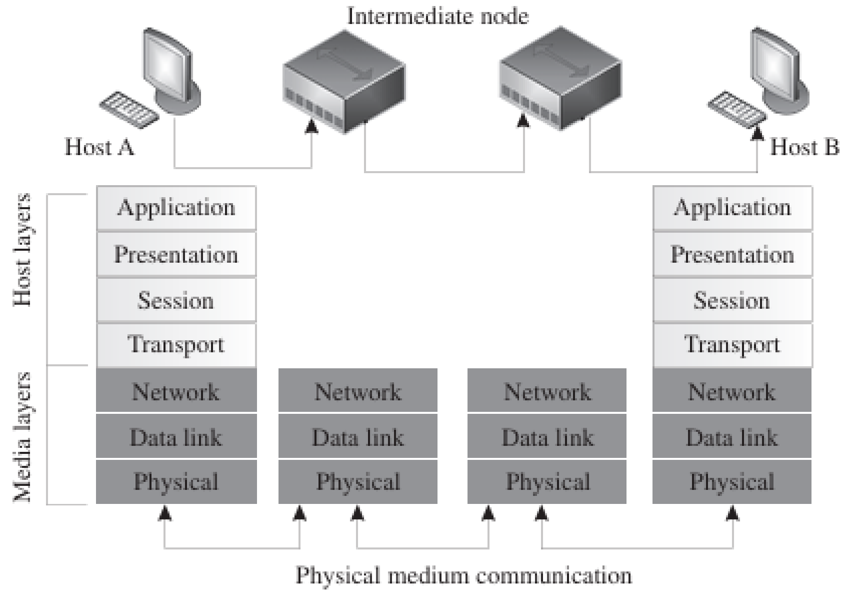
Solution QP – Feb 2024

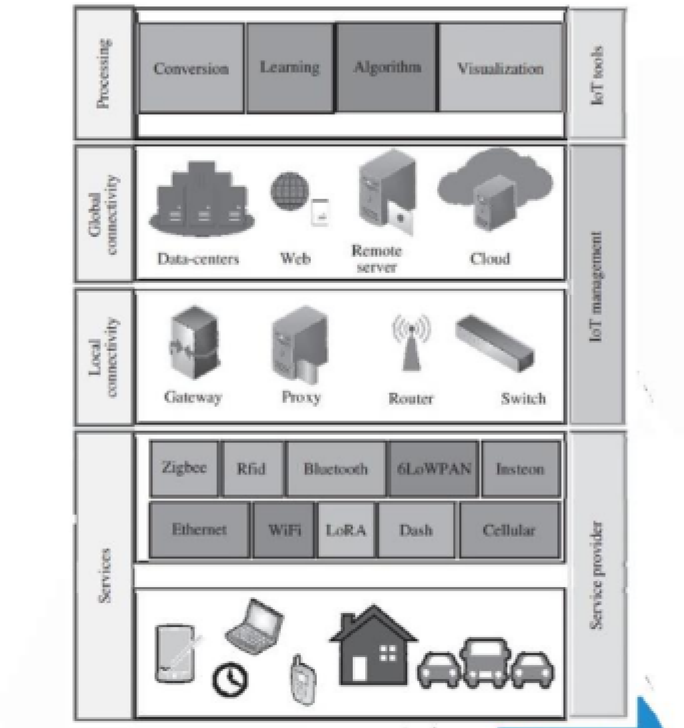
Sub:	Introduction to Internet of Things (IOT)			Sub Code:		BETCK105H	Branch :	CSE,AIML ,ISE,AIDS	
Date:	Feb 2025	Duration:	3 hours	Max Marks:	100	Sec:	I A-F	OBE	

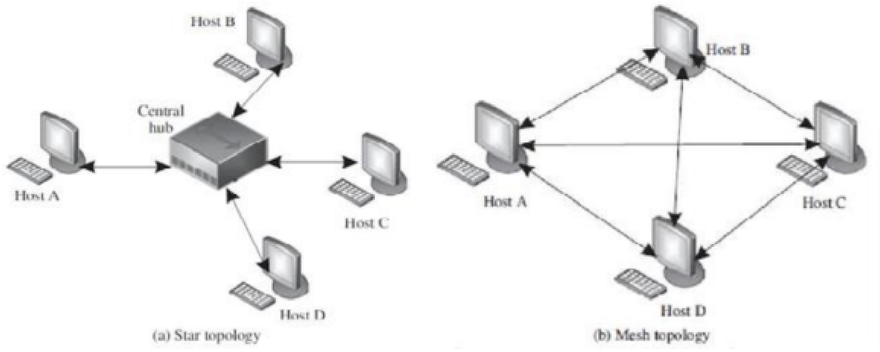
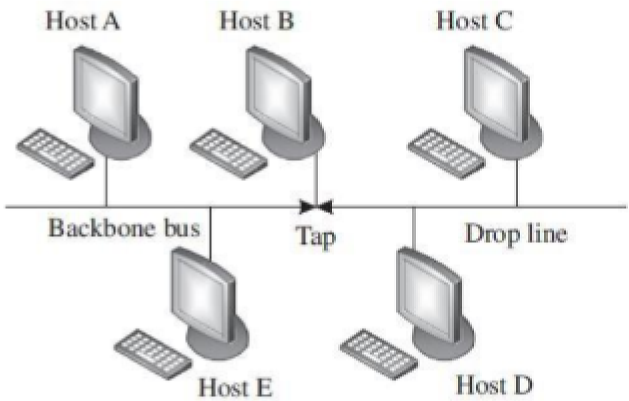
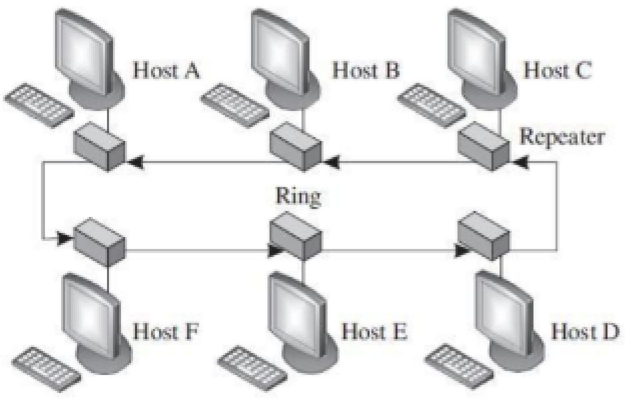
		MARKS	CO	RBT
1	a) With a neat diagram, explain the network communication between two hosts following the OSI model. The ISO-OSI model is a conceptual framework that partitions any networked communication device into seven layers of abstraction, each performing distinct tasks based on the underlying technology and internal structure of the hosts. These seven layers, from bottom-up, are as follows: 1) Physical layer, 2) Data link layer, 3) Network layer, 4) Transport layer, 5) Session layer, 6) Presentation layer, and 7) Application layer. The major highlights of each of these layers are: (i) Physical Layer: This is a media layer and is also referred to as layer 1 of the OSI model. The physical layer is responsible for taking care of the electrical and mechanical operations of the host at the actual physical level. This layer is responsible for the topological layout of the network (star, mesh, bus, or ring), communication mode (simplex, duplex, full duplex), and bit rate control operations. The protocol data unit associated with this layer is referred to as a symbol. (ii) Data Link Layer: This is a media layer and layer 2 of the OSI model. The data link layer is mainly concerned with the establishment and termination of the connection between two hosts, and the detection and correction of errors during communication between two or more connected hosts. The protocol data unit associated with this layer is referred to as a frame. (iii) Network Layer: This layer is a media layer and layer 3 of the OSI model. It provides a means of routing data to various hosts connected to different networks through logical paths called virtual circuits. These logical paths may pass through other intermediate hosts (nodes) before reaching the actual destination host. The primary tasks of this layer include addressing, sequencing of packets, congestion	[8]	CO1	L2

	control, error handling, andInternetworking.Theprotocol data unit associated with this layer is referred to as a packet. (iv) Transport Layer: This is layer 4 of the OSI model and is a host layer. The transport layer is tasked with end-to-end error recovery and flow control to achieve a transparent transfer of data between hosts. This layer is responsible for keeping track of acknowledgments during variable-length data transfer between hosts. In case of loss of data, or when no acknowledgment is received, the transport layer ensures that the particular erroneous data segment is re-sent to the receiving host. The protocol data unit associated with this layer is referred to as a segment or datagram. (v) SessionLayer: ThisistheOSImodel'slayer5andisahostlayer. It is responsible for establishing, controlling, and terminating of communication between networked hosts. The session layer sees full utilization during operations such as remote procedure calls and remote sessions. The protocol data unit associated with this layer is referred to as data. (vi) Presentation Layer: This layer is a host layer and layer 6 of the OSI model. It is mainly responsible for data format conversions and encryption tasks such that the syntactic compatibility of the data is maintained across the network, for which it is also referred to as the syntax layer. The protocol data unit associated with this layer is referred to as data. (vii) Application Layer: This is layer 6 of the OSI model and is a host layer. It is directly accessible by an end-user through software APIs (application program interfaces) and terminals. Applications such as file transfers, FTP (file transfer protocol), e-mails, and other such operations are initiated from this layer. The application layer deals with user authentication, identification of communication hosts, quality of service, and privacy. The protocol data unit associated with this layer is referred to as data			

	 **Figure 1.3** Networked communication between two hosts following the OSI mo			
1B)	Explain the loT planes, various enablers of loT, and the complex interdependencies among them with a neat diagram. IoT is a paradigm built upon complex interdependencies of technologies (both legacy and modern),which occur at various planes. we can divide the IoT paradigm into four planes: services, local connectivity, global connectivity, and processing. The service plane is composed of two parts: 1)things or devices and 2) low-power connectivity. Typically, the services offered in this layer are a combination of things and low-power connectivity. The things may be wearables, computers, smartphones, household appliances, smart glasses, factory machinery, vending machines, vehicles, UAVs, robots, and other such contraptions (which may even be just a sensor). The immediate low-power connectivity, which is responsible for connecting the things in local implementation, may be legacy protocols such as WiFi, Ethernet, or cellular. In contrast, modern- day technologies are mainly wireless and often programmable such as Zigbee, RFID, Bluetooth, 6LoWPAN, LoRA, DASH, Insteon, and others.The range of these connectivity technologies is severely restricted; they are responsible for the connectivity between the things of the IoT and the The local connectivity plane falls under the purview of IoT	[6]	CO1	L2

	management as it directly deals with strategies to use/reuse addresses based on things and applications. The modern-day “edge computing” paradigm is deployed in conjunction with these first two planes: services and local connectivity. In continuation, the penultimate plane of global connectivity plays a significant role in enabling IoT in the real sense by allowing for worldwide implementations and connectivity between things, users, controllers, and applications. This plane also falls under the purview of IoT management as it decides how and when to store data, when to process it, when to forward it, and in which form to forward it. The Web, data-centers, remote servers, Cloud, and others make up this plane. The paradigm of “fog computing” lies between the planes of local connectivity and global connectivity. The final plane of processing can be considered as a top-up of the basic IoT networking framework.The nearest hub or gateway to access the Internet. The local connectivity is responsible for distributing Internet access to multiple local IoT deployments. This distribution may be on the basis of the physical placement of the things, on the basis of the application domains, or even on the basis of providers of services. Services such as address management, device management, security, sleep scheduling, and others fall within the scope of this plan.			

	 **Figure 9: The IoT planes, various enablers of IoT, and the complex interdependencies among**			
1c	Briefly explain various network topologies with suitable diagrams. Depending on the physical manner in which communication paths between the hosts are connected, computer networks can have the following four broad topologies: Star, Mesh, Bus, and Ring. i. Star: In a star topology, every host has a point-to-point link to a central controller or hub. The hosts cannot communicate with one another directly; they can only do so through the central hub. The hub acts as the network traffic exchange. The main advantages of the star topology areeasy installation and the ease of fault identification within the network. However, the main disadvantage of this topology is the danger of a single point of failure. If the hub fails, the whole network fails. ii. Mesh: In a mesh topology, every host is connected to every other host using a dedicated link (in a point-to-point manner). This implies that for n hosts in a mesh, there are a total of n(n-1)/2 dedicated full duplex links between the hosts. This massive number of links makes the mesh topology expensive. However, it offers certain specific advantages over other topologies. Even if a link is down or broken,the network is still fully functional as there remain other pathways for the traffic to flow through. The second advantage is the security and privacy of the traffic as the data is only seen by the intended recipients and not by all members of the network. The third advantage is the reduced data load on a single host, as every host in this network takes care of its traffic load. iii. Bus: A bus topology follows the point-to-multipoint connection.A backbone cable or bus serves as the primary traffic pathway between the hosts. The hosts are connected to the main bus employing drop lines or taps. The main advantage of this topology is the ease of installation. However, there is a restriction on the length of the bus and the number of hosts that can be simultaneously connected to the bus due to signal loss over the extended bus. Multiple drop lines and taps can be used to connect various hosts to the bus, making installation very easy and cheap. However, the main drawback of this topology is the difficulty in fault localization within the network.	[6]	CO1	L2

	iv. Ring: A ring topology works on the principle of a point-to-point connection. Here, each host is configured to have a dedicated point-to-point connection with its two immediate neighboring hosts on either side of it through repeaters at each host. The repetition of this system forms a ring. The repeaters at each host capture the incoming signal intended for other hosts, regenerates the bit stream, and passes it onto the next repeater. Fault identification and set up of the ring topology is quite simple and straightforward. However, the main disadvantage of this system is the high probability of a single point of failure. If even one repeater fails, the whole network goes down.    (c) Bus topology			
2a	With a neat diagram, explain internet protocol suite. The Internet protocol suite is yet another conceptual framework that provides levels of abstraction for ease of understanding and development of communication and networked systems on the Internet. However, the Internet protocol suite predates the OSI model and provides only four levels of abstraction: 1) Link layer, 2) Internet layer, 3) transport layer, and 4) application layer. This collection of protocols is commonly referred to as the TCP/IP protocol suite as the foundation technologies of this suite are transmission control protocol (TCP) and Internet protocol (IP). Link Layer: The first and base layer of the TCP/IP protocol suite is also	[8]	CO1	L2

known as the network interface layer. This layer is synonymous with the collective physical and data link layer of the OSI model. It enables the transmission of TCP/IP packets over the physical medium.

ii. Internet Layer: Layer 2 of the TCP/IP protocol suite is somewhat synonymous to the network layer of the OSI model. It is responsible for addressing, address translation, data packaging, data disassembly and assembly, routing, and packet delivery tracking operations. Traditionally, this layer was built upon IPv4, which is gradually shifting to IPv6, enabling the accommodation of a much more significant number of addresses and security measures.

iii. Transport Layer: Layer 3 of the TCP/IP protocol suite is functionally synonymous with the transport layer of the OSI model. This layer is tasked with the functions of error control, flow control,congestion control, segmentation, and addressing in an end-to-end manner; it is also independent of the underlying network. Transmission control protocol (TCP) and user datagram protocol (UDP)are the core protocols upon which this layer is built, which in turn enables it to have the choice of providing connection-oriented or connectionless services between two or more hosts or networked devices.

iv. Application Layer: The functionalities of the application layer, layer 4, of the TCP/IP protocol suite are synonymous with the collective functionalities of the OSI model's session, presentation, and application layers. This layer enables an end-user to access the services of the underlying layers and defines the protocols for the transfer of data. Hypertext transfer protocol (HTTP), file transfer protocol(FTP), simple mail transfer protocol (SMTP), domain name system (DNS), routing information protocol (RIP), and simple network management protocol (SNMP) are some of the core protocols associated with this layer.

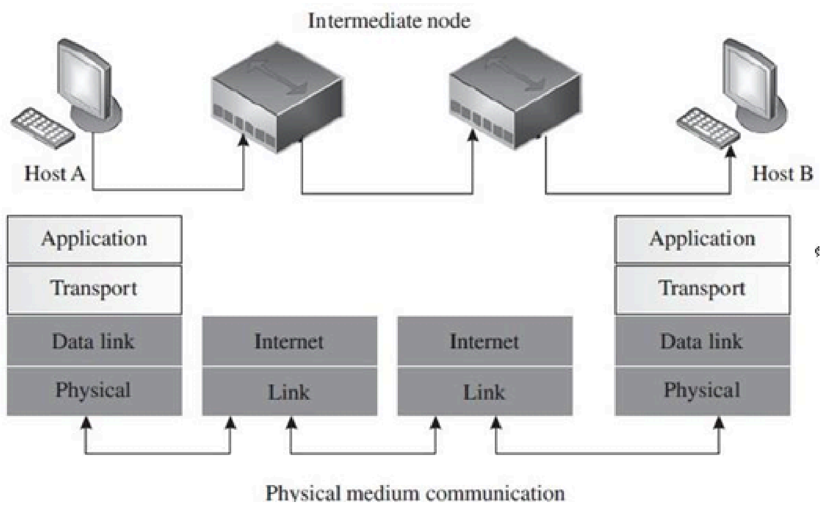
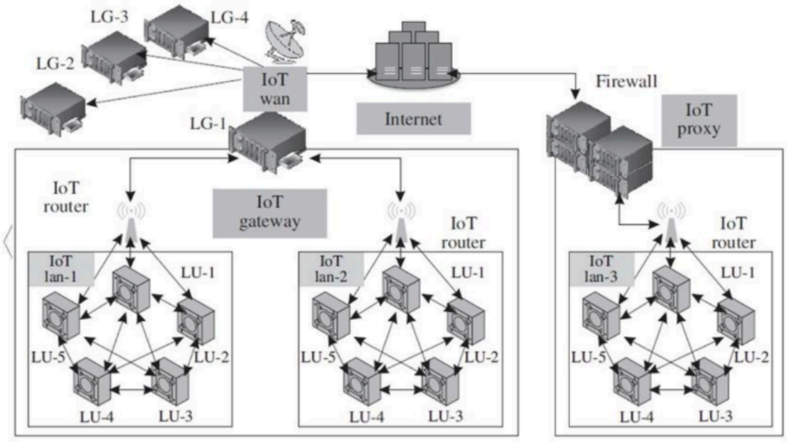


Figure 4: Networked communication between two hosts following the TCP/IP suite

2b	Explain various networking components of loT  **Figure 10: A typical IoT network ecosystem highlighting the various networking components from IoT nodes to the Internet.** i. IoT Node: These are the networking devices within an IoT LAN. Each of these devices is typically made up of a sensor, a processor, and a radio, which communicates with the network infrastructure (either within the LAN or outside it).The nodes may be connected to other nodes inside a LAN directly or by means of a common gateway for that LAN. Connections outside the LAN are through gateways and proxies. ii. IoT Router: An IoT router is a piece of networking equipment that is primarily tasked with the routing of packets between various entities in the IoT network; it keeps the traffic flowing correctly within the network. A router can be repurposed as a gateway by enhancing its functionalities. iii. IoT LAN: The local area network (LAN) enables local connectivity within the purview of a single gateway. Typically, they consist of short-range connectivity technologies. IoT LANs mayor may not be connected to the Internet. Generally, they are localized within a building or an organization. iv. IoT WAN: The wide area network (WAN) connects various network segments such as LANs. They are typically organizationally and geographically wide,with their operational range lying between a few kilometers to hundreds of kilometers. IoT WANs connect to the Internet and enable Internet access to the segments they are connecting. v. IoT Gateway: An IoT gateway is simply a router connecting the IoT LAN to a WAN or the Internet. Gateways can implement several LANs and WANs. Their primary task is to forward packets between LANs and WANs, and the IP layer using only layer. vi. IoT Proxy: Proxies actively lie on the application layer and performs application layer functions between IoT nodes and other entities. Typically, application layer proxies are a means of providing security to the network entities under it ; it helps to extend the addressing range of its network.	[6]	CO1	L2

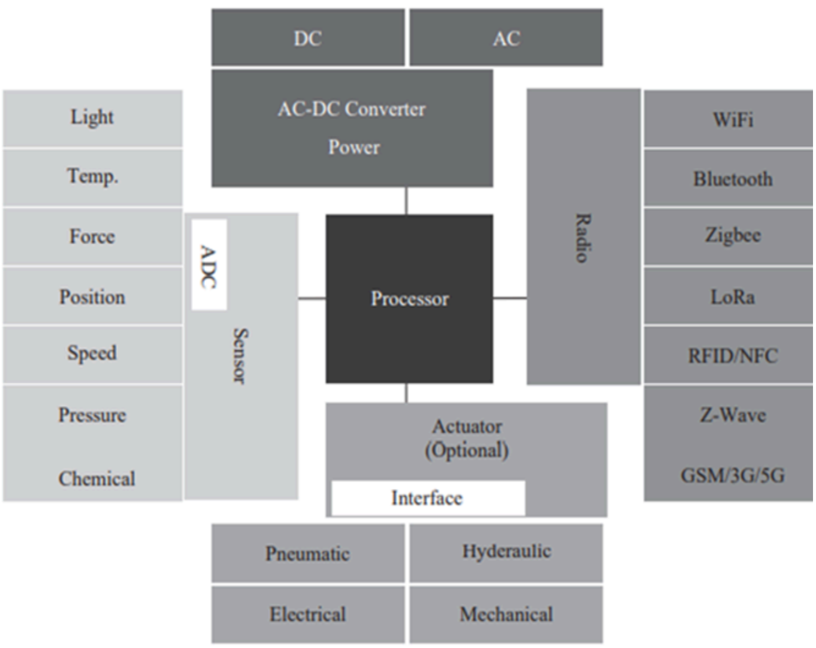
2c	What is loT? Write the characteristics of an IoT system. a The Internet of Things (IoT) is the network of physical objects that contain embedded technology to communicate and sense or interact with their internal states or the external environment.	[6]	CO1	L2

Figure 6: The three characteristic features-anytime, anywhere, and anythi highlight therobustness and dynamic nature of IoT.

IoT is an anytime, anywhere, and anything-network of Internet-connected physical devices or systems capable of sensing an environment and affecting the sensed environment intelligently. This is generally achieved using low-power and low-form-factor embedded processors on-board the "things" connected to the Internet. In other words, IoT may be considered to be made up of connecting devices, machines, and tools; these things are made up of sensors/actuators and processors, which connect to the Internet through wireless technologies.

<table>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>3a</td><td>

With a neat diagram, explain the functional blocks of a typical sensor node in IoT.



Sensing Unit: This block includes various sensors, such as temperature, humidity, pressure, or motion sensors, that detect and measure physical parameters from the environment.

Processing Unit: It consists of a microcontroller or processor that processes data received from the sensors, performs necessary computations, and prepares data for transmission.

Communication Unit: This block enables the sensor node to communicate with other devices, systems, or the cloud. It includes

</td><td>10</td><td>L2</td><td>CO2</td></tr>
</table>

wireless communication modules like Wi-Fi, Bluetooth, Zigbee, or LoRa for sending the processed data.

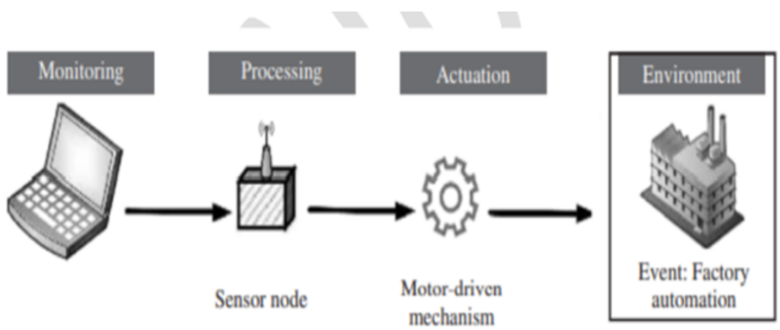
Power Supply: This provides the necessary power for the sensor node to function. It can come from batteries, energy harvesting systems, or even power-over-wireless technologies.

Storage Unit: This is where data can be temporarily or permanently stored. It could be used for buffering data before transmission or saving important information for future processing.

3b	Outline basic differences between transducer, sensor and an actuator.	10	L4	CO2

Aspect	Transducer	Sensor	Actuator
Definition	A device that converts one form of energy to another.	A type of transducer that detects and measures a physical quantity, converting it to an electrical signal.	A device that converts electrical energy into physical motion or action.
Function	Converts energy from one form to another (e.g., mechanical to electrical).	Detects a physical parameter (e.g., temperature, pressure) and converts it into an electrical signal.	Receives an electrical signal and performs a physical task or action (e.g., moving a part).
Output	Can be electrical or physical.	Electrical signal (voltage, current, etc.).	Physical motion, force, or other mechanical output.
Example	Microphone, speaker, thermocouple.	Temperature sensor, pressure sensor, photodiode.	Electric motor, solenoid, hydraulic actuator.
Purpose	Energy conversion (from one form to another).	Measurement of a physical parameter for monitoring or control.	Performing a physical task or movement based on input signals.

4a	With a neat diagram, explain working mechanism of actuator.	10	L2	CO2



Data Collection by Sensors:

- Sensors in an IoT system continuously monitor physical parameters, such as temperature, humidity, pressure, light, or motion. The data from these sensors is sent to a processing unit (e.g., microcontroller or cloud platform).

Data Processing:

- The processing unit analyzes the sensor data to determine whether any action needs to be taken. This could be based on specific conditions, thresholds, or user commands (e.g., if temperature exceeds a certain limit, the actuator must activate).

Sending Command to Actuator:

- Once the processing unit decides that an action is required, it sends a command to the actuator. This command can be transmitted via wireless communication protocols like Wi-Fi, Bluetooth, Zigbee, or LoRa, depending on the system's configuration.

Actuator Response:

- The actuator receives the electrical signal or command from the controller or IoT system.
- It then converts this electrical signal into a physical action. For example:
  - A **motor actuator** might move a robotic arm.
  - A **servo actuator** might adjust the position of a valve.
  - A **solenoid actuator** might open or close a door or valve.
  - A **heater actuator** might turn on or off based on temperature data.

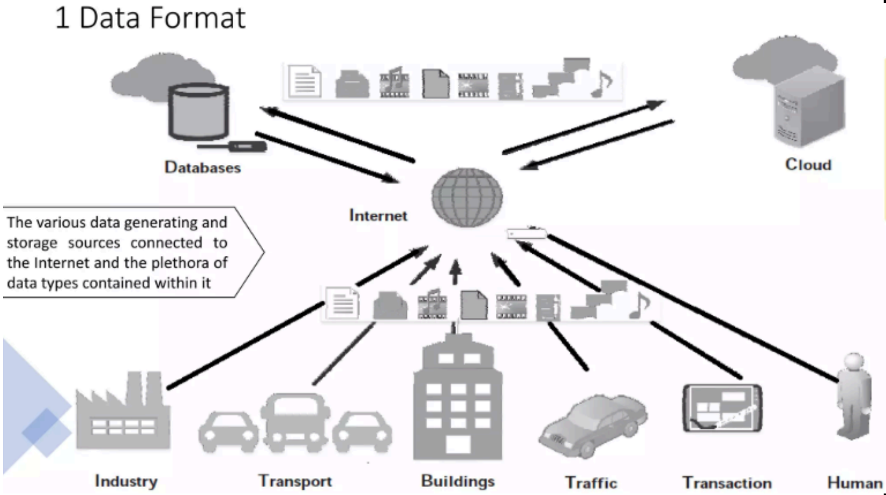
Feedback and Control:

- In many IoT systems, the actuator's action might also be monitored by sensors to provide feedback. For example, after the actuator turns on a fan, temperature sensors could monitor whether the temperature has decreased. If the desired condition is met, the actuator can be turned off automatically.

4b	Compare mechanical, soft and shape memory based actuators.	10	L4	CO2

Aspect	Mechanical Actuators	Soft Actuators	Shape Memory-based Actuators
Definition	Actuators that use mechanical force or motion to perform tasks, typically through gears, motors, or levers.	Actuators made from flexible, deformable materials that can change shape when activated.	Actuators that change shape or move in response to temperature or other external stimuli, utilizing materials with shape memory properties.
Operating Principle	Converts electrical, hydraulic, or pneumatic energy into mechanical motion.	Utilize materials that can deform elastically or plastically in response to stimuli like pressure or temperature.	Rely on materials (like shape memory alloys) that "remember" specific shape and return to it when heated or cooled.
Common Materials	Metals (e.g., steel, aluminum), plastics, or composites.	Elastomers, soft polymers, and gels.	Shape memory alloys (e.g., Nitinol), polymers.
Movement Type	Typically linear or rotary movement.	Often flexible, bending, or stretching motions.	Change in shape (e.g., bending, contracting) based on temperature.
Response Time	Typically fast, with precise movement control.	Generally slower compared to mechanical actuators.	Response time is dependent on the material and temperature change; can be slower.
Power Source	Electric motors, hydraulic fluid, pneumatic pressure.	Pressure (air or liquid), electric current, or thermal energy.	Heat or temperature changes, sometimes electrical current.
Precision and Control	High precision and control, often used in robotics and machinery.	Less precise, suited for applications where flexibility is more important than exact control.	Moderate precision, with shape change being predictable but not as precise as mechanical systems.
Examples	Electric motors, hydraulic cylinders, pneumatic	Pneumatic artificial muscles, soft robots,	Nitinol-based actuators, thermoresponsive polymer

5a).	With a neat diagram, Explain two types of data formats and processing importance in IOT. **Diagram:**	10	L2	CO1



Types of Data Formats in IoT & Their Processing Importance.

1. Structured Data Format

Structured data follows a well-defined schema, making it easy to store and retrieve in databases. Common formats include JSON (JavaScript Object Notation), XML (Extensible Markup Language), and CSV (Comma-Separated Values).

Example (JSON Format)
json
CopyEdit

```
{

  "temperature": 25.5,

  "humidity": 60,

  "timestamp": "2025-02-14T10:30:00Z"

}
```

- Importance in IoT Processing
  - Efficient Querying: Structured data enables fast searching and filtering in databases.
  - Interoperability: JSON and XML are widely used in IoT applications for seamless communication between devices.
  - Easy Storage & Analysis: Structured formats are ideal for storing sensor data in cloud databases.

---

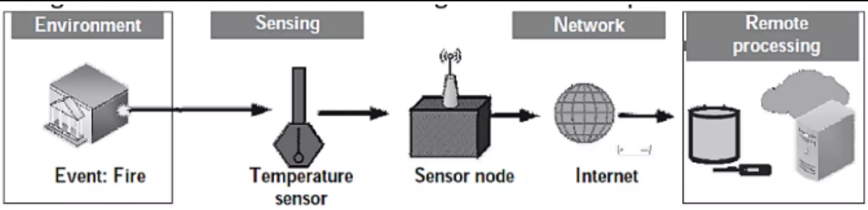
2. Unstructured Data Format

Unstructured data does not follow a predefined schema and includes data types such as images, videos, logs, and raw sensor readings.

- Example
  - Camera Footage from a Smart Surveillance System
  - Raw Audio Data from a Voice Sensor
- Importance in IoT Processing

	○ AI & ML Applications: Image recognition and audio processing enable advanced IoT functionalities. ○ Predictive Maintenance: Analyzing unstructured sensor logs can detect faults before failure. ○ Security & Monitoring: Unstructured video streams aid in real-time surveillance.	10	L2	CO1
5b).	Explain IoT Device selection considerations. 1. Device Compatibility & Interoperability ● Must support standard communication protocols (e.g., MQTT, CoAP, HTTP, Bluetooth, Zigbee). ● Should integrate seamlessly with existing IoT platforms and cloud services. ● Ensure support for multiple operating systems (Linux, Windows, RTOS) if needed. Example: A smart thermostat should be able to communicate with both mobile apps and home automation systems like Google Home or Alexa. --- 2. Power Consumption & Battery Life ● Devices should be optimized for low power consumption, especially for battery-powered IoT devices. ● Consider devices supporting low-power communication protocols (LoRaWAN, Zigbee, NB-IoT). ● Energy-efficient components (e.g., ARM Cortex M processors) are preferred for longevity. Example: Wearable fitness trackers must have long battery life and low power consumption. --- 3. Connectivity & Network Requirements ● Wired vs. Wireless: Choose between Ethernet (wired) or Wi-Fi, 4G/5G, LoRaWAN, and NB-IoT (wireless). ● Ensure network coverage and signal strength in the deployment area. ● Latency considerations: Real-time applications like healthcare monitoring require ultra-low latency networks. Example: A smart irrigation system in agriculture might require LPWAN (LoRaWAN) for long-range connectivity. --- 4. Processing Power & Storage	[10]	CO3	L2

	● Devices must have sufficient processing capability (MCU vs. MPU) based on application needs. ● Edge computing is useful for reducing cloud dependency and enhancing local processing. ● Storage considerations: Devices collecting high-resolution images/videos require more onboard storage. Example: A video surveillance IoT device needs high processing power and storage for local video analytics. --- 5. Security & Data Protection ● Must support encryption protocols (TLS, AES) to protect data transmission. ● Devices should have secure boot mechanisms to prevent unauthorized firmware modifications. ● Authentication features (biometric, password, or key-based access) should be included. Example: A smart home security camera should encrypt live feeds and use two-factor authentication for access. --- 6. Scalability & Future Upgradability ● Devices should support firmware updates (OTA - Over-the-Air) for future security patches and upgrades. ● Should be modular and scalable, allowing new sensors or features to be added. ● Cloud compatibility ensures future expansion of IoT networks. Example: A smart city lighting system should be easily upgradable to add new sensors or AI-based automation. --- 7. Cost & Return on Investment (ROI) ● Consider initial costs vs. long-term operational costs (maintenance, power consumption). ● Subscription & licensing fees for IoT platforms or cloud storage should be factored in. ● Devices should provide a clear business value or operational efficiency improvement. Example: In industrial IoT, selecting rugged, durable sensors with lower maintenance needs can save long-term costs.			
6 a)	Discuss with a neat diagram event detection using offsite using remote and collaborative processing topologies. Diagram:			



Event detection using an off-site remote processing topology

Event detection in IoT refers to the process of identifying and responding to specific occurrences (e.g., temperature rise, motion detection, equipment failure) using sensor data. Two major topologies for offsite event detection are Remote Processing and Collaborative Processing.

---

1. Remote Processing Topology

In remote processing, IoT devices transmit raw or pre-processed data to a centralized cloud or data center, where event detection and decision-making take place.

Workflow:

1. IoT sensors collect real-time data.
2. Data is transmitted to the cloud or a remote server via Wi-Fi, 4G/5G, LoRaWAN, or MQTT.
3. The cloud platform processes the data using AI/ML models for event detection.
4. Alerts or actions are sent back to the IoT devices or end-users.

Advantages:

✔️ High computational power for event detection.
✔️ Centralized data storage & long-term analytics.
✔️ Simplifies device hardware requirements.

Disadvantages:

- High network dependency and latency.
- Bandwidth consumption due to continuous data transmission.

Use Case Example:

A smart city traffic monitoring system where cameras send footage to the cloud for congestion and accident detection.

---

2. Collaborative Processing Topology

In collaborative processing, edge devices (IoT gateways, edge servers, or fog nodes) share processing workloads with the cloud, reducing latency and improving efficiency.

Workflow:

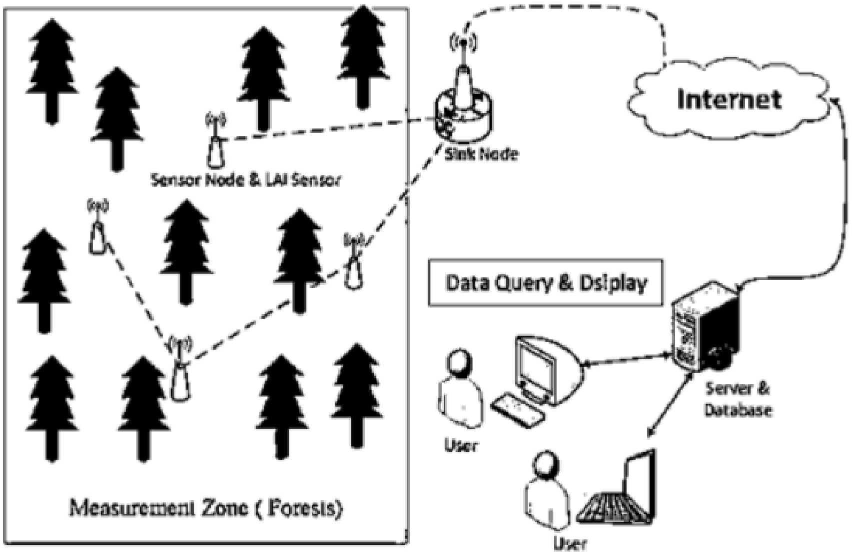
1. IoT sensors collect data and perform initial local filtering or processing.

	2. Edge devices (e.g., local servers, fog nodes, or AI-capable IoT devices) process event-related data. 3. Only critical data or detected events are transmitted to the cloud. 4. The cloud aggregates and refines event detection for large-scale decision-making. Advantages: ✔️ Low latency event detection due to local processing. ✔️ Reduced bandwidth usage, as only processed data is transmitted. ✔️ Increased reliability, since local processing reduces dependency on cloud connectivity. Disadvantages: ● Requires more advanced edge devices with computational capabilities. ● Complexity in managing distributed processing nodes. Use Case Example: An industrial predictive maintenance system, where local edge gateways analyze vibration and temperature data from machines to detect early signs of failure before sending alerts to the cloud.			
6 b)	Explain different data offloading strategies with location and decision making. Data offloading in IoT refers to transferring data processing tasks from resource-constrained devices to more powerful computing entities (edge servers, cloud servers, or fog nodes) to enhance performance and efficiency. Different offloading strategies depend on where the processing occurs and how decisions are made. --- 1. Location-Based Data Offloading Strategies Data offloading can occur at different locations within an IoT ecosystem, affecting latency, bandwidth, and computational efficiency. a) Edge Offloading (Near the Device) ● Where? Data is processed on local edge devices (gateways, routers, or microservers). ● Best for: Real-time applications with low latency needs. ● Example: Smart cameras performing face recognition locally before sending only metadata to the cloud. b) Fog Offloading (Between Edge and Cloud) ● Where? Data is processed at fog nodes (local servers closer to devices but more powerful than edge devices). ● Best for: Applications needing moderate processing with some cloud support. ● Example: Industrial IoT (IIoT) systems analyzing machine sensor data at fog nodes to predict maintenance needs.			

	c) Cloud Offloading (Remote Processing) ● Where? Data is sent to centralized cloud servers for processing. ● Best for: Applications requiring heavy computation and long-term storage. ● Example: Smart healthcare systems where patient health data is analyzed in cloud AI models. --- 2. Decision-Based Data Offloading Strategies The decision to offload data can be static (predefined rules) or dynamic (real-time conditions). a) Full Offloading (All Data is Sent for Processing) ● Decision: The device always offloads 100% of its data to the cloud or edge. ● Advantage: No need for complex decision-making algorithms. ● Disadvantage: High bandwidth usage, potential latency issues. ● Use Case: IoT security cameras streaming continuous video to a cloud-based AI for surveillance analysis. b) Partial Offloading (Selective Data Transmission) ● Decision: Only critical or high-priority data is offloaded. ● Advantage: Reduces bandwidth usage and optimizes resource allocation. ● Disadvantage: Requires intelligent filtering mechanisms on the device. ● Use Case: A smart city traffic system analyzing road congestion locally and sending only alerts to cloud servers.			
7a	What is service level agreement (SLA)? Explain its importance and merits used while defining SLA. A Service Level Agreement (SLA) is a formal contract between a service provider and a customer that defines the level of service expected. It outlines key performance indicators (KPIs), service quality, responsibilities, and penalties for non-compliance. SLAs are commonly used in IT services, cloud computing, customer support, and managed services to ensure consistent and reliable service delivery. **Importance of SLA** Clarifies Expectations – Clearly defines the services, performance standards, and responsibilities of both parties. Improves Service Quality – Encourages the provider to maintain a high level of performance. Enhances Accountability – Establishes penalties or compensation if the service provider fails to meet agreed standards. Minimizes Disputes – Provides a legal framework that prevents misunderstandings and conflicts. Ensures Business Continuity – Guarantees service availability, response time, and resolution time, which are crucial for business operations. **Merits Used While Defining SLA** Service Scope – Clearly defines what services are covered under the agreement.	10	CO4	L2

	Performance Metrics – Includes measurable standards like uptime (e.g., 99.9% availability), response time, and resolution time. Roles and Responsibilities – Specifies duties of both the service provider and the client. Monitoring and Reporting – Details how service performance will be measured and reported. Escalation Process – Defines steps for issue resolution if services fall below agreed standards. Penalties and Remedies – Specifies consequences for failing to meet SLA commitments, such as refunds or service credits. Review and Updates – Allows periodic revisions to reflect changes in business needs and service requirements.			
7b	Explain cloud models and its features of commercial cloud Amazon Web Servers (AWS) Cloud computing is categorized into three main models based on the service provided: Infrastructure as a Service (IaaS) Platform as a Service (PaaS) Software as a Service (SaaS) Cloud Models Explanation 1. Infrastructure as a Service (IaaS) Provides virtualized computing resources over the internet. Users can access virtual machines (VMs), storage, networking, and OS on-demand. Example: Amazon EC2, Google Compute Engine, Microsoft Azure Virtual Machines Features: ✔ Scalable and flexible computing resources ✔ Pay-as-you-go pricing ✔ Full control over virtual machines ✔ Ideal for IT administrators and developers 2. Platform as a Service (PaaS) Provides a development and deployment environment for applications. Users get a platform to build, test, and deploy applications without managing infrastructure. Example: AWS Elastic Beanstalk, Google App Engine, Microsoft Azure App Services Features: ✔ Supports multiple programming languages ✔ Automates OS updates and system maintenance ✔ Scalable and cost-efficient ✔ Ideal for developers 3. Software as a Service (SaaS) Provides ready-to-use software applications over the internet. Users don’t need to install, maintain, or update software manually. Example: Google Workspace, Microsoft 365, Dropbox Features: ✔ Accessible from any device with an internet connection ✔ No need for software installation or maintenance ✔ Subscription-based pricing ✔ Ideal for end-users	10	CO4	L2

8a.	What is virtualization? Explain its advantages from end-user and service provider point of view. Virtualization is a technology that allows multiple virtual instances of operating systems, applications, or servers to run on a single physical machine. It is achieved through a software layer called a hypervisor, which creates and manages these virtual machines (VMs). From an End-User Perspective: Cost Savings – Users can access virtual desktops or applications without needing expensive hardware. Flexibility and Accessibility – Users can access their virtual machines or applications from anywhere, on any device. Enhanced Security – Virtual desktops provide a controlled and secure environment, reducing the risk of malware infections. Improved Performance – Applications run smoothly with centralized management and optimized resource allocation. Disaster Recovery – Users' data and applications can be quickly restored in case of system failure. From a Service Provider Perspective: Efficient Resource Utilization – Virtualization allows providers to maximize server utilization, reducing the need for excess hardware. Scalability – Service providers can easily allocate or scale resources based on demand. Reduced Operational Costs – Less physical infrastructure means lower power, cooling, and maintenance costs. Faster Deployment – Virtual machines can be created, cloned, or migrated quickly, improving service efficiency. High Availability and Reliability – Virtualization enables redundancy and load balancing, ensuring continuous service delivery.	10	CO4	L2
8b	With a diagram, briefly explain the architecture of the leaf Area Index system. The Leaf Area Index (LAI) system architecture consists of several key components designed to measure and analyze canopy structure. It typically includes a light sensor (such as a ceptometer or sunfleck sensor) to capture the amount of sunlight passing through the foliage, along with a canopy analyzer or digital camera to record leaf coverage. These sensors send data to a processing unit, which calculates the LAI based on light attenuation or image analysis. In some advanced systems, a wireless communication module enables real-time data transmission to cloud storage or remote servers for further analysis. This system helps researchers and farmers assess plant growth, optimize irrigation, and improve crop yields.	10	CO4	L2

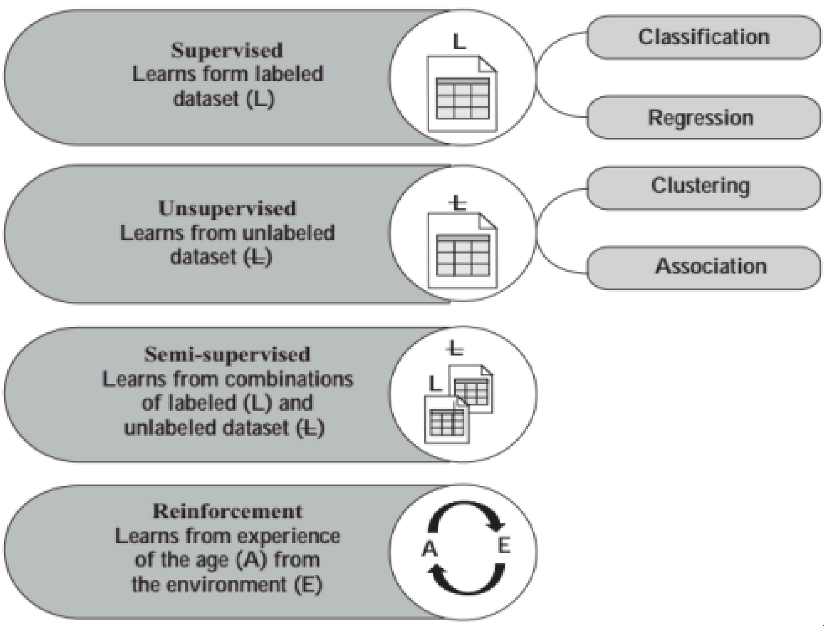


9 a	With a neat diagram, explain the categories of Machine Learning and its advantages.	10	CO5	L2

Sol:

Typically, ML algorithms consist of four categories:
(i) Supervised
(ii) Unsupervised
(iii) Semi-supervised
(iv) Reinforcement Learning

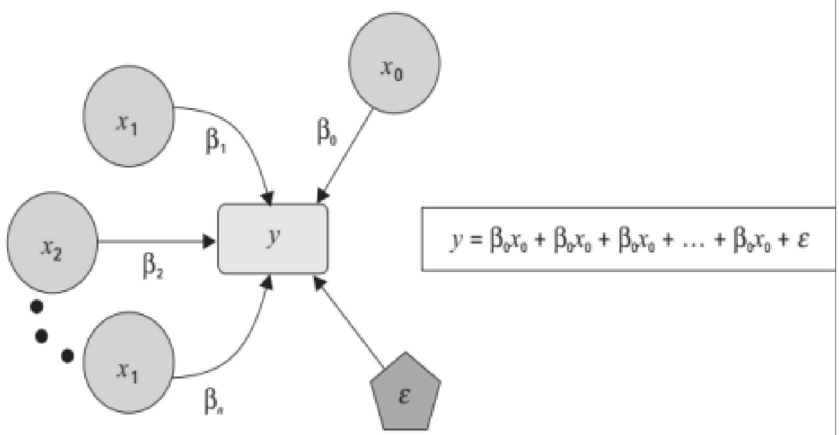
Labeled data contain certain meaningful tags, known as labels. Typically, the labels
correspond to the characteristics or properties of the objects.
For example, in a dataset containing the images of two birds, a particular sample is tagged
as a crow or a pigeon.
The unlabeled dataset does not have any tags associated with them.
For example, a dataset containing the images of a bird without mentioning its name.



1. Supervised Learning:

1. This type of learning supervises or directs a machine to learn certain activities using labeled datasets.
2. Consider an example of a student who tries to learn to solve equations using a set of labeled formulas.
3. The labels indicate the formulae necessary for solving an equation.
4. The student learns to solve the equation using suitable formulae from the set.
5. In the case of a new equation, the student tries to identify the set of formulae necessary for solving it.
6. Similarly, ML algorithms train themselves for selecting efficient formulae for solving equations.
7. Supervised ML algorithms are popular in solving classification and regression problems.
8. The classification deals with predictive models that are capable of approximating a mapping function from input data to categorical output.
9. The Regression provides the mapping function from input data to numerical output.
10. We use regression to estimate the relationship among a set of dependent variables with independent variables, as shown in Fig 11.
11. The dependent variables are the primary factors that we want to predict.

Let x and y be the independent and dependent variables, respectively. Mathematically, a simple regression model is represented as:



$y = \beta0x0 + \beta x + \varepsilon$

where β represents the amount of impact of variable x on y and denotes an error.

Similarly, for multiple variables, say n, the regression model is represented as:

$y = \sum\beta ixi + \varepsilon$

2. Unsupervised Learning:

1. Unsupervised learning algorithms use unlabelled datasets to find scientific trends.
2. Consider an example of the student similar to that described in the case of supervised
3. learning, and illustrate how it differs in case of unsupervised learning.
4. ML algorithms in this category try to identify the nature and properties of the input equation
5. and the nature of the formulae responsible for solving it.
6. Unsupervised learning algorithms try to create different clusters based on the features ofthe formulae and relate it with the input equations.

	7. Unsupervised learning is usually applied to solve two types of problems: clustering and association. 8. Clustering divides the data into multiple groups. In contrast, association discovers the relationship or association among the data in a dataset. **3. Semi-Supervised Learning:** 1. Semi-supervised learning belongs to a category between supervised and unsupervised learning. 2. Algorithms under this category use a combination of both labeled and unlabeled datasets for training. 3. Labeled data are typically expensive and are relatively difficult to label correctly. 4. Unlabeled data is less expensive than labeled data. 5. Unsupervised learning is usually applied to solve two types of problems: clustering and association. 6. Traditionally, semi-supervised learning uses mostly unlabeled data, which makes it efficient to use, and capable of overcoming samples with missing labels. **4. Reinforcement Learning:** 1. Reinforcement learning establishes a pattern with the help of its experiences by interacting with the environment. 2. It aims to achieve a particular goal in an uncertain environment. 3. Typically, the model starts with an initial state of a problem, for which different solutions are available. 4. Based on the output, the model receives either a reward or a penalty from the environment. 5. The output and reward act as inputs for proceeding to the next state. 6. Thus, reinforcement learning models continue learning iteratively from their experiences while inducing correctness to the output.			
9b Sol	With the help of a block diagram explain the components of IoT Health care. A typical IoT healthcare architecture is composed of several components that are essential to generate the whole architecture. Figure depicts different components and their usage in an IoT healthcare system. Each of these components plays a distinct role in the smooth execution of the system as a whole. **1. Sensors:** Layer 1 mainly consists of physiological sensors that collect the physiological parameters of the patient. Few commonly used physiological sensors and their uses are depicted in Table 1. **2. Wireless Connectivity:** ● The communication between the wearable sensors and the LPU is through either wired or wireless connectivity. ● The wireless communication between the physiological sensors and LPU occurs with the help of Bluetooth and ZigBee. ● The communication between the LPU and the cloud or server takes place with Internet connectivity such as WiFi and WLAN. ● For example, when a service is received by a cellphone, it uses GSM (global system for mobile communications). On the other hand, if the same service is received on a desktop, it can be through Ethernet or Wi-Fi.	10	CO5	L2

3. Privacy and Security:

- Moreover, between LPU and the server/cloud, different networking devices work via network hops (from one networked device to another) to transmit the data.
- If any of these devices are compromised, it may result in the theft of health data of a patient, leading to serious security breaches and ensuing lawsuits.
- In order to increase the security of the healthcare data, different healthcare service providers and organizations are implementing healthcare data encryption and protection schemes.

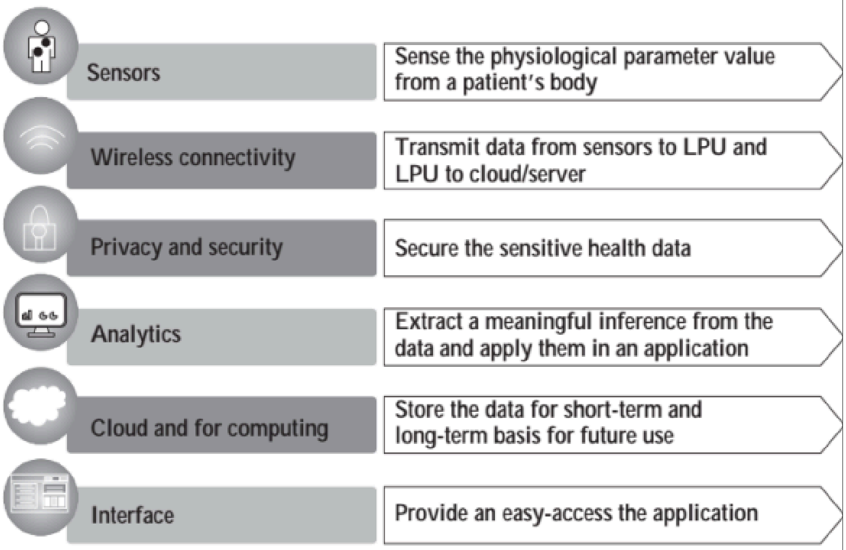


Fig 6: Components of healthcare IoT

4. Analytics:

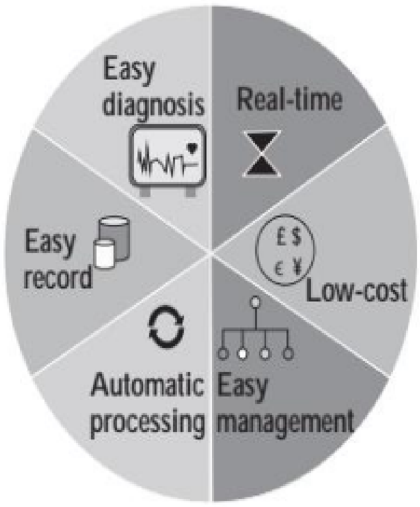
- For converting the raw data into information, analytics plays an important role in healthcare IoT.
- Several actors, such as doctors, nurses, and patients, access the healthcare information in a different customized format.
- Analytics plays a vital role in providing different actors in the system access to meaningful information extracted from the raw healthcare data.
- Analytics is also used for diagnosing a disease from the raw physiological data available.

5. Cloud and Fog Computing:

- For storing these huge amounts of heterogeneous health data, efficient storage space is essential.
- These data are used for checking the patient's history, current health status, and future for diagnosing different diseases and the symptoms of the patient.
- To store health data in a healthcare IoT system, cloud storage space is used.
- The major challenges in storage are security and delay in accessing the data.

6. Interface:

- Healthcare IoT is a very crucial and sensitive application.
- Thus, the user interface must be designed in such a way that it can depict all the required information clearly and, if necessary, reformat or represent it such that it is easy to understand.

	● To store health data in a healthcare IoT system, cloud storage space is used. ● An interface must also contain all the useful information related to the services.			
10 a Sol	Explain the advantages and risks associated with Healthcare IOT Healthcare IoT helps in managing different healthcare subsystems efficiently. Although it has many advantages, healthcare IoT has some risks too, which may be crucial in real-life applications.  Fig 7: (a) Advantages in healthcare IoT  Fig 7: (b) Risk in healthcare IoT **1. Real Time:** ● One of the important characteristics of an IoT-based healthcare system is real-timeliness. ● A healthcare IoT system enables users, such as doctors, end users at the patient-side, and staff in a healthcare unit, to receive real-time updates about the healthcare IoT components. ● A healthcare IoT system can enable a doctor to observe a patient's health condition in real-time even from a remote location, and can suggest the type of care to be provided to the patient. ● On the other hand, users at the patient-end can easily take different decisions, such as where to take a patient during critical situations. ● The staff in a healthcare unit are better aware of the current situation of their unit, which includes the number of patients admitted, availability of the doctors and bed, total revenue of the unit, and other such information. **2. Low Cost:** ● An authorized user can easily find the availability of the beds in a hospital with simple Internet connectivity and a web-browser-based portal. ● Moreover, multiple registered users can retrieve the same information simultaneously **3. Easy Management:** ● Healthcare IoT is an infrastructure that brings all its end users under the same umbrella to provide healthcare services. ● The management of numerous tangible and intangible entities (such as users, medical devices, facilities, costs, and security) is a challenging task. However, healthcare IoT facilitates easy and robust management of all the entities.	10	L2	CO5

<table>
<tr>
<td></td>
<td>
4. Automatic Processing:

Automatic processing features can remove such manual intervention with a fingerprint sensor/device.
Healthcare IoT enables end-to-end automatic processing in different units and also consolidates the information across the whole chain: from a patient’s registration to discharge.

5. Easy Record Keeping:

A healthcare IoT enables the user to keep these records in a safe environment and deliver them to the authorized user as per requirement.
Moreover, these recorded data are accessible from any part of the globe.

6. Easy Diagnosis:

For diagnosing a disease, a huge chunk of prior data is required.
The diagnosis of the disease becomes easier with the help of certain learning mechanisms along with the availability of prior datasets.

Risk in Healthcare IoT
In a healthcare IoT system, there are multiple risks as well.
1. Loss of Connectivity:

Intermittent connectivity may result in data loss, which may result in a life- threatening situations for the patient.
Proper and continuous connectivity is essential in a healthcare IoT system.

2. Security:

The healthcare system must keep the data confidential.
On the other hand, different persons and devices are associated with a healthcare IoT system. In such a system, the risk of data tampering and unauthorized access is quite high.

3. Error:

In the healthcare system, errors in data may lead to misinterpretation of symptoms and lead to the wrong diagnosis of the patient.
It is a challenging task to construct an error-free healthcare IoT architecture.

</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 b

Sol</td>
<td>
Explain the fog framework of intelligent public safety in vehicular environment with a block diagram.

The primary aim of this system is to ensure smart transportation safety (STS) in public bus services. The system works through the following three steps:

The vehicle is equipped with a smart surveillance system, which is capable of executing video processing and detecting criminal activity in real time.
A fog computing architecture works as the mediator between a vehicle and a police vehicle.
A mobile application is used to report the crime to a nearby police agent

The architecture of the fog-FISVER consists of different IoT components. The developers utilized the advantages of the low-latency fog computing architecture for designing their system.
Fog-FISVER is based on a three-tiered architecture, as shown in Fig 4.
</td>
<td>10</td>
<td>L2</td>
<td>CO5</td>
</tr>
</table>

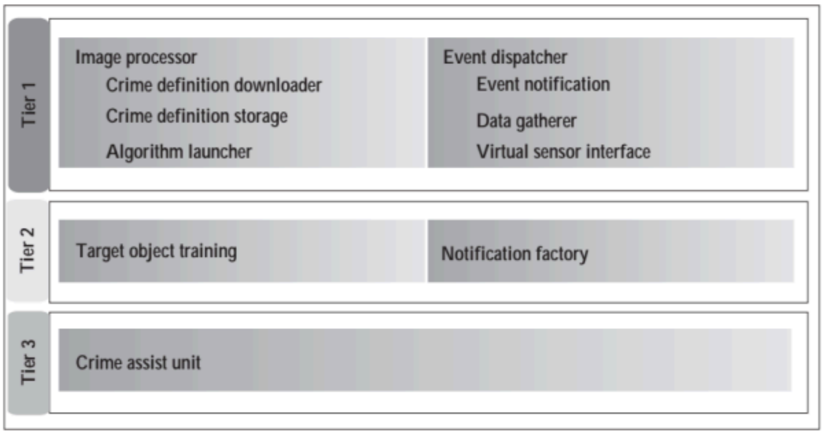


Fig 4: Architecture of Fog-FISVER

1. Tier 1--- In-vehicle FISVER STS Fog :

- A fog node is placed for detecting criminal activities. This tier accumulates the real sensed data from within the vehicle and processes it to detect possible criminal activities inside the vehicle.
- This tier is responsible for creating crime-level metadata and transferring the required information to the next tier.

Tier 1 consists of two subsystems: Image processor and event dispatcher

Image Processor:

- The image processor inside Tier 1 is a potent component, which has a capability similar to the human eye for detecting criminal activities.
- Developers of the system used a deep-learning-based approach for enabling image processing techniques in the processor.
- To implement the fog computing architecture in the vehicle, a Raspberry-Pi-3 processor board is used, which is equipped with a high-quality camera.
- This architecture uses template matching and correlation to detect the presence of dangerous articles (such as a pistol or a knife) in the sub-image of a video frame.

The image processor is divided into the following three parts:
a) Crime definition downloader: This component periodically checks for the presence of new crime object template definitions in fog-FISVER STS fog infrastructure. If a new crime object template is available, it is stored locally.
b) Crime definition storage: In order to use template matching, the crime object template definition is required to be stored in the system. The crime definition storage is used to store all the possible crime object template definitions.
c) **Algorithm launcher:** This component initiates the instances of the registered algorithm in order to match the template with the video captured by the camera attached in the vehicles. If a crime object is matched with the video, criminal activity is confirmed.

Event dispatcher:
The event dispatcher is responsible for accumulating the data sensed from vehicles and the image processor. After the successful detection of criminal activity, the information is sent to the fog-FISVER STS fog infrastructure. The components of the event dispatcher are as follows:
a) **Event notifier:** It transfers the data to the fog-FISVER STS fog infrastructure, after receiving it from the attached sensor nodes in the vehicle.
b) **Data gatherer:** This is an intermediate component between the event notifier and the physical sensor; it helps to gather sensed data.

	c) **Virtual sensor interface:** Multiple sensors that sense data from different locations of the vehicle are present in the system. The virtual sensor interface helps to maintain a particular procedure to gather data. This component also cooperates to register the sensors in the system. **2. Tier 2--- FISVER STS Fog Infrastructure :** Tier 2 works on top of the fog architecture. Primarily, this tier has three responsibilities— keep updating the new object template definitions, classifying events, and finding the most suitable police vehicle to notify the event. FISVER STS fog infrastructure is divided into two sub-components. **(i) Target Object Training:** ● This subcomponent of Tier 2 is responsible for creating, updating, and storing the crime object definition. ● The algorithm launcher uses these definitions in Tier 1 for the template matching process. ● The template definition includes different features of the crime object such as color gradient and shape format. A new object definition is stored in the definition database. ● The database requires to be updated based on the availability of new template definitions. **(ii) Notification Factory:** ● This sub-component receives notification about the events in a different vehicle with the installed system. ● Further, this component receives and validates the events. In order to handle multiple events, it maintains a queue. 3. **Tier 3** consists of mobile applications that are executed on the users' devices. The application helps a user, who witnesses a crime, to notify the police.			

COURSE INSTRUCTOR CCI HOD