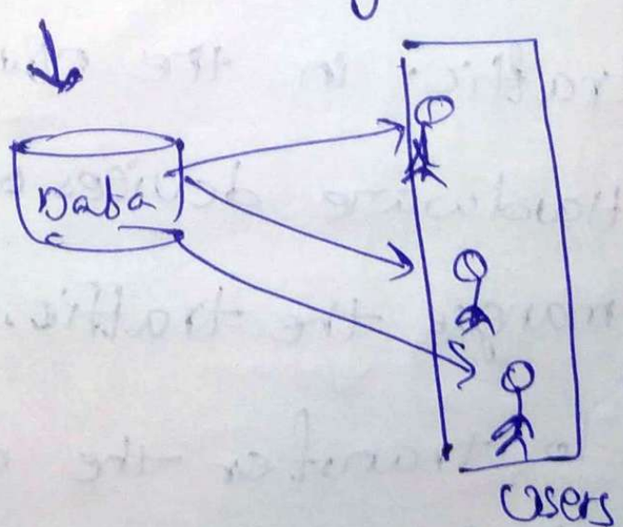
* It can control the traditional n/w with "SW".

* Components of SDN:

* It is "decoupling" of "control plane" (which decides where traffic is sent) and "data plane" (which will send the packets to the destination).

Components:

1. Data plane: It will do the below activities or operations:
   1. Forwarding of packets.
   2. Segmentation and reassembling the data.
   3. Replication (Copy) of packets for multi casting.



2. Control plane:- This is the 'brain' of the network.

* It will control whole activities of network. (like cpu). Infrastructure.

* The activities of control plane are:

1. Making routing tables.
2. Setting packet handling policies.

* It makes decisions about how network traffic should be handled.

* Applications:-

1. Data Center Networking
2. Wide Area Networks (WANs)
3. IoT Applications.

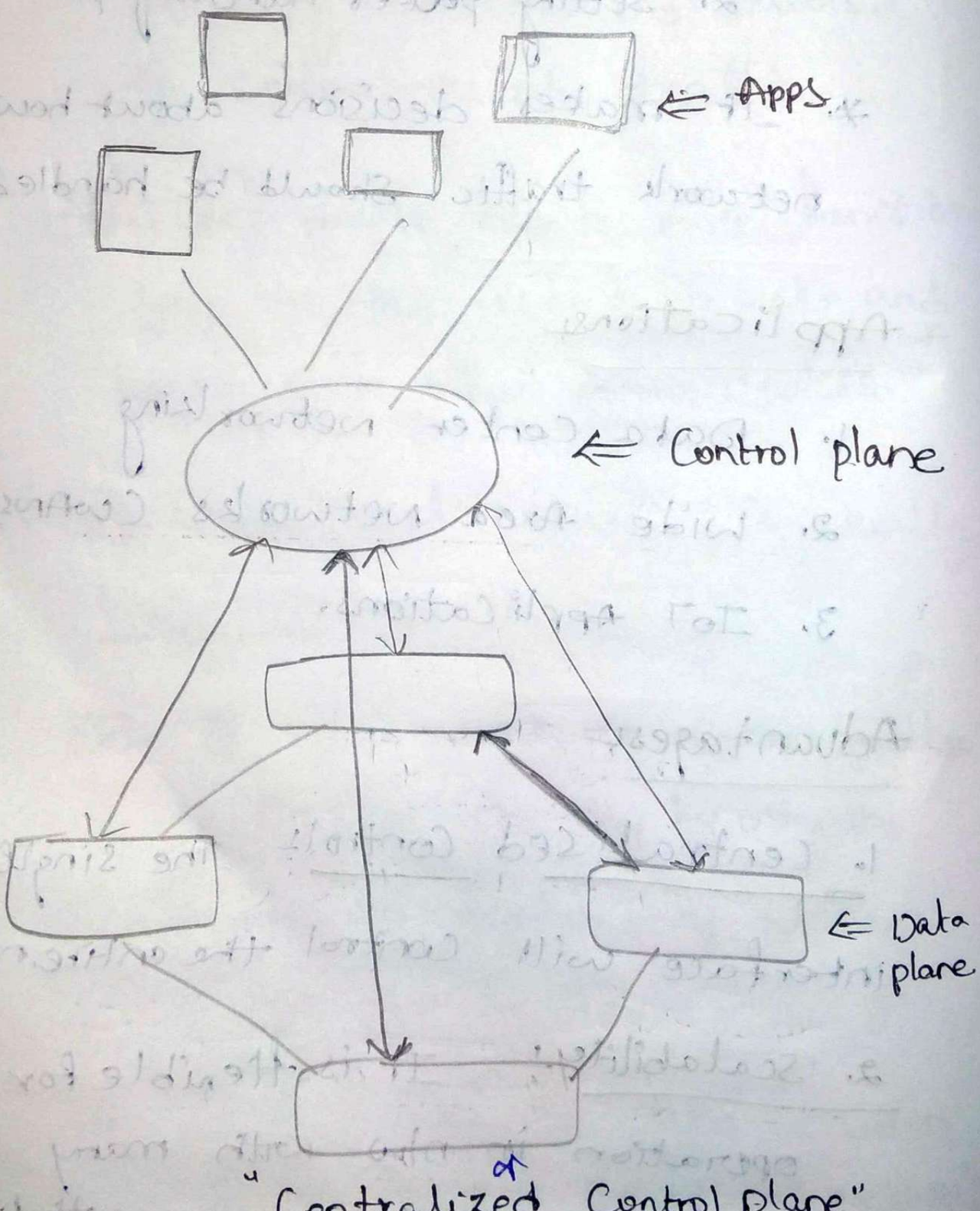
* Advantages:-

1. Centralised Control:- The single interface will control the entire n/w.

2. Scalability:- It is flexible for operation in n/w with many users as well as in a n/w with limited users.

3. Automation :- Automates the many n/w management decisions via programmable n/w requirements.

* which decrease the human help or intervention to make decisions.



* Traditional Network:-

* The most common networking, traditional networking uses "fixed functions and dedicated hardware and network devices" including switches and routers, to control the nlw traffic.

* Every device has it's own functionality in the network. to help and work with other ~~devices~~. devices.

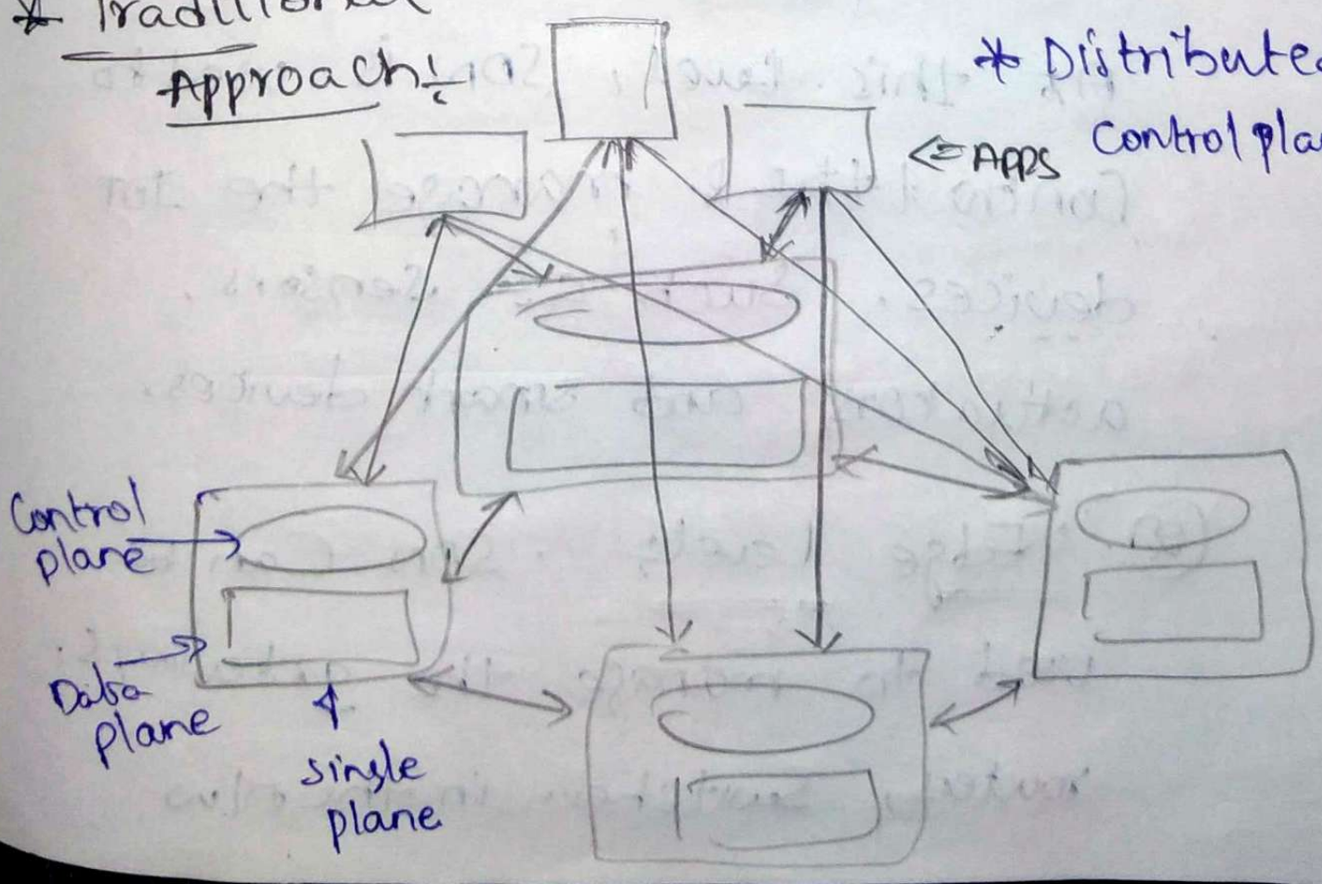
* Traditional networking is usually hardware-based.

* If you want to control the traffic in the nlw by using the Hardware devices only we can mangue the traffic.

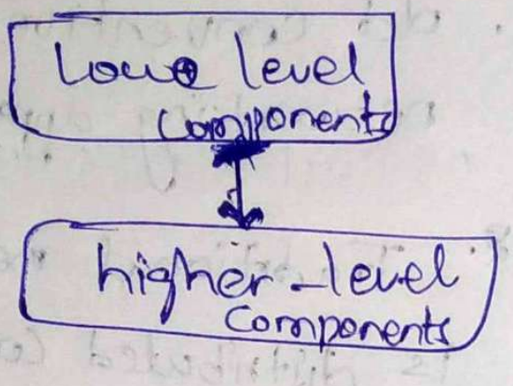
* To transfer the data between devices it uses traditional networking protocols.

SDN	Traditional.
1. virtual n/w king approach	1. old conventional networking approach.
2. it is centralised control,	2. Traditional network is distributed control.
3. This is programmable	3. it is non programmable.
4. decouples the control plane & data plane	4. Both presents in the same plane.
5. maintenance cost is lower than Traditional N/w	5. Traditional n/w maintenance cost is higher than SDN

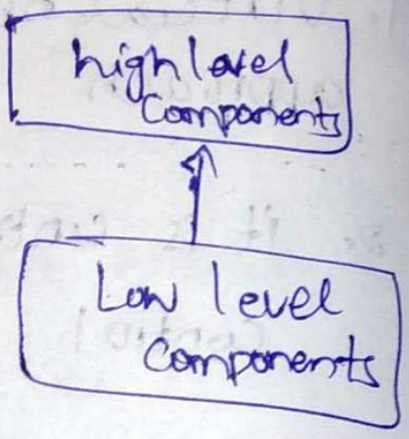
* Traditional Approach:-



Northbound APIs



Southbound APIs



Q. Describe how SDN can be used at various levels of IoT?

* SDN can be used at various levels of IOT to improve the management, scalability and security of IoT networks.

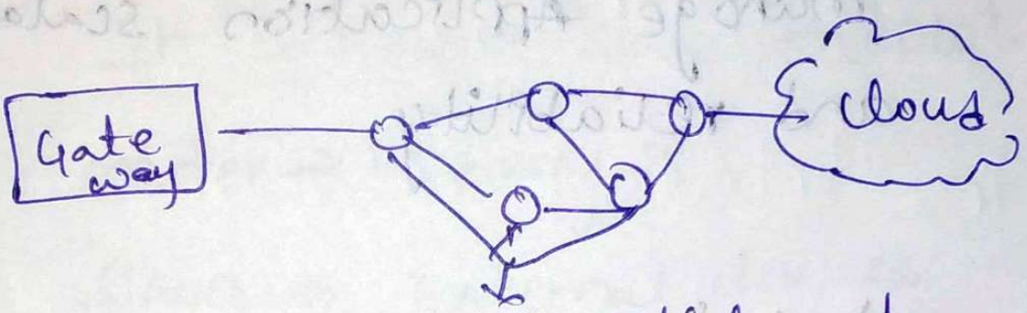
① Device level:-

At this level, SDN is used to control & manage the IOT devices. Such as sensors, actuators and smart devices.

② Edge Level:- SDN can be used to manage the gateways, routers, switches in the n/w.

* It will manage & optimise the data transmission.

* Fog Level:- At this fog level can be used to manage and control the fog nodes (which are intermediate nodes between devices and cloud.).



* Cloud Level:-

SDN can be used to manage the data processing and Analytics at cloud level.

* Manage data processing and analytics in the cloud.
* Optimise data transmission and reduce latency.

* Application Level:

SDN can be used to control and manage the IoT Applications and its services.

- Manage Apps performance.
- manage Application scalability and reliability

Q. What is the use of SPI and I2C interfaces on Raspberry Pi?

The Raspberry Pi ~~provides lot of~~ is a popular single board. Computer that provides several interfaces for interacting with external devices.

* Two of those are SPI (serial peripheral interface) and I2C (Inter Integrated Circuit).

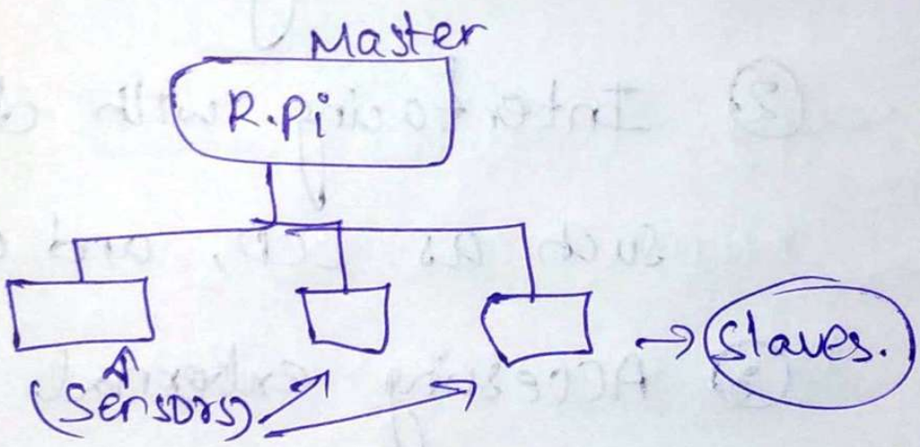
1. SPI interface:

It is a synchronous serial communication protocol that allows the raspberry Pi to communicate with external devices (such as sensors, displays, and memory chips.)

* It is master-slave protocol.

master → Raspberry Pi

Slaves → External devices.



* This interface consist of four pins:

1. SCLK (Serial Clock): generates the clock signal for data transfer.

2. MOSI (master Out slave In):

transfer the data from raspberry pi to external device.

3. MOSO (master In Slave out):-

Raspberry Pi recieves the data from sensors, devices (external).

4. CS (chip Select):-

selects the external device to communicate with.

* This interface is commonly used for:

(1) Communicating with Sensors.

(2) Interfacing with displays, such as LCD, and OLED displays.

(3) Accessing external memory such as flash memory and EEPROM.

* I2C Interface:

The I2C interface is a synchronous serial communication protocol that allows the Raspberry Pi

to communicate with others devices. such as sensors, displays, and micro controllers.

* It is "multi-master" protocol.

* where multiple devices act as masters and communicate with each other.

* It consist of two pins:

1. SCL (Serial Clock): generates the clock signal for data transfer.

2. SDA (Serial Data): transmits and recieves data between devices.

* The I2C interface commonly used for:

- communicating with other devices such as sensors, memory chips, lights, displays, motors..etc.
- communicating with micro-controllers and devices.

SUBSCRIBE

@CodingFatafat

Youtube Channel.

Follow Us on Telegram,

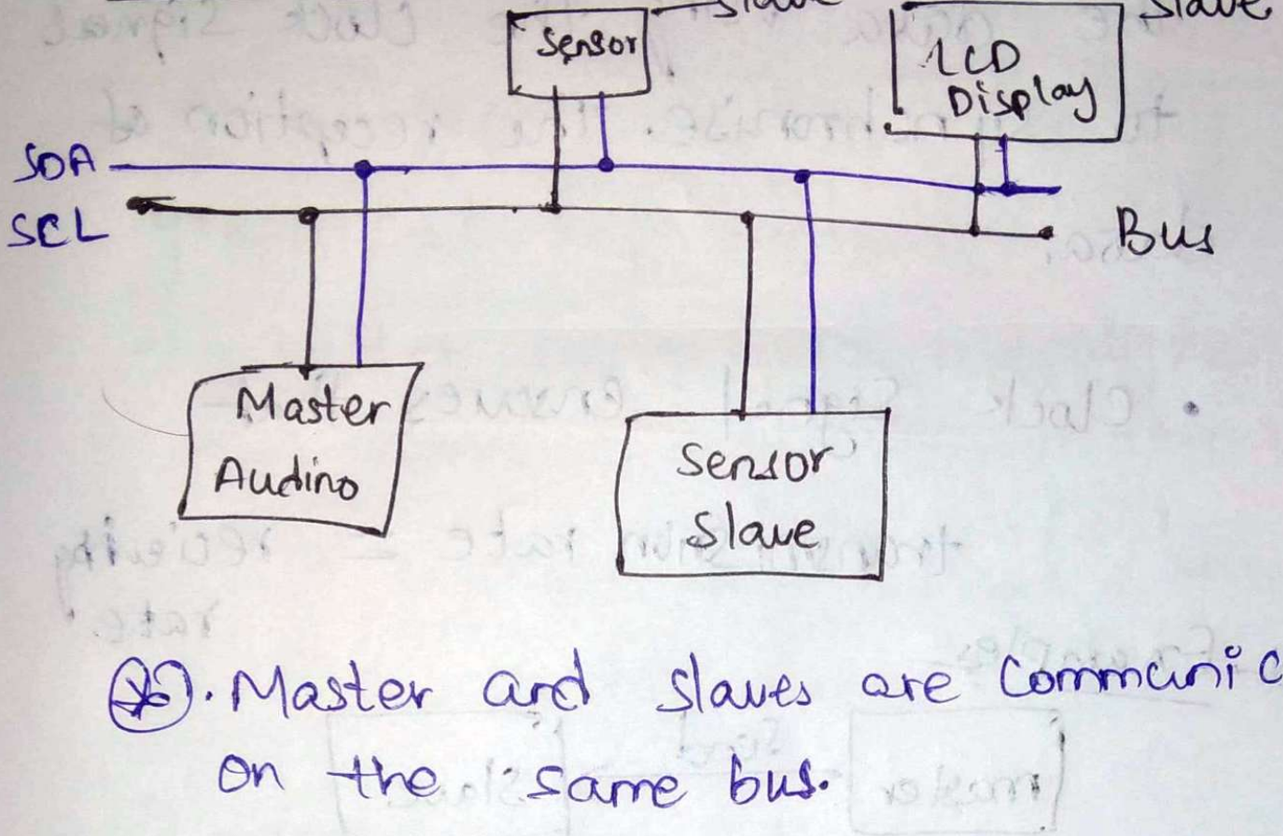
Instagram for Q/A.

SPI	I2C.
• master – slave protocol • It has four pins • high data transfer rate than I2C. • ~~No~~ less flexible than I2C.	• multi-master slave protocol • It has two pins. • Medium – data transfer rate. More flexible than SPI

[*] SPI used for higher speed communication with devices that require a clock signal.

[*] while I2C used for low to medium speed communication with devices that require more flexible and multi-master protocol.

* I2C Diagram :-



(*). Master and slaves are communicate on the same bus.

(**). It contains bi-directional lines for communication.

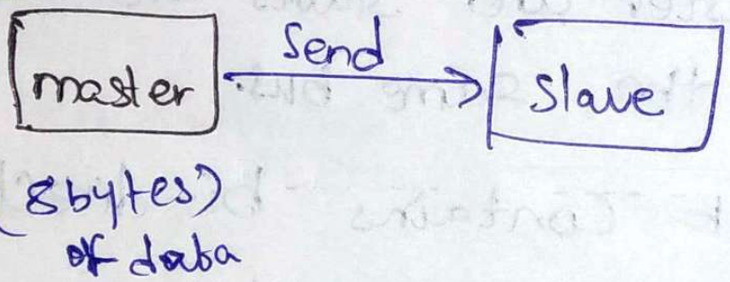
(***). Transmission rate == recieving rate (synchronus)

- first the clock signal is generated from the master to slaves. through the clock line (SCLK, SCL for I2C)
- The master device transmit the data through data line (SDA for I2C, MOSI for SPI).
- The slaves transmit the data back to masters device through the data line

* The master device recieves the the data using the clock signal to synchronise the reception of data:

- Clock Signal ensures that transmission rate = recieving rate

* Examples



⊙ 1byte sent for one cycle

⊙ 8 cycles required for ~~8bytes~~ to send the 8bytes (data).

* Data Visulization:-

It is the process of creating graphical representation of data to better understand, analyze, and communicate information.

- It involves using visual elements like graphs, maps, charts... etc.

* Importance in IoT:-

1. In the context of IoT, data visulization plays a crucial role in extracting the insights from vast ammount of data generated by connected nodes. Such as sensors, actuators, and smart devices.

(*) Data visulization is essential for IoT for several reasons.

1. pattern recognition: Identify the patterns, trends and anomolies (outliers) in IoT data.

2. Real-Time Monitoring:-

real time monitoring allows for prompt detection of issues.

3. Complexity reduction:-

Data Visalization simplifies complex IoT data, maker it easier to understand and analyze even for non-technical stake holders.

* Common techniques visulization techniques in IoT:-

* 1. Time Series Analysis: Visulizing the patterns in the data over time.

2. Scatter plots:- Visulizing relation ship between multiple variables.

3. Gauge charts:- Displaying metrics

like temperature, pressure,

* Tools for IoT Data Visualization:-

- Tableau
- PowerBI
- Matplotlib
- D3.JS

→. The Role of Data Handling and Analytics in IoT:-

**. Data Handling and Analytics plays a critical role in the IOT.

*. Data Handling:- It is the process of managing or Handling the data which is generated from the different IoT sensors.

(1) Data Ingestion:- collecting the data, processing the data from various IoT devices.

(2) Data storage:- Storing and managing large volumes of IoT Data. is Scalable

③. Data processing :-

Cleaning, transforming, and preparing IoT data for analytics.

* Data Analytics :-

After cleaning the data we have to extract the meaning ful insights from the data to make decisions via ~~by~~ fore casting. by observing historical data.

① Descriptive Analytics :-

Analysing historic data to understand what happened.

② predictive Analytics :- forecasts what happen based on the patterns observed in the data.

③. Prescriptive Analytics :- providing Analytics or recommendations for actions to take.

SUBSCRIBE

@CodingFataka

Youtube Channel,

Follow Us on Telegram,

Instagram for Q/A.

😊--!