Code 417

Under NSQF

artificial intelligence

A Textbook of Class IX

Part A: Employability Skills

Part B: Vocational Skills

SUMITA ARORA

DHANPAT RAI & Co.

VOCATIONAL SKILLS

PART B

Unit 1

Introduction to AI

In This Unit

SESSION 1

Introduction to AI

- What is Artificial Intelligence (AI) ?
- Domains of AI
- Artificial Intelligence and Future

1.2 WHAT IS ARTIFICIAL INTELLIGENCE (AI) ?

Artificial Intelligence (AI) refers to application of technology (through machine or application), which carries out a task requiring some degree of intelligence comparable to human intelligence.

Artificial Intelligence

Artificial Intelligence refers to the ability of a machine or application to carry out a task that requires some degree of intelligence comparable to human intelligence.

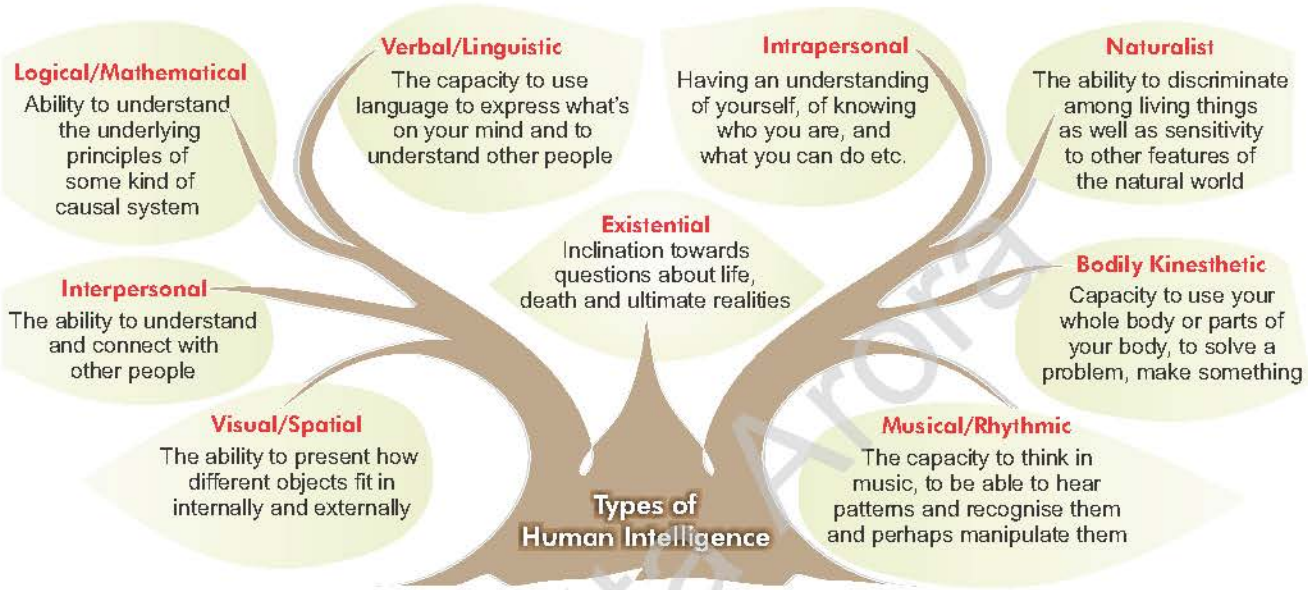


Figure 1.1 Types of Human Intelligence

Activity 1 DREAM SMART HOME IDEA

PART B

You must be knowing what a *smart home* means. Well, it is a home with smart devices using a specialised technology called **IoT (Internet of Things)**. A *smart device* (called a **thing** in this context) is an electronic device that is able to connect, share and interact with its user and other smart devices and IoT is a technology that forms a network of smart devices (**things**).

Let us play a game where you will design a dream smart home for you. Make groups of 4 students and discuss what should be your answer for the below given questions.

Smart Home

A **smart home** is a residence equipped with a number of Internet-enabled interconnected devices (using IoT) that automate tasks normally handled by humans.

(*i*) List the most important things, which you need to connect with these devices.

(*You may refer to* : **https://www.youtube.com/watch?v=jyMC_HAIzpg @0.33seconds**)

(*ii*) Now think of the functionality you would want for your home.

(You may refer to the following videos to check for various technologies and functionalities available today :

https://www.youtube.com/watch?v=9u9kqhHC6Ok

https://www.youtube.com/watch?v=F-mEA5RN_yo)

(*a*)

(*b*)

(*c*)

:

1.3 DOMAINS OF AI

Artificial Intelligence is not a single technology. It is a mix of technologies. Broadly there are *three* domains of AI :

(*i*) Data (Data Science) (*ii*) Computer Vision

(*iii*) NLP (Natural Language Processing)

(*i*) Data / Data Science

This domain of AI may deal with data (structured and unstructured data) and specific methods to use it in AI applications. Data science is a field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data.

So, if your AI powered refrigerator knows in advance that what types of food you would prefer in summers, it is because it has all your past data and it applied data science methods to determine your choice this summer.

Data Science

Data Science is a field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data to apply in AI applications.

(*ii*) Computer Vision (CV)

Computer vision is a field of artificial intelligence that trains computers to interpret and understand the visual world. Using digital images from cameras and videos and deep learning models, computers and machines can accurately identify and classify objects — and then react to what they "see".

So, if your AI powered lock with camera can identify people and then opens door only for authentic members, it is all because of computer vision.

Computer Vision

Computer Vision refers to the training of computers to have a vision somewhat like humans so that machines can accurately identify and classify objects and produce specific 'reactions'.

(*iii*) NLP (Natural Language Processing)

Natural Language Processing is a subfield of linguistics and AI concerned with the interactions between computers and human languages, in particular how to program computers to process and analyse large amounts of natural language data.

So, if you buy a robot to assist you in work and it is capable of understanding your language and speech, and can carry out work and also talk to you – it is all thanks to NLP.

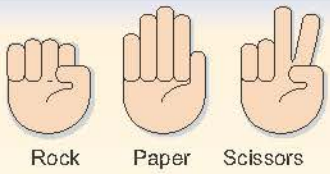
NLP

Natural Language Processing (NLP) is a subfield of AI, mainly concerned with the interactions between computers and human languages in the form of speech (natural language) input and speech (natural language) output.

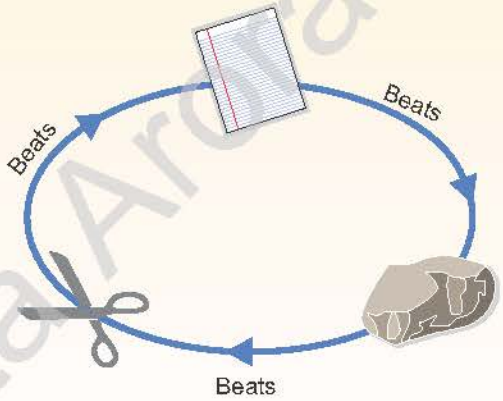
GAME 1

Rock, Paper and Scissors (based on Data)

The game of ***Rock, Paper, Scissors*** is very simple. Each player picks one of the three objects (usually by making the appropriate hand shape on a count of three!) and these rules are applied to see who has won that round :



- Paper wraps (beats) Rock
- Scissors cut (beat) Paper
- Rock blunts (beats) Scissors



The challenge of the game is to guess what your opponent will choose and pick the appropriate object to beat them.

GAME 2

Mystery Animal (based on NLP – Natural Language Processing)

Mystery Animal is a classic 20-questions game based on natural language processing abilities of a computer (AI based). In this game, the computer pretends to be an animal, and you have to guess what it is using your voice. Ask any yes-or-no question you want, like *"Do you have feathers ?"* or *"Do you sleep at night ?"* or *"Do you crawl ?"*, *"Do you dance ?"* and so forth.

This game is NLP based game where it takes, decodes and under- stands your voice input, given in the form of questions (**Natural- Language Understanding**) and gives a response through voice output (**Natural-Language Generation**).

You can play this game on a **Google Home** application by saying *"Hey Google, talk to Mystery Animal"*, or try it on a site with **URL** :
https://mysteryanimal.withgoogle.com/

GAME 3

Emoji Scavenger Hunt (based on Computer Vision (CV))

This game, the ***Emoji Scavenger Hunt*** is an AI's Computer Vision domain-based game. In this game, you have to use your phone's camera and point to things/objects it asks you to find an object that matches the emoji before time runs out! The computer calls the object to be found as emoji and hence the name ***Emoji Scavenger Hunt***.

Using Computer Vision based Artificial Intelligence (i.e., interpreting the imagery input using AI techniques), the computer will guess what is on the screen (as pointed out by you through your camera)!

You can play the game by downloading it from the *Play Store* on even on its site with **URL:https://emojiscavengerhunt.withgoogle.com/** Make sure to give permission to use your phone's camera.

LET US REVISE

- ***Artificial Intelligence*** *refers to ability of a machine or application to carry out a task that requires some degree of intelligence comparable to human intelligence.*
- *The term "Artificial Intelligence" was coined by an American Computer Scientist* ***John McCarthy*** *in 1955.*
- *AI has three domains :* ***Data and Data Science, Computer Vision*** *and* ***Natural Language Processing.***
- ***Data science*** *is a field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data to apply in AI applications.*
- ***Computer vision*** *refers to the training of computers to have a vision somewhat like humans so that machines can accurately identify and classify objects – and then react to what they "see".*
- ***Natural Language Processing (NLP)*** *is a subfield of AI, mainly concerned with the interactions between computers and human languages in the form of speech (natural language) input and speech (natural language) output.*
- *FaceBook's* ***DeepFace*** *technology is based on Computer Vision AI.*
- *Modern day chatbots and virtual assistants are based on NLP.*

SESSION 2

AI in Daily Lives

- Smart Cities
- Smart Schools
- Smart Home

2.2 SMART CITIES

A **smart city** is a city that uses technology with AI to provide services ; anticipate and solve city problems and enhance the lives of citizens. A smart city offers :

- improved transportation and accessibility,
- improved social services and inclusion,
- sustainability,
- interactive control and voice to its citizens.

Smart City

A **smart city** is a city that plans, designs and manages its needs and services using technology and AI in a smart way.

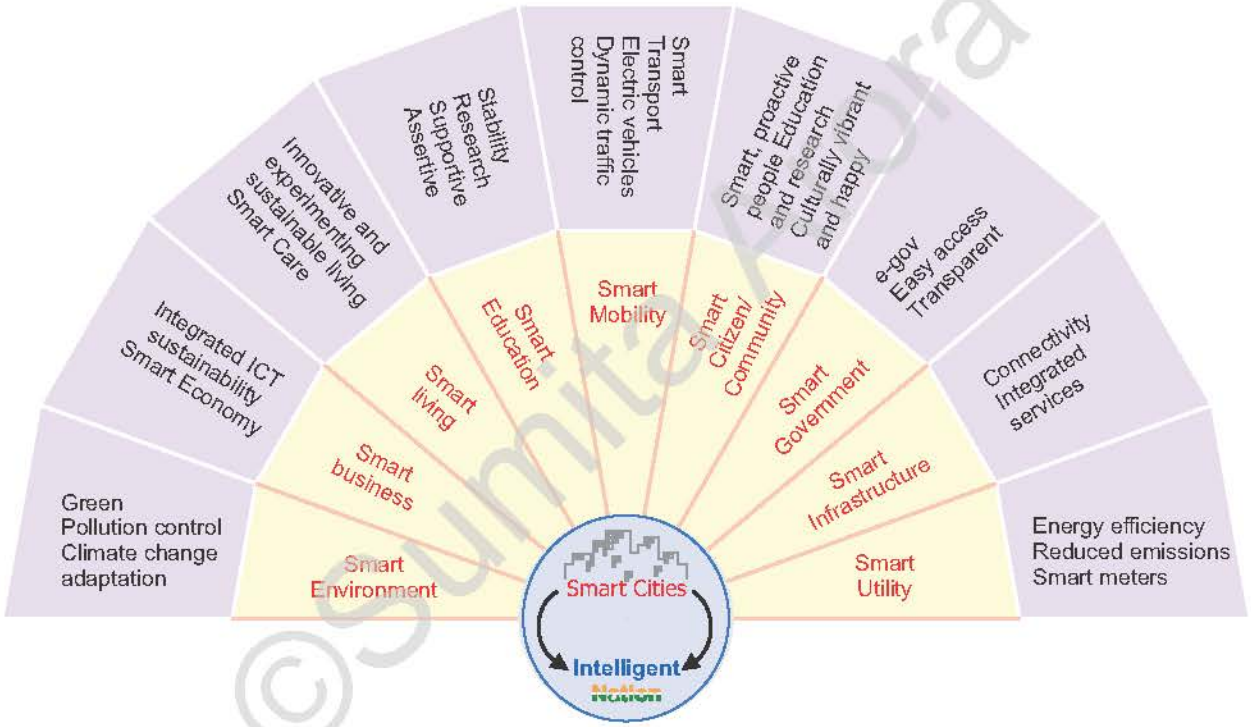
2.2.1 Smart Cities' Functioning

- Connected traffic lights receive data from sensors and cars adjusting light cadence and timing to respond to real-time traffic, reducing road congestion.
- Connected cars communicate with parking meters and direct drivers to the nearest available spot.
- Electric vehicles (EV) get charging docks at their parking spots — *smart charging* (charging while parking).
- Over speeding cars automatically get challans and automatically fine is debited from their linked accounts.
- Smart garbage cans automatically send data to waste management companies and schedule pick-up as needed versus on a pre-planned schedule.
- Citizens' smartphone acts as the legit document holder with information such as their mobile driver's license and ID card with digital credentials.
- Smartphones act as one point access system for the city and local government services.
- Smartphones act as smart wallets linked to the bank accounts and capable of making payments on the fly.

2.2.2 Technologies for Smart Cities

A smart city is possible only because of the technologies and AI, which are the real systems behind its functioning. A smart city uses the following technologies :

- ***IoT Enabled Devices.*** The **Internet of Things (IoT)** is a special type of network of physical objects, called "**things**", which are embedded with smart sensors and software (and other technologies). The **things** (IoT devices) connect and exchange data with other devices and systems over the Internet. These devices range from ordinary household objects to sophisticated industrial tools.
- ***Video Cameras.*** Innovative surveillance cameras with integrated IoT technology and software capture and collect data.
- ***Social Media.*** Social media posts, location etc. give an idea about how people are using places (and hence can help city planners).
- ***Artificial Intelligence.*** Various AI based technologies and algorithms help analyse collected data and provide efficient services.



2.3 SMART SCHOOLS

A smart school is a school, which uses modern technology along with AI to manage all functions of a school. A school has broadly following functional components :

- Imparting Education
  - Teaching and Learning
  - Examination and Evaluation
  - Result and Report card generation
- Administration
  - Admission
  - Timetable and Attendance
  - Fee collection
  - Provide required service
  - Employee payroll
- Transportation
  - School bus and other transport management

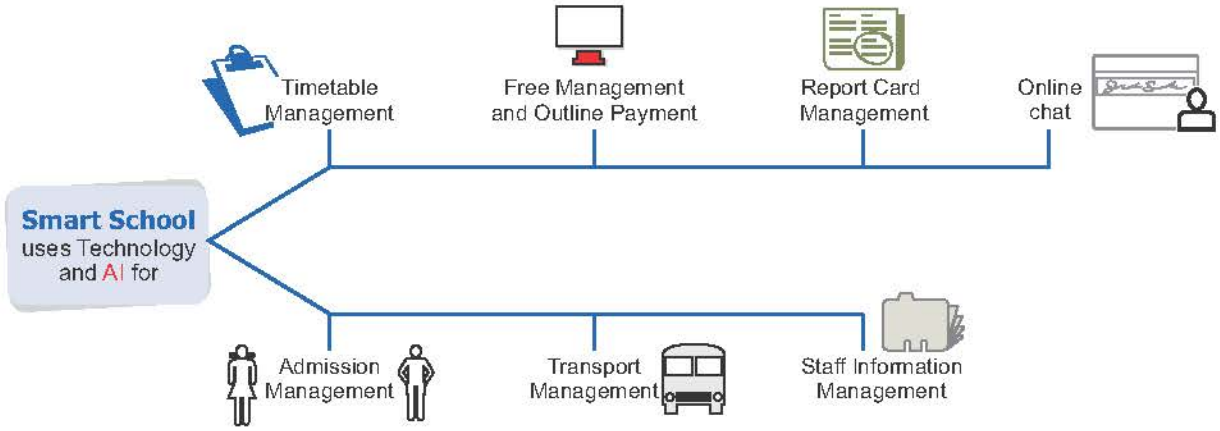
A school broadly has the following stakeholders :

- Teachers
- Administrative staff
- Parents
- Students

Smart School

A **smart school** is a school, which uses modern technology along with AI to manage all functions of a school.

A smart school uses technology with data and AI to automate each of the functional components and make it easy-to-use and much more useful for each of the stakeholder.



2.3.1 Functioning of Smart Schools (Features and Technologies)

- Its classrooms are fitted with **multimedia courseware and presentation** facilities.
- It has arrangements and media centres for **multimedia and AR/VR based teaching and learning resources**.
- All its rooms, labs, offices and staff rooms are continuously ***connected through network and the Internet***.
- It has multimedia development centre for **helping teachers create multimedia/AR-VR based teaching materials**.
- It has proper setup for **centralized teaching and video-conferencing**.
- It has **multiple libraries** and **access points** that provide online access to courseware catalogues and databases, information and resource management systems.
- Its administration offices have **high performance computers** connected with networks and databases for managing students' data and facilities for tracking student and teacher performance or resources.
- It has intelligent AI based software installed which keeps an eye on teachers' and students' posts and usage pattern to identify problematic or attention seeking areas beforehand.
- Its transport system is fully connected with school network and with parents through **GPS and provides real-time monitoring** of school buses and taxis.
- It has **specialised server room(s)** to handle applications, management databases, and web servers; provide security.

2.4 SMART HOME

Smart home is a network of hardware, communication, electronic interfaces coupled with AI based algorithms that work to integrate everyday devices with one another via the Internet. A smart home gives its users a **'remote'** control over devices at home (even if the user is away from home) and enhanced security features. This allows you to turn on the lights, lock the front door, or even turn down the heat, no matter where you are.

Smart Home

Smart home is a network of hardware, communication, electronic interfaces coupled with AI based algorithms that work to integrate everyday devices with one another via the Internet and gives its users a **'remote'** control over devices at home and enhanced security features.

2.4.1 Components of a Smart Home

There are *three* main elements of a smart home that work with the help of IoT and the Internet, which are :

(*i*) **Sensors.** A sensor is a device that is capable of identifying gestures or monitor changes and convert them in electrical form. Sensors can monitor changes in daylight, temperature, or motion detection.

(*ii*) **Controllers.** Controllers refer to the smart devices of today – personal computers, tablets or smartphones – used to send and receive messages about the status of automated features in your home.

(*iii*) **Actuators.** Actuators may be light switches, motors, or motorized valves that control the actual mechanism, or function, of a home automation system. They are programmed to be activated by a remote command from a controller.

2.4.2 Features of Smart Homes

Smart homes offer a variety of services and functions. Some of the these include :

- Fire and carbon monoxide monitoring
- Smart lighting
- Remote lighting control
- Thermostat control
- Appliance control
- Remote Locking
- Home automation security systems and cameras
- Live video surveillance
- Alarm systems
- Real-time text and email alerts
- Digital personal assistant integration
- Keyless entry
- Voice-activated control

Activity INTERACTIVE STORY

Let us create an interactive story practically, based on the steps listed in above infobox.

My Smart Home Interactive Story

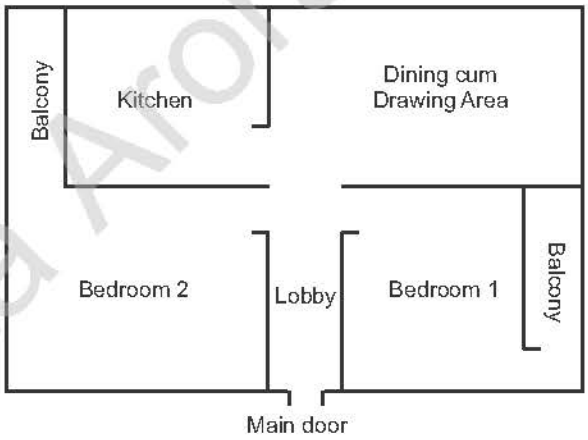
We are giving below an interactive story created via Google Docs Presentation tool or through Microsoft PowerPoint. The text on the slides is shown normally. The links/action buttons in bold and the instructions for story creator are shown in colour. The text of an instruction will not appear on the slide, rather the story-creator should act as per the instruction and take necessary action(s).

Slide 0 Title of the story :

My Smart Home

Note

It is a 2BHK house with 2 bedrooms, kitchen, a dining hall, a drawing room and a small lobby.



Slide 1 **The House Floor Plan**

Illustration shown on the right

Slide 2 **Enter My House**

My house is situated in a calm area of our city and has a beautiful main door.

Main Door. *(link to slide 3)*

Slide 3 **Verifying Entry**

- The main door has a Smart Lock that can take a passcode (through keyboard or through voice) or can verify you if you have a valid card.

(Image of a **Smart Lock** and a robotic-sound file for, *'Please tell or enter your passcode or show your card'*)

Valid Passcode/card *(link to slide 4)*

Invalid Passcode/card *(link to slide 12)*

Slide 4 **Welcome to My Home (Lobby Area)**

Through main door you enter into an open lobby.

The house has one bedroom on the right of lobby, kitchen on the left of dining area, dining cum drawing hall in front and second bedroom on the left of the lobby.

Four arrows, each linking to an area of the house with smart features of the room.

Link (To Dining cum Drawing Hall) /Slide 7

Link (To Bedroom 2) / Slide 9 ⇦ ⇨ Link (To Bedroom 1 / Slide 5)

⇧ ⇩

Link (Exit through main door) /Slide 11

Slide 5 **Bedroom 1**

- The lights switch on automatically when you enter the room. Smart blinds open if it is daytime and the natural light is too low.
- The room also has smart heating and cooling depending upon the season.

Link (Back to open lobby) / Slide 4 ⇦ ⇨ Link (To Balcony) / Slide 6

Slide 6 **Balcony of Bedroom 1**

The balcony of bedroom 1 has seating of 3-4 people with planters on the wall and fairy lights around them.

- The Smart Speakers of balcony automatically start playing lovely soothing melodies whenever someone goes to sit there.

Link (Back to bedroom1) / Slide 5 ⇦

Slide 7 **Dining cum Drawing Hall**

- The Dining cum Drawing Area has Smart Lights and Smart blinds. It also has Smart Air Monitor which keeps monitoring the air purity.

Link (Kitchen) / Slide 8 ⇦

⇩

Link (Lobby) / Slide 4

Slide 8 **Kitchen**

- The kitchen of my smart home is a smart kitchen with ***Smart Cooking Range, Smart Refrigerator, Smart Microwave, Smart Oven*** etc., which all can respond to voice command and take intelligent decisions.

⇨ Link (Back to Dining Area) / Slide 7

Slide 9 **Bedroom 2**

- The **lights switch on automatically** when you enter the room. **Smart blinds** open if it is daytime and the natural light is too low.
- The room also has **smart heating** and **cooling** depending upon the season.

Link (Back to balcony) / Slide 10

⇧

⇨ Link (To Lobby) / Slide 4

Slide 10 **Balcony of Bedroom 2**

The balcony of bedroom 2 has seating of 3-4 people with plants kept here and fairy lights around them.

- The Smart Speakers of balcony automatically start playing lovely soothing melodies whenever someone goes to sit there.

Link (Back to bedroom 2) / Slide 10

Slide 11 Thank you for visiting my smart home. Hope you enjoyed the tour.

Slide 12 Your passcode or card is invalid. Please leave fast else the burglar alarm will go off in 7 seconds and the police will be notified.

Note

You may add more smart home devices and gadgets to this interactive story. You refer to many sites such as :
http://www.home-designing.com/2016/03/50-insanely-useful-smart-home-products-you-can-buy-right-now.

This section/content is not part of teacher-support PDF

LET US REVISE

- *A smart city is a city that plans, designs and manages its needs and services using technology and AI in a smart way.*
- *A smart city uses technologies like **IoT enabled Devices**, **Video Cameras**, **Social Media**, **Artificial Intelligence** etc.*
- *The **Internet of Things (IoT)** is a special type of network of physical objects, called "things" (embedded with smart sensors and software) to connect and exchange data with other devices and systems over the Internet.*
- *A **smart school** is a school, which uses modern technology along with AI to manage all functions of a school.*
- *A smart school uses technologies like high performance computers, multimedia courseware and presentation, multimedia and AR/VR based teaching and learning resources, interconnected and connected to the Internet, centralized teaching and video-conferencing, multiple libraries and access points, GPS and provides real-time monitoring, GPS and provides real-time monitoring, specialised server rooms etc.*
- *A smart school uses technologies for various things such as for teaching and learning, for examination and evaluation, for result and report cards, administration, transportation.*
- *Smart home is a network of hardware, communication, electronic interfaces coupled with AI based algorithms that work to integrate everyday devices with one another via the Internet and gives its users a 'remote' control over devices at home and enhanced security features.*
- *A smart home works through these components along with IoT and the Internet : **sensors** (devices that identify gestors or motion), **controllers** (smart devices like smartphones, tablets and laptops etc.) and **actuators** (light switches, motors, or motorized valves that control the actual mechanism).*

SESSION 3

AI and UN Sustainable Development Goals

- What is Sustainable Development ?
- UN Sustainable Development Goals
- Impact of AI on Sustainable Development Goals

3.2 WHAT IS SUSTAINABLE DEVELOPMENT ?

Sustainable development is the *"Development that meets the needs of the present without compromising the ability of future generations to meet their own needs."*

Sustainable Development

Sustainable development is the development that meets the needs of the present without compromising the ability of future generations to meet their own needs.

3.3 UN SUSTAINABLE DEVELOPMENT GOALS (SDGs)

In September 2015, the General Assembly adopted the 2030 Agenda for Sustainable Development that includes *17 Sustainable Development Goals (SDGs) with 169 targets to be achieved*. Building on the principle of "**leaving no one behind**", the new Agenda emphasizes a holistic approach to achieving sustainable development for all.

Note

UN agenda 2030 has 17 SDGs with 169 targets to be achieved.

The 17 SDGs included in agenda 2030 are :

GOAL 1	No Poverty	GOAL 10	Reduced Inequality
GOAL 2	Zero Hunger	GOAL 11	Sustainable Cities and Communities
GOAL 3	Good Health and Well-being	GOAL 12	Responsible Consumption and Production
GOAL 4	Quality Education	GOAL 13	Climate Action
GOAL 5	Gender Equality	GOAL 14	Life Below Water
GOAL 6	Clean Water and Sanitation	GOAL 15	Life on Land
GOAL 7	Affordable and Clean Energy	GOAL 16	Peace and Justice Strong Institutions
GOAL 8	Decent Work and Economic Growth	GOAL 17	Partnerships to achieve the Goal
GOAL 9	Industry, Innovation and Infrastructure		

SDG 1 No poverty AI can help with this important goal through techniques like *satellite mapping* and *data analysis of poverty* and allocating required resources in real-time based on the outcome.

SDG 2 Zero hunger AI-based predictive analysis from imaging with automated drones and satellites can increase Agriculture productivity and reduce livestock production losses.

SDG 3 Good health and well-being AI-based preventative healthcare programs and diagnostics significantly improve, leading to new scientific breakthroughs.

SDG 4 Quality education AI can impact this important goal through techniques and methods like Virtual, Augmented resources, intelligent mentoring of students and responsive personalized learning.

Using *Big data analysis,* the graduation rates of students from low-income groups and first-generation learners can be deeply studied and warning signs can be spotted in time and actions can be taken accordingly.

SDG 5 Gender equality AI tools and technologies along with data analysis help identify gender bias, and thus allows for timely invention and proper action. AI tools of today allow for automated or augmented tasks, thus empowering women for making better contribution in society.

SDG 6 Clean water and sanitation Modern age AI tools along with the Internet-of-Things (IoT) can timely analyse the sanitation and consumption patterns. This analysis can then lead to better sanitisation services and safe water consumption.

SDG 7 Affordable and clean energy This goal is about providing Green energy in all forms and is greatly helped through AI based real-time analysis for the same.

SDG 8 Decent work and economic growth AI tools and AI augmentation along with intelligent devices can help the workers and thereby result in increased productivity and satisfaction. All this will act significantly for improving work environment and economic-growth.

SDG 9 Industry innovation and infrastructure Modern day hybrid manufacturing technology incorporates technologies like AI, IoT sensors, 4D printing etc., which is leading to infrastructure innovation of industries and thereby yielding exponential innovation unprecedented in world history.

SDG 10 Reduced inequalities AI is significantly contributing towards this goal majorly through Human augmentation using AI-inspired devices, which enhance physical capabilities of people, and corrects disabilities, which might not have been possible a few years ago. This provides super senses and knowledge and leads to equal and inclusive society.

SDG 11 Sustainable cities and communities The AI plays a very significant role in modern age smart cities. Combined with techniques like big data analysis and data sciences, it contributes towards creating sustainable cities and communities.

SDG 12 Responsible consumption and production AI technologies and techniques are successfully used for identifying unnecessary wastage, improved and optimized utilization of resources (resource efficiency). Such intelligent use of AI has resulted in yielding optimal consumption and production levels.

SDG 13 Climate action Climate and Climate change data are enormous. AI helps in analysing this big data and aids in climate modelling. This way, AI timely predicts climate-related problems and disasters and helps counter them.

SDG 14 Life below the water Marine life is crucial for the world. AI techniques such as Pattern recognition can monitor factors like oceanic life migration, increasing/decreasing population levels, and fishing activities and habits. This can help in creating sustainable marine ecosystems and combat illegal fishing.

SDG 15 Life on land AI techniques such as game theory and pattern recognition and other modern applications can help track land-animal migration, their increasing/decreasing population levels, etc. These can also monitor hunting behaviour. This can help in creating sustainable land ecosystems and combat illegal poaching.

SDG 16 Peace, justice, and strong institutions AI tools and techniques can study past available crime and peace data. Careful study of this analysis can help reduce discrimination and corruption. The outcome of this analysis can timely predict and prevent global cyber threats and lead to personalized and responsive intelligent services, thereby creating robust institutions.

SDG 17 Partnerships for goals Agencies like ITU, UN Agencies, and XPRIZE Foundation are working with governments, academia, and industries worldwide, intending to develop human-centric AI to solve humanity's challenges and meet all SDGs as proposed by UN.

LET US REVISE

- ***Sustainable development*** *is the development that meets the needs of the present without compromising the ability of future generations to meet their own needs.*
- *Sustainability is the goal of sustainable development.*
- *UN has proposed 17 sustainable development goals (SDGs) for agenda of 2030 with 169 targets.*
- *AI is playing a significant role in achieving the UN SDGs.*

SESSION 4

Skills Required for the Jobs of the Future

- Present AI Trends
- Some Future Trends of AI
- AI Jobs and Skills of the Future

4.2 PRESENT AI TRENDS

Past decade was of AI's applications in various fields. Slowly AI is making inroads to all spheres of life. Let us talk about some current trends of AI usage which will also be used in future.

1. AI Customer Support and Assistance

Try chatting with customer service of big companies like Amazon, Jio etc. and you will find that actually. You are interacting with a **bot** or software or a **robot** initially.

2. Decision-Making through AI Powered Data Analysis

The **Data Science** branch of AI can generate, curate and analyse huge amounts of data and even make AI-powered automated decisions.

3. Predictive Analytics

Artificial Intelligence has successfully predicted the anticipated changes in various fields.

4. Enhanced Customization

The power of AI is being used to determine and predict customers' changing preferences and expectations and thus accordingly customisation and personalized services can be provided.

5. AI-Powered Chatbots

These days chatbot programs perform basic conversation with clients.

4.4.1 Effective Communication and Collaborative Work Skills

Other than the technical skills, the AI professionals of tomorrow must have effective communication and collaborative skills listed below :

(*i*) Critical Thinking. AI professionals should be highly critical thinkers. They should take nothing at face value and be naturally curious. They should believe in trial and error and must test and experiment before making a concrete decision.

(*ii*) Analytical Thinking and Innovation. Analytical thinking is the ability to tackle complicated issues by evaluating information available. AI professionals are analytical thinkers who can detect patterns between datasets that often lead to creative solutions and innovation.

(*iii*) Complex Problem-Solving. Problem-solving can be defined as searching for steps that will lead to a successful outcome. The complex problem solving is essential for AI Problems. It includes the problem itself and a set of many highly interrelated elements to move it.

Complex Problem-Solving skill involves : (*a*) Define a problem, its goals and challenges (*b*) Identify various solutions and possible outcomes (*c*) Choose a plan on the basis of accuracy of solution, ease of implementation and cost (*d*) Implement the idea.

(*iv*) Curiosity and Communication. AI professionals are curious and observant - how things work; how things don't work; what change brings efficiency and what not. All this they notice and communicate it to the right person — their team, leader or head (internally) and by reflecting in the products (externally).

(*v*) Team Player. AI professionals are team players and believe in collaborative efforts. AI itself is a collaborative result of multiple technologies and theories; AI professionals reflect this in their nature and collaborate with other people and different technologies.

LET US REVISE

- *AI has found its applications in nearly all the fields around us.*
- *AI is currently being used in areas like customer support, data analysis, predictive analysis, enhanced customisation, AI-powered chatbots etc.*
- *Future trends of AI will have : AI powered robots, advanced NLP, human like text generation, AIoT (AI & IoT), Generative Adversarial Networks (GANs) etc.*
- *For AI related jobs other than AI based technical skills, skill like critical thinking, complex problem solving, analytical thinking and innovation etc.*

SESSION 5

AI Ethics

- Ethical Issues Around AI
- AI Bias and AI Access

5.2 ETHICAL ISSUES AROUND AI

The dictionary defines ethics as, *"the moral principles that govern a person or a group's behaviour or actions"* **Or** *"the moral correctness of a conduct or action"*. In short, **Ethics** are the moral responsibility of anyone or anything that can impact others.

This section/content is not part of teacher-support PDF

Figure 5.1 lists various ethical issues surrounding AI.

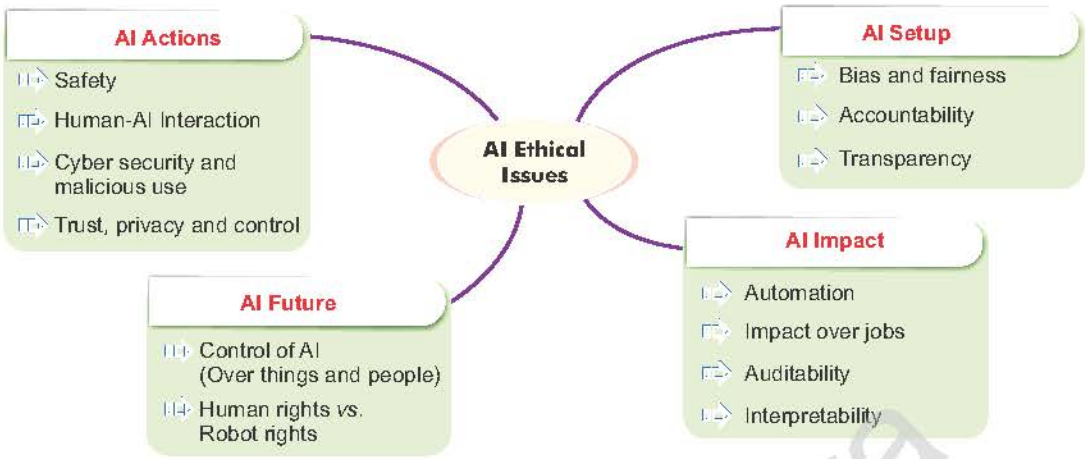


Figure 5.1 Ethical Issues of AI

5.2.1 Examples of AI Ethical Issues

Let us discuss some examples of AI ethical issues.

1. Bias and Fairness

Ethically an AI system should be free from all types of biases and be fair, *e.g.*, an AI system designed for picking candidates for a job must not be biased against any gender, race, colour or sexuality and so forth. It should be free from all such things and be totally fair.

2. Accountability

AI learns and evolves over time and data. What if an evolved algorithm makes some big mistake? Who would be accountable for it? For instance, when an autonomous Tesla car hit a random pedestrian during a test, Tesla was blamed and not the human test driver sitting inside, and certainly not the algorithm itself. But what if the program was created by dozens of different people and was also modified with each incident and more data available? Can the developers or the testers be blamed then?

3. Transparency

Transparency means nothing is hidden and everything that AI performs is explainable. Transparency ensures that there is full information and knowledge about these :

- data used, its range, interval and sources etc.
- models used are appropriate for the context make sense,
- models are thoroughly tested
- why particular decisions are made

4. Safety

AI technology, tools and practices should be so implemented such that they cause no direct or indirect harm to *data, people* and the *outcomes*. AI practices must be safe to ensure the well being of individual persons and the public welfare. AI practices must uphold public trust through the responsible use of technologies.

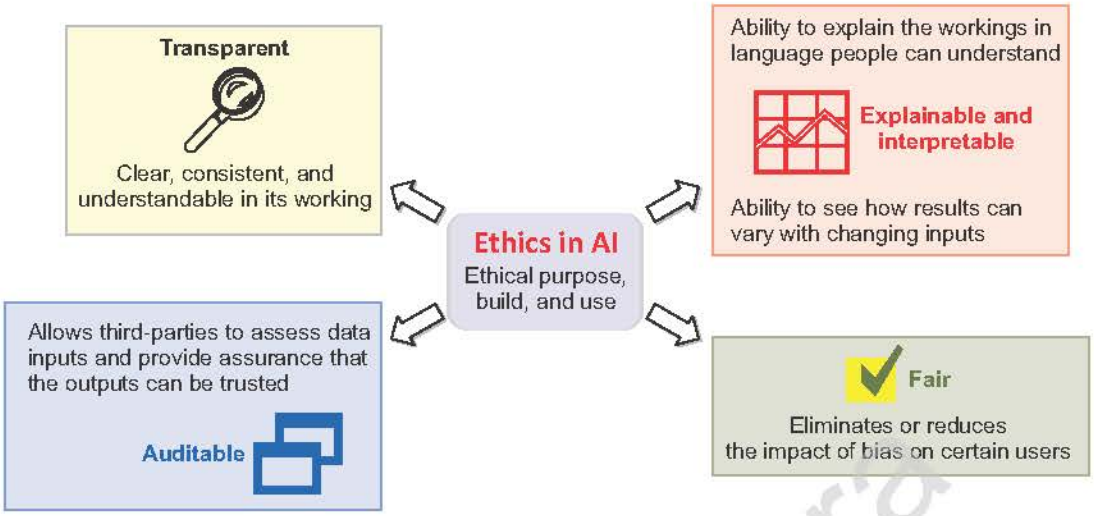


Figure 5.2

5. Human AI Interaction

AI must not deceive humans or other living beings, and it must not threaten or violate human dignity in any way.

PART B

6. Trust, Privacy and Control

Improved AI "faking" technologies make what once was reliable evidence — into unreliable evidence – this has already happened to *digital photos*, *sound recordings*, and *video*.

Thus, it is the ethical responsi- bility of the creator and user of AI to ensure that these are not misused.

7. Cyber Security and Malicious use

It is the ethical responsibility of an organisation to have human control over AI usage in terms of its span and control so that it is not available to hackers for malicious use.

This section/content is not part of teacher-support PDF

8. Automation and Impact over Jobs

AI does not mean that jobs are reduced, it just means that the nature of jobs and work is predominantly changing.

It is the ethical responsibility of an organisation to upgrade the skillset of its workers so that they upgrade their skillset and be ready for futuristic AI oriented jobs. It is ethical responsibility of governments too (equally and even more) to bring appropriate changes.

9. Human Rights in the Age of AI

AI has generated new form of threats

- huge risk to data privacy and protection — *violates human right to privacy.*
- biased decision and *violates human right to fair chance and justice.*

This section/content is not part of teacher-support PDF

- As AI can process humongous sets of data, it can analyse huge symptoms dataset of a person and can predict may possible future ailments and disease. Using this analysis, the health insurance companies may deny insurance to some people and thus would *violate human right to affordable healthcare.*

AI Ethics

AI ethics is a set of values, principles, and techniques that employ widely accepted standards of right and wrong to guide moral conduct in the development and use of AI technologies.

AI Bias

AI bias is an anomaly (irregularity of abnormality) in the result produced through AI based programs and algorithms because of prejudiced (discriminatory) assumptions made during the algorithm development process or prejudices in the training data.[1]

Training Data is used to store people characteristics in the form of feature values. Bias in a data collection happens if some data representing a *feature*, *group*, *ethnicity* etc. is under-represented or over-represented. For instance, consider this — a minority group's preferred colour for car is *'Red'* because of their cultural influence. Now if another dataset stores that 'Red' is also a preferred choice of colour for aggressive drivers, then without much representation, this dataset may link the minority group with aggressive driving — an AI bias, here.

Training Data

Training data is a huge collection of labelled information that's used to build an AI model (*e.g.*, machine learning model).

Reasons for AI Bias in Data

Other than the over- and under-representation, there are many more reasons that cause or contribute to AI bias.

These are :

(*i*) Human bias in decisions

(*ii*) Flawed and unbalanced data collection

(*iii*) Under- or over-representation of specific features

(*iv*) Wrong assumptions

(*v*) No proper bias testing

(*vi*) No bias mitigation (*i.e.*, reducing the severity of bias)

Bias in Data Collection

Bias in data collection refers to flawed or unbalanced data with over- or under-representation of data related to specific features or groups or ethnicity etc. in the final data-collection.

Ensuring Data Fairness

It is very important for the collectors to do the following for the fairness of data and thereby, fair decision-making

- Identifying the correlation of features with data (should be diverse)
- Identifying the correlation among sets of data, while studying the impact of data and minimising the impact so as to have fair decisions
- Observing biases in human decisions and the data collected
- Ensuring "balanced" data
- Supervised decision-making
- Regular Bias testing by learning about biases and inducing fairness by thorough and repeated testing of data and modification in training data

Reducing and Mitigating AI Bias

Let us now learn how the AI people can reduce the AI biases in data collections and decisions :

(*i*) **Thorough Research.** The data collector must research their users or subjects in advance about which the data are being collected. They should be aware of general results and odd results of the data.

(*ii*) ***Diversity of Team.*** The team working for data collection or algorithm development must be diverse so that one person or team does not have major influence on data and algorithm of decision-making.

(*iii*) ***Data Diversity.*** Combine inputs from multiple sources to ensure data diversity.

(*iv*) ***Standardised Data Labelling.*** The team must have standardised way of labelling so that accurate, consistent and standardised data labels are used in data collection.

(*v*) ***Identify Bias-proneness.*** The team should identify the possible occurrences of biases among data sets and use multi-pass annotations, *i.e.,* multiple set of annotators label the data so as to minimise the possible bias.

(*vi*) ***Data Review.*** Enlist the help of someone with domain expertise to review collected and/or annotated data. Someone from outside of the team may see biases that the team has overlooked.

(*vii*) ***Regular Data Analysis.*** The team should keep track of errors and problem areas so as to respond to and resolve them quickly.

(*viii*) ***Regular Bias Testing.*** The team must test the collected data, training data and the overall performance of the algorithm against biases and use approaches and tools to mitigate the biases.

PART B

LET US REVISE

- *Various ethical issues of AI are : Bias and Fairness, Accountability, Transparency, Safety, Human – AI interaction, Trust, Privacy and Control, Malicious use of AI, Impact over jobs, Human and Robot rights and so forth.*
- *Deepfake is a technology that can generate fake digital photos, sound recordings, and video, which look just as original as possible.*
- *AI ethics is a set of values, principles, and techniques that employ widely accepted standards of right and wrong to guide moral conduct in the development and use of AI technologies.*
- *AI bias is an anomaly (irregularity of abnormality) in the result produced through AI based programs and algorithms because of prejudiced (discriminatory) assumptions made during the algorithm development process or prejudices in the training data.*
- *Training data is a huge collection of labelled information that's used to build an AI model (e.g., machine learning model).*
- *Some possible reasons of AI bias are : Human bias in decisions; flawed and unbalanced data collection; Under- or over-representation of specific features; Wrong assumptions; No proper bias testing; No bias mitigation (i.e., reducing the severity of bias).*
- *Bias in data collection refers to flawed or unbalanced data with over- or under-representation of data related to specific features or groups or ethnicity etc., in the final data-collection.*
- *A given AI model is fair if the outputs are independent of sensitive parameters (e.g., gender, race, sexuality, religious faith, disability, etc.) for a specific task that is already affected by social discrimination.*

UNIT 2

Problem Scoping

IN THIS UNIT

SESSION 1

Introduction to AI Project Cycle

Stages of AI Project Cycle

AI Project Cycle

An AI project undergoes some phases from its initiation to closure, which together are known as **AI Project Cycle**.

1.2 STAGES OF AI PROJECT CYCLE

An artificial intelligence based project undergoes broadly five (below given) stages. In other words, the stages of an *AI Project Cycle* are :

(*i*) Problem Scoping (*ii*) Data Acquisition

(*iii*) Data Exploration (*iv*) Modelling

(*v*) Evaluation

(*i*) Problem Scoping

In this stage, broadly the aim and scope of the project undertaken are decided. In other words, during this stage, following things are decided :

- strategic business objectives
- expected outcomes of the project
- stakeholders' expectations
- key resources and steps
- success metrics

Note

In the ***problem scoping stage***, broadly the *aim* and *scope* of the project undertaken are decided.

(*ii*) Data Acquisition

This stage is a crucial stage of AI project cycle. In this stage, the data are acquired keeping in mind the scope and parameters of the previous stage.

Standardised forms of data are extracted from dissimilar sources of data (bulk of sources) and produced in the most appropriate input form, most suitable form to produce the desired form of outcome, *e.g.*, for an AI based insurance project, the data will be heavily numbers and figures, while for a social-media based AI project or for providing security to a bank, the data may be mostly in visual form. However, to interpret the patterns and trends in the data, data in visual forms such as *charts*, *graphs*, *maps*, *histograms* and so forth is preferred.

Thus, the data acquired during this stage must be :

- as per the scope and parameters of the project
- accurate
- reliable

In the ***data acquisition stage***, the data are acquired keeping in mind the scope and parameters of the previous stage.

Note

To interpret the patterns and trends in the data, which is in large quantities, data in visual forms such as ***charts***, ***graphs***, ***maps*** (*e.g.*, ***heat maps***, ***density maps*** etc.), ***Tree diagrams***, ***Venn diagrams***, ***histograms*** and so forth is preferred.

(iii) Data Exploration

The purpose of the third stage is to explore data and its patterns so as to :

- choose the types of models for the project that can solve the problem.
- clean and normalize the data to standardise and correlate the data.

The *'cleaning and normalizing dirty data'* requires data scientists to make decisions on data they may not understand, like what to do with *missing data*, *incomplete data*, and *deviating data* (called *outliers*).

Thus, during this phase after deciding about the type of models that may serve :

- data from multiple sources is aggregated into a format suitable for the AI project's model.
- the form (*e.g.*, database or visual forms of data or metadata[1] etc.) and structure of data is such chosen so that it is likely to produce the desired outcome.

In the ***data exploration stage***, the data are explored to choose the possible models and clean & normalise the data.

(iv) Modelling

The next phase of an AI project cycle is to model the data that will be used for the prediction. So, the following activities take place in this phase :

- The selected models (from the previous stage) are tested and analysed and the most suitable AI-model is chosen that matches the requirements of the project.
- Once the most efficient model is chosen, AI algorithm(s) is developed around it.

1. *Metadata* means information/details about the data such as its *type*, *format* and other details etc.

- During this stage only, the ***training data*** and ***testing data*** are also decided.
  - A **training data (model)** is a dataset that is used to train an AI algorithm. The ***testing data*** is used with the **AI algorithm** to correlate the input data with the *processed output*. The result from this correlation is used to modify the model so that the processed output becomes more and more matching to the *sample output*.
  - The **testing dataset** is a set of observations used to validate the developed models after training is complete using some performance metric.

In the ***modelling stage***, the selected models are tested and analysed and the most suitable AI-model is chosen and AI-algorithms are developed around it.

(v) Evaluation

In this final phase, now the developed model is actually evaluated for accuracy and performance using new data. Then the results are evaluated to determine if the model can be deployed or requires some improvement prior to the final deployment.

Before a model can be deployed,

- it must meet some minimum accuracy standard required for deployment.
- there must be a clear agreement on what level of risk is acceptable for inaccuracy.

In the ***evaluation stage***, the developed model is actually evaluated for accuracy and performance using new data so as to determine if the developed model is deployable or not.

LET US REVISE

- *An AI project undergoes some phases from its initiation to closure, which together are known as **AI Project Cycle**.*
- *The five stages of AI project cycle are : **Problem Scoping, Data Acquisition, Data Exploration, Modelling, and Evaluation.***
- *In the **problem scoping stage**, broadly the aim and scope of the project undertaken are decided.*
- *In the **data acquisition stage**, the data are acquired keeping in mind the scope and parameters of the previous stage.*
- *In the **data exploration stage**, the data are explored to choose the possible models and clean & normalise the data.*
- *In the **modelling stage**, the selected models are tested and analysed and the most suitable AI-model is chosen and AI-algorithms are developed around it.*
- *In the **evaluation stage**, the developed model is actually evaluated for accuracy and performance using new data so as to determine if the developed model is deployable or not.*

SESSION 2

Problem Scoping – Set Goals

- Choosing a Theme for Problem Scoping
- Identify Problems Around the Selected Topic

This section/content is not part of teacher-support PDF

2.2 CHOOSING A THEME FOR PROBLEM SCOPING

As your aim is to develop an AI based solution for a problem, look for problems around you that require a novel solution. For this, firstly choose a theme and then choose a problem from the domain of chosen theme. You may choose a theme from one of the following :

- Environment
- Healthcare
- Education and E-education
- Women Safety
- Wellness
- Cyber Security
- Agriculture
- Infrastructure
- Transport

and many more.

2.3.1 Set Goal for Your AI Project

Once you have studied and researched on your chosen topic and listed its problems, it is time to set goal for your AI Project.

To set a goal for your AI project, carefully study the problems listed and pick one or two problems where Artificial Intelligence can make a difference.

This section/content is not part of teacher-support PDF

Let Us Revise

- *For your AI project, first decide about a theme and its subtopics.*
- *Then choose a subtopic and list a set of problems in the chosen subtopic.*
- *From the list of chosen problems. Choose one or two problems as the Goal of AI project.*

SESSION 3

Problem Scoping – Identify Stakeholders, Actions and Ethical Issues

- 4Ws Canvas
- Set Actions Around the Goal
- Ethics Involved in the Goal of the Project

3.2 4Ws CANVAS

Setting goals for a project is not enough. In order to solve a problem, the *problem statement* must be clearly defined. It is very important to know how this problem is *affecting whom*, *what is the cause* and *why should it be solved* ? For this purpose, a useful tool — **4Ws Canvas** is used.

1. 'Who'

'Who' are the ones for which the solution is to be developed. Who are the **stakeholders** ? That is,

- Who have the problem ?
- Who are getting affected by it (directly/indirectly) ?

Know more about the stakeholders.

4Ws Canvas

4Ws Canvas is a framework to define the problem statement articulately (*i.e.*, effective and clear definition) by finding factors like 'Who', 'What', 'Where'/'When' and 'Why'.

2. 'What'

What exactly the problem is :

- What is the nature of the problem ?
- What is the evidence that proves that it is a problem (and must be solved) ?

Stakeholders

The **stakeholders** are the directly or indirectly affected people who are facing the problem and will benefit from the solution.

3. 'Where'/'When'

Where does the problem arise ? Here, the context, situation/location of the problem is studied :

- Figure out the context/location of the problem.
- Observe the problem in its context.
- Describe the context of the problem.

4. 'Why'

After understanding the first 3 canvases of 'WHO', 'WHAT' and 'WHERE', you will be able to understand 'WHY' this problem should be solved and how it will affect the stakeholders.

- Why the problem should be solved ?
- Is it worth-solving ?
- How will it benefit the stakeholders ?

PART B

INFOBOX 3.1

Who will Bell the Cat ?

In a small village full of mud houses and lots of crop, there lived a lot of mice. They used to enjoy plenty of food and plenty of space for their playing and jumping around.

As there were plenty of mice, a cat also came there in search of food. Every time she saw mice, she started chasing them and often succeeded in eating them. This scared mice a lot. They could not now play and jump around. They could not go out to search for their food.

Cat had become their major enemy and thus mice decided to call a meeting. So, the mice called a meeting to decide on a plan to free themselves of their enemy, the cat. At least they wished to find some way of knowing when she was coming, so they might have time to run away. Indeed, something had to be done, for their lived in such constant fear of her claws that they hardly dared stir from their dens by night or day.

For understanding the 4Ws Canvas, we have used the famous *Aesop fable* 'Who will Bell the Cat ?' If you haven't read it, please refer to InfoBox 3.1 which has this fable.

It is important to know the stakeholders for whom the problem exists and solution is to be found/developed. For instance, consider the fable of *'Who will bell the Cat ?'*,

- If 'Cat' approaches you to find the solution, then the stakeholder is the 'Cat' and the problem is 'Mice'.
- If 'Mice' approach you to find the solution, then the stakeholder is the 'Mice', and the problem is 'Cat'.

'WHO' Canvas of the Problem

In context of the fable 'Who will Bell the Cat ?'.

Stakeholders	Problem
Mice	Cat
Cat	Mice

Considering the same fable, 'Who will bell the Cat ?'.

'WHAT' Canvas of the Problem

Stakeholders	Problem	Nature of the Problem
Mice	Cat	★ Cat eats up the mice. ★ Mice always live in fear. ★ Mice struggle getting food as cat always chases them.
Cat	Mice	★ Cat does not get enough mice to eat. ★ Cat goes hungry at times.

Thus, it is important to understand the nature of the problem, with evidence, as part of 'What'.

For instance, considering our example fable 'Who will bell the Cat ?'.

'WHERE' Canvas of the Problem

Stakeholders	Location/Context of the problem
Mice	When mice come out in open (out of their dens) for their foods, Cat also comes there for its food scaring the mice.
Cat	When mice come out in open, Cat also comes there for its food but has to struggle as very few mice now come out in open.

'WHY' Canvas of the problem

Stakeholders	Why the Problem should be solved ?
Mice	To free them from fear when they come out in open for food or for play
Cat	To help her find staple supply of food

Finally, you can fill the Problem Statement template (empty shown below) using 4Ws canvas, by being explicit about WHO, WHAT, WHERE/WHEN and WHY ?

A Problem Statement Template

WHO has the problem? *Who else face the problem?*	WHAT is the problem? *What are the elements of the problem?*	WHERE/WHEN does the problem arise? *What is the context/location of it?*
_____ _____ _____ _____	_____ _____ _____ _____	_____ _____ _____ _____
WHY the problem should be solved? *Is it actually worth solving? Why hasn't it been solved earlier ? (Is it hard or low-priority ?)*		
_____ _____ _____ _____		

4Ws Canvas for AI Project for Clinical Trials

WHO has the problem? *Who else face the problem?*	WHAT is the problem? *What are the elements of the problem?* (The ***Pain-Points***)	WHERE/WHEN does the problem arise? *What is the context/location of it?*
Pharma Company conducting the Clinical Trials	**1.** Finding appropriate number of criteria matching patients for enrolling **2.** Problems in Monitoring the enrolled patients	**1.** Not getting enough health records for criteria matching in rural India **2.** Difficulty in convincing (enough) matching patients to get enrolled **3.** Difficulty in monitoring the enrolled patients through *pen* and *paper method* (there is a scope of lots of errors) **4.** Difficulty in conducting routine meetings with enrolled patients in case of some issue such as patients can't visit because of sickness or not in the town or because of extreme weather conditions etc.
WHY the problem should be solved? *Is it actually worth solving? Why hasn't it been solved earlier ? (Is it hard or low-priority ?)*		
1. It is important to get enough criteria matching patients to conduct clinical trials in order to launch a drug/vaccine in market. **2.** The enrolled patients must be minutely monitored CORRECTLY and TIMELY as it is important to study the impact of the drug under trial.		

Problem Statement of AI Project of Clinical Trials

MIDA Pharma Company (WHO) of *India* (WHERE) conducts pharmaceutical research and clinical trials for drugs and vaccines. To conduct the clinical trials, it requires enough number of criteria matching patients who agree to enrol for the clinical trials and follow the recommended routine and note down the details (WHAT).

It is important to find ways/solutions to find enough matching patients in rural India and to monitor the patients and to meet them in some unexpected circumstances such as sickness or weather problems etc. (WHEN) Less number of criteria matching patients and dropping out of patients often leads to the failure or delay of the clinical trials.

It is important to conduct clinical trials on enough number of patients in multiple phases; to monitor the enrolled patients minutely and timely, and to conduct routine meetings with the enrolled patients. This is required to determine the efficiency and efficacy of the drug/vaccine to assess if the drug/vaccine qualifies to be launched in market as a treatment. (WHY)

3.3 SET ACTIONS AROUND THE GOAL

In order to set the actions around your goal, you can use the **DOIT** principle as described below :

(*i*) **D**escribe the specific pain-point(s) or the goal(s), which you have already done through 4Ws Canvas. List down the stakeholders involved in the problem.

(*ii*) **O**utline multiple different ways to reach the goal or solve each of the pain-points after searching on the current actions taken to solve the problems and the issues around them.

(*iii*) **I**dentify the consequences of each of these options (of step 2), compare them in terms of cost and benefits, and choose the best plan.

(*iv*) **T**ake the most useful option and apply. Check the progress toward the goal while thinking and working around the ethics involved in the goal of your project.

PART B

LET US REVISE

- ***4Ws Canvas*** *is a framework to define the problem statement articulately (i.e., effective and clear definition) by finding factors like 'Who', 'What', 'Where'/'When' and 'Why'.*
- *The WHO canvas lists about the stakeholders for which the solution is to be developed.*
- *The WHAT canvas lists what exactly the nature of the problem is.*
- *The WHERE/WHEN canvas lists the location and context of the problem.*
- *The WHY canvas lists if the reasons are strong enough to find the solution of the problem.*
- *A problem statement is a short, clear description in words of the problem listing its stakeholders, their issue(s), context, and reasons to solve the problem.*
- *To set actions around the goal DOIT principle may be used as in Describe the pain-points; Outline multiple possible solution alternatives; Identify consequences of each alternative; Take the best possible alternative.*
- *Every project must be ethically followed.*

SESSION 4

Preparing and Assessing Data to be Collected

Problem Scoping – Explore and Understand Data

4.2 PREPARING AND ASSESSING DATA TO BE COLLECTED

Before you go to the next stage of AI Project Cycle, *Data Acquisition*, it is important to assess data and prepare for data collection.

4.2.1 What are the Data Features Needed ?

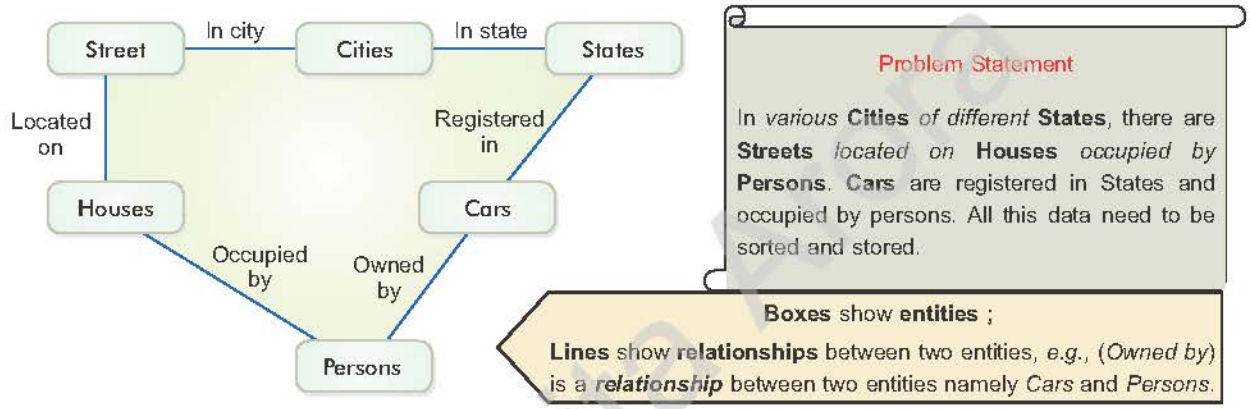
Every AI project requires data to work upon. A dataset (also known as ***database***) is represented through a collection of data objects (which are related to a common situation/theme or topic), known as these alternate names : ***records***, ***cases***, ***samples***, ***observations***, or ***entities***.

Data Feature

A **data feature** is an individual measurable property or characteristic of a data object being recorded or stored.

In order to determine the data features for your project, do the following :

- *Carefully go through the detailed problem statement.*
- *Identify the* ***entities*** *and the* ***relationships among the entities****. Entities are objects that exist either physically (such as a car) or conceptually (such as a bank account).*
- Your datasets would be storing the *entities and relationships*. For example, carefully look at the following diagram that lists the entities and relationships for a specific problem :



- For a dataset, there must be data features to identify each of the *entity* and *relationship* being stored.
- Each entity has some characteristics, which must be stored as per their relevance to your project. Say if your dataset involves an entity of a *car*. There would be many characteristics of a *car entity*, such as :
  - (*i*) Type of car (*e.g.*, Sedan or SUV etc.)
  - (*ii*) Colour of Car
  - (*iii*) Engine Type (Diesel or Petrol or electric etc.)
  - (*iv*) Price of the car
  - (*v*) Mileage of the car
  - (*vi*) Seating capacity

  From the above set of data features, you should only choose what is relevant as per your project, *e.g.*, to test mileage and efficiency of a car, the colour characteristic is irrelevant. So, choose only the relevant data features.
- There must be a way to store the relationships and entities involved in it.

4.2.2 Where can you Get the Data ?

Depending upon the *theme*, *topic* and *goal* of your AI project, the sources of data may be different. For instance, for an AI project based on a social welfare scheme for school education, the data would be mostly sourced from the schools and area benefitting from it.

For our *AI project of Clinical Trials*, we need to source data from the agencies and institutions that record this type of data, such as hospitals, clinics, and labs etc.

4.2.3 How frequent do you have to collect the data ?

The data should be collected at least **as often as the process is expected to change.**

Data collection frequency must be setup keeping in mind the following things :

- the aim of analysing or processing the data
- most frequently changing factor
- what additional information is important for the project ?

4.2.4 What happens if you do not have enough data ?

Data Sample

Collected data (at regular intervals) for the processing purposes, is known as **data sample**.

4.2.5 What kind of analysis needs to be done ?

Data analysis may be of the following types :

- Quantitative Analysis
- Qualitative Analysis

(*i*) ***Quantitative Analysis.*** If the data you have is quantitative in nature (such as numbers, amounts, counts etc.), then you should choose the quantitative method. In the case of quantitative analysis, tools like *average*, *mean*, *mode*, *median*, *regression*, *correlation*, *variance*, and *frequency* etc. are used.

(*ii*) ***Qualitative Analysis.*** If the data you have is qualitative in nature (*i.e.*, textual, factual data such as *sex* (*male* or *female*), *name*, *state of origin*, *citizenship* etc.), you should choose the qualitative method. In the case of qualitative analysis, tools like *content analysis*, *narrative analysis*, *discourse analysis*, and *ground theory* etc. are used.

Types of Data Validation

Depending upon the type of data (quantitative or qualitative), there are a number of validation types that can be used to check the data that is being entered. These are listed in the table below.

Data Validation

Data Validation refers to automatic computer check(s) to ensure that the data entered is sensible and reasonable.

Table : *Some Common Data Validation Techniques*

Validation type	*How it works*	*Example usage*
Check digit	The last one or two digits in a code are used to check the other digits are correct.	Bar code readers in supermarkets use check digits.
Format check	Checks the data is in the right format	The pin codes can be only numeric and of 6 digits only; the Aadhar card numbers are 12 digit numbers in 3 sets of 4 numbers each.
Length check	Checks the data isn't too short or too long	A password must be 6-12 letters long.
Lookup table	Only values in the lookup table can be accepted ; values different from the values of the table will be rejected.	The lookup table contains the admission numbers of students of class 10. So, for filling the board examination form for class 10, only the values in this lookup table are accepted and all other admission nos. are rejected.
Presence check	Checks that data has been entered into a field	For example, account number cannot be left blank for a bank transaction.
Range check	Checks that a value falls within the specified range	Marks must be between 0 to 100.
Spell check	Looks up words in a dictionary	For writing textual information like memo

4.2.7 How does the analysis inform the action ?

Data analysis converts the validated data into a useful information. Now, the data can be presented in a form that consists of the intelligence or insights that can be garnered from the data. For example, after data analysis, using tools like, *mean*, *median* etc., bulk of data may be represented visually in the form of *charts*, *graphs* and so on, that highlight a trend of some important information.

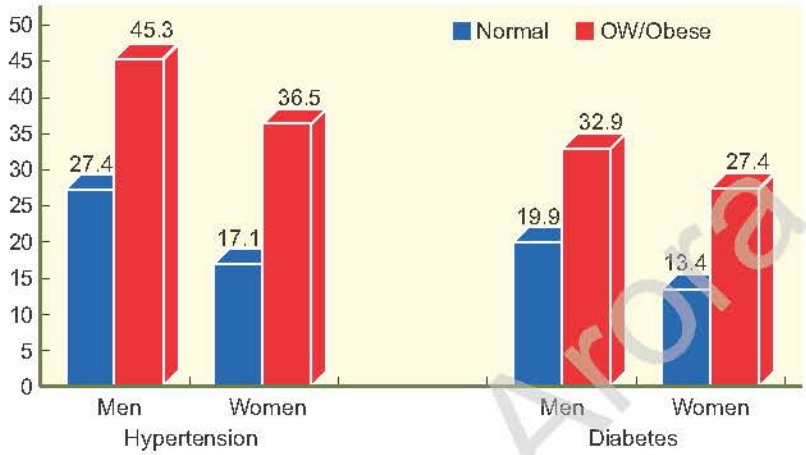


Figure 4.2 Hypertension and Diabetes with BMI more than 22

This section/content is not part of teacher-support PDF

Let Us Revise

- *A dataset is represented through a collection of data objects, also known as records, cases, samples, observations, or entities.*
- *A data feature is an individual measurable property or characteristic of a data object being recorded or stored.*
- *Data features are also alternatively called variables, characteristics, fields, attributes, or dimensions.*
- *Data features can either be of categorical type or numerical type.*
- *Data being processed must possess these qualities :* ***Relevance, Accuracy, Completeness, Timeliness,*** *and* ***Reliability.***
- *Data features must be chosen as per their relevance to be part of the dataset.*
- *Depending upon the theme, topic and goal of an AI project, the sources of data may be different.*
- *The 'frequency of data collection' means how often one should record data.*
- *It is very important to choose correct frequency of data collection otherwise, the project would yield incorrect results.*
- *Data analysis may be done quantitatively or qualitatively.*
- *Data validation refers to automatic computer check(s) to ensure that the data entered is sensible and reasonable.*
- *Some common data validation techniques are check digit, range check, format check, length check, lookup table, presence check, spell check etc.*
- *Data analysis presents data in more informative and summarised form that guides more informed decisions and actions.*

SESSION 5

Data Acquisition

- Identifying Data Requirements
- Finding Reliable Data Sources

5.2 IDENTIFYING DATA REQUIREMENTS

Here you will mainly do the following :

(*i*) Group together the **relevant data features** in logically related structures.

(*ii*) Be clear about the relationship of data in and outside the logical data structure.

(*iii*) Use consistent and standardised terminology and format.

5.3 FINDING RELIABLE DATA SOURCES

There can be many different sources wherefrom data may be collected. Most commonly used data sources are being discussed below.

1. Interview

It is one of the most effective sources of data gathering. In this method, an analyst talks to the users and clients who know about the system, its functions and flaws. A capable analyst is able to extract relevant information as per the new system requirements by asking relevant questions in interview.

Interview

An **interview** refers to a one-on-one conversation between an analyst and the users and clients to find out about the systems, its functions, shortcomings and flaws.

2. Survey

In Surveys, first the goal of the survey is ascertained and thereafter the questionnaires are drafted accordingly. Then using this, the responses of all users and stakeholders are documented.

Survey

A **survey** refers to a study of the opinions, responses, etc. of a group of stakeholders.

3. Observation

Under the observation method, the responsible person observes the team in a real working environment and gets ideas about the required data and its form, and subsequently documents the observation.

Observation

The **observation** method refers to human or mechanical watching, noticing or perceiving of what people actually do or what events take place in a specific working environment.

4. Application Programming Interface (API)

API is a specialized technique in which specific type of data is collected through the use of a programming interface, such as using social media programs' interface, data like people's most preferred game, most liked post, most used time etc. may be gathered.

API

An **API** refers to Application Programming Interface that works behind a popular software program or game to collect specific type of data pertaining to users' way of using that program.

5. Web Scraping

Web scraping, web harvesting, or web data extraction is data scraping used for extracting data from websites. A web scraper is a specialized tool designed to carry the web scraping.

Web Scraping

Web Scraping refers to a data collection technique using a tool called web scraper that extracts data from websites.

6. Sensors

Sensors or electronic sensors can measure various different parameters such as *weather, humidity, body temperature, blood pressure, heart beat, weight* and many more. For instance, you can see that modern medical diagnosis and wearables like *Fitbit, 'Apple watch'* make good use of sensors.

Note

Internet of Things (IoT) cannot function without sensors.

Sensor data is tasked with capturing information relevant to an environment or a particular task, so the data can be used to make process improvements for the purpose of saving money or increasing efficiency.

Sensors

Sensors are mini devices that can collect data about an environment or a body or a specific task.

7. Cameras

Cameras, because of their video recording and image capturing features have proven to be good data collection tools in various situations such as *traffic rules violations, automatic detection of flaws in design and outlook of products, places, buildings* etc.

Cameras

The method of **data collection using Cameras** is a way to collect data graphically or in video form about the look, design or action as per the requirements.

8. The Internet

Searching the Internet for data as per one's requirements is a commonly used technique. However, you should not take data directly from the Internet for the following two reasons :

(*i*) The data might not be authentic as its accuracy cannot be proved. Studies have shown that more than half of the data of the Internet comes from unreliable sources or is inaccurate.

(*ii*) Even if the data is reliable, it cannot be directly taken if it is copyright protected because of IPR (Intellectual Property Rights).

Thus, you can take data from the Internet only after ensuring the following two things:

(*i*) The source of data is authentic and reliable.

(*ii*) The data has been licensed for public use through licenses like ***Creative-commons, Copyleft*** and other open-source licenses.

There are some open-sourced websites hosted by the government, which host various types of data and information (such as ***data.gov.in***, ***india.gov.in***, ***mospi.nic.in/time-use-survey-0***, ***censusindia.gov.in*** and many others). You can use such websites for taking data if it meets your requirements.

9. Problem Reports

For every type of systems be it mechanical, electronic, software or human body systems, there are ways to find out problems (such as, for mechanical systems, system problems may be reported manually; for electronic and software systems, logs may be generated; and for human body systems, certain tests may be conducted to find out about the problems).

Reports also prove to be a reliable data source as one can look through such a report and sort the problems into some key areas that are troubling the client.

Problem Reports

Problem Reports are authentic documents listing the problems of a system either after conducting some tests or via logs prove to be a reliable data source.

Let Us Revise

- ***Data Acquisition*** *refers to processes, methods or systems that are used to collect information related to a certain theme or objective, to document or analyse some phenomenon.*
- *To identify data requirements, one needs to group together the relevant data features in logically related data structures.*
- *It is important to be clear about the relationships of data in and outside the logical data structure.*
- *Data must be gathered from reliable sources.*
- *Most commonly used data sources are Interview, Survey, Observation, API, Web Scraping, Sensors, Cameras, the Internet, Problem reports etc.*
- *An interview refers to a one-on-one conversation between an analyst and the users and clients to find out about the systems, its functions, shortcomings and flaws.*
- *A survey refers to a study of the opinions, responses, etc. of a group of stakeholders.*
- *The observation method refers to human or mechanical watching, noticing or perceiving of what people actually do or what events take place in a specific working environment.*
- *An API refers to Application Programming Interface that works behind a popular software program or game to collect specific type of data pertaining to users' way of using that program.*
- *Web Scraping refers to a data collection technique using a tool called web scraper that extracts data from websites.*
- *Sensors are mini devices that can collect data about an environment or a body or a specific task.*
- *The method of data collection using Cameras is a way to collect data graphically or in video form about the look, design or action as per the requirements.*
- *Problem Reports are authentic documents listing the problems of a system either after conducting some tests or via logs prove to be a reliable data source.*

SESSION 6

System Maps for the Problem Depiction

- System Map
- How to Draw a System Map ?

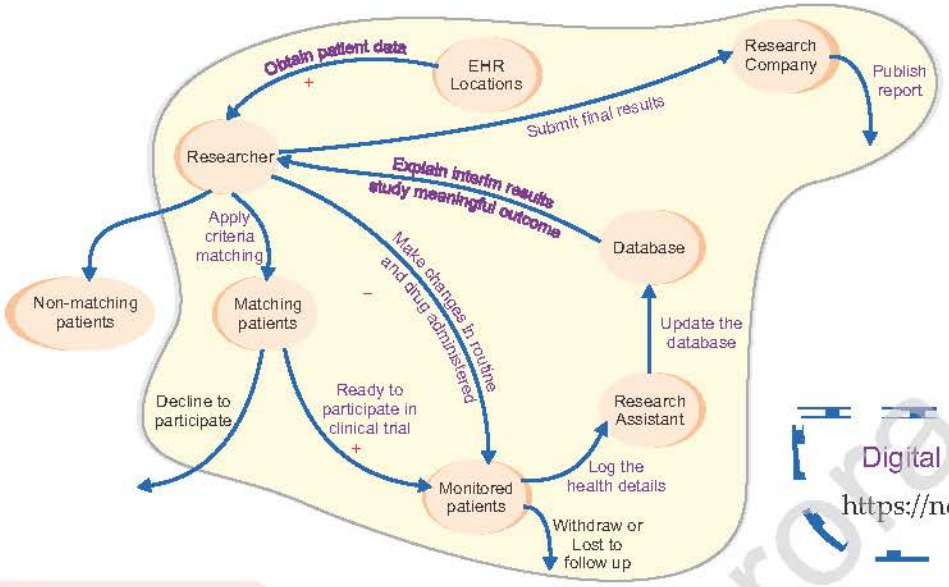
6.1 INTRODUCTION

Visual representation of any concept or problem is always useful and handy. There are many tools that help you visualise a bigger concept or a problem, to provide better understanding. One such tool is System Map, which can be used to depict a system graphically, showing its environment, its boundaries, components and how its components interact with one another. In this session, you learn to draw system maps for your project using its problem statement.

System Map

A **system map** is a visual representation of the components of a system and their interrelationships.

Let us now create a system map for our chosen project of AI, *i.e.*, Clinical Trails (goal 1)



Digital Resource Link

https://ncase.me/loopy/

Note

The scope of a system is shown through its boundary line. The components inside the boundary of System map are the core components of the system while outside the boundary shows the environment in which the system is operating. Arrows mark the relationship and interaction between components while +/– signs mark the impact of components over one another.

Let Us Revise

- *A system map is a visual representation of the components of a system and their interrelationships.*
- *The scope of a system is shown through its boundary line.*
- *The components inside the boundary of System map are the core components of the system while outside the boundary shows the environment in which the system is operating.*
- *The arrows show how the components are interrelated and influence each other.*

SESSION 7

Data Visualisation

- Data Visualisation and Its Need
- Ways to Visualise Data

7.2 DATA VISUALISATION AND ITS NEED

Data visualisation refers to the process of representing data visually or graphically, by using visual elements like charts, graphs, diagrams and maps etc.

Data Visualisation

Data visualisation refers to the process of representing data visually or graphically, by using visual elements like charts, graphs, diagrams and maps etc.

Need for Data Visualisation

Data visualisation is important and useful for understanding and comprehending the information stored in data. The importance of data visualisation is summarised as follows :

(*i*) Data visualisation is a powerful way to represent a bulk of data in a collective visual form.

(*ii*) It is a way to explore data with presentable results.

(*iii*) Data visualisation makes it easy to interpret and comprehend data.

(*iv*) It becomes easier to see the trends, relationships and trends of data through data visualisation.

(*v*) Data visualisation is useful for combining categories of data and thereby reducing data for processing.

This section/content is not part of teacher-support PDF

Let us now briefly talk about these data visualisation tools, one by one.

1. SCATTER CHART (Used with numeric type of data)

An XY (scatter) chart either shows the relationships among the numeric values in several data series or plots two groups of numbers as one series of *XY* coordinates.

How to draw?

The scatter chart is drawn by plotting the independent variable on the horizontal axis *X*, the dependent variable on the vertical axis *Y* and then by marking data points as per their *XY* values.

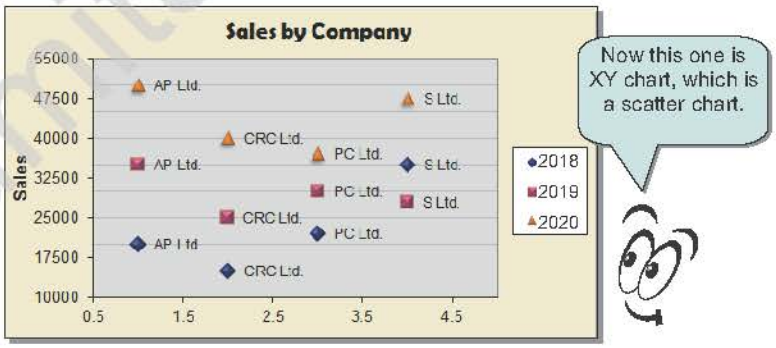


Figure 7.1 A Scatter Chart

2. BUBBLE CHART (Used with numeric type of data)

A bubble chart is primarily used to depict and show relationships between numeric variables with marker size as additional dimension. Bigger marker means bigger value.

How to draw?

The bubble chart is drawn by plotting the independent variable on the horizontal axis (*X*), the dependent variable on the vertical axis (*Y*) and then by marking bubbles at their *XY* values. The *Y* values will determine the bubble size.

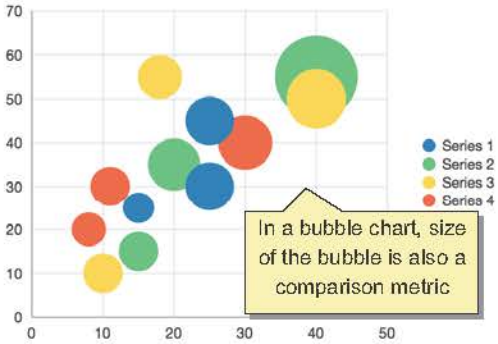


Figure 7.2 A Bubble Chart

3. LINE GRAPH (Used with numeric type of data)

A line chart shows trends in data at equal intervals. Line charts are useful for depicting the change in a value over a period of time.

How to draw?

The line chart is drawn by plotting the independent variable on the horizontal axis (*X*), the dependent variable on the vertical axis (*Y*) and then by marking data points as per their *XY* values. Then a line is drawn by joining the marked data points.

Figure 7.3 A Line Chart

4. PIE GRAPH (Used with numeric type of data)

A pie chart shows the proportional size of items that make up a single data series to the sum of the items.

How to draw?

The pie chart represents single data series, whole of which represents full circle (360°). Each data value is calculated as a percentage of whole and drawn as a pie of the circle.

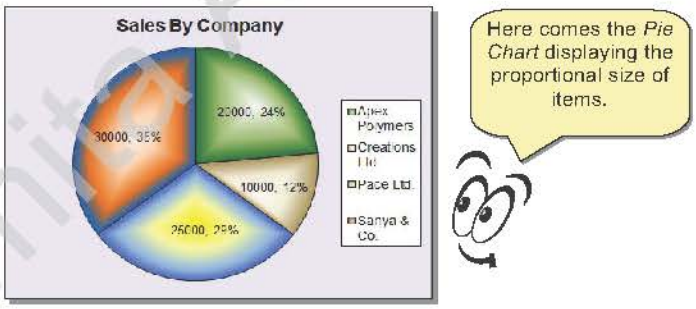


Figure 7.4 A Pie Chart

5. BAR GRAPH (Used with numeric type of data)

A bar chart illustrates comparisons among individual items, mainly of number types.

How to draw?

The bar chart is drawn by plotting the independent variable on the horizontal axis (*X*), the dependent variable(s) on the vertical axis (*Y*) and then by marking bars for their *Y* values.

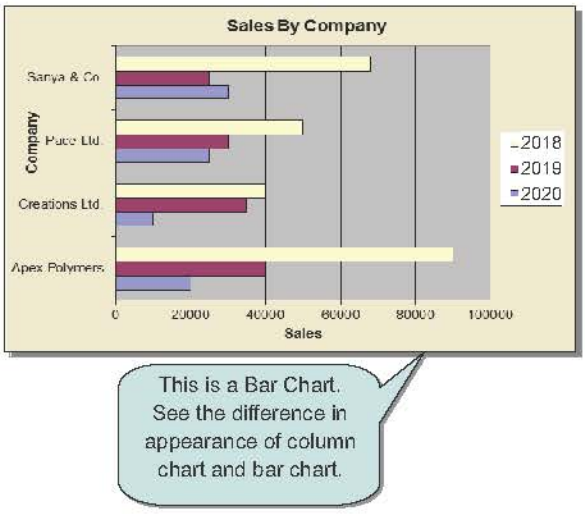


Figure 7.5 A Bar Chart

6. HISTOGRAM (Used with numeric type of data)

A histogram is used to summarize discrete or continuous data by showing the number of data points that fall within a specified range of values (called "**bins**"). Unlike a bar chart, there are no gaps in between in a histogram.

How to draw?

Like bar chart, rectangles of varying height are used to represent the frequency of different values of the continuous variable (*Y* values). There are no spaces between the rectangles.

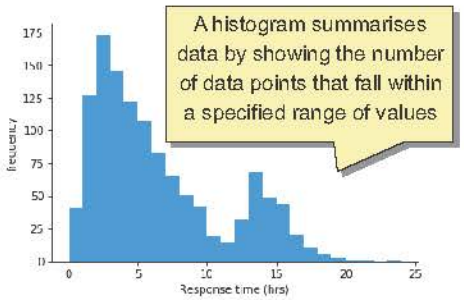


Figure 7.6 A Histogram Chart

7. CHOROPLETH (Used with processed numeric data linked with textual units)

Choropleth maps are used with statistical data (numeric, processed data) attached to enumeration units (textual data *e.g.*, counties, provinces, states etc.) to depict data for geographic regions. For example,

- world map of income tax rates, country wise.
- world map of Covid 19 spread, country wise.
- map showing the percentage increase in real estate value, state wise.

How to draw ?

In the region map, firstly the statistical values are written in the sub-regions. Then the sub-regions are filled with the corresponding colour for that value.

Covid 19 spread all over the world

Figure 7.7 Choropleth Chart

8. HEAT MAP (Used with numeric data depicted through colour codes)

A **heat map** is a graphic representation of data in which values are represented by colours. Some examples of heat maps are :

- A geographical heatmap representing areas of high and low density of a certain parameter (population density, network density, etc.) by displaying data points on a map through different colours.
- A stock index heatmap depicting prevailing trends in the market through colours, *e.g.*, cold-to-hot colour scheme to indicate which stock options are bullish and which are bearish.

How to draw?

The region is divided into smaller squares. Then each square is filled with the colour code as per the data it is storing.

India Climate heat map

Figure 7.8 Heat Map

9. TIMELINE (Used to represent all types of data against time)

A Timeline Chart shows a series of events in chronological order. It can be used to depict the order of historical events, critical milestones of a project schedule, and so on.

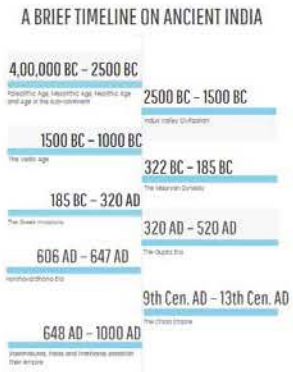


Figure 7.9 Timeline Chart

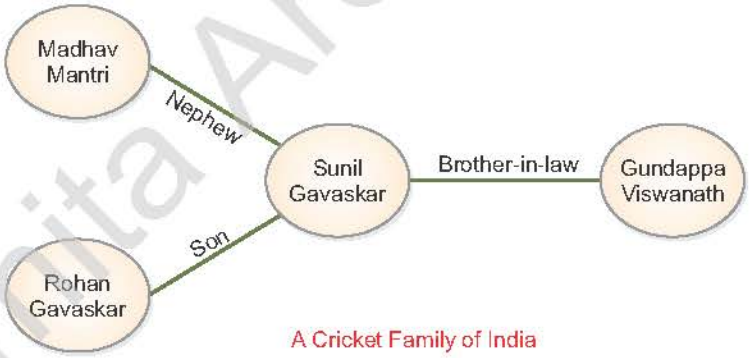
How to draw?

Draw a horizontal/vertical line with ends marking the start and end dates. Mark the points on the line for each of the events. Mark at each point the date and event.

10. NODE LINK DIAGRAM (Used with all types of data)

A node-link diagram shows how things are interconnected through the use of nodes/vertices and link lines to represent their connections and the type of relationships between a group of entities.

These are used in many applications, for example, for analysis of social networks or mapping product sales across geographic areas and many similar ones.



A Cricket Family of India

Figure 7.10 Node Link Diagram

How to draw?

Points are represented as nodes (vertices) that are linked by lines (edges).

11. WORD CLOUD (used with textual data)

The word cloud data visualisation technique represents the frequency of a word within a body of text with its relative size in the cloud. This technique is used on unstructured data as a way to display high- or low-frequency words.

How to draw?

The frequency of each word determines its weight, which determines its priority. The words with the highest priority get drawn first, and will be drawn with larger font-size.

Figure 7.11 Word Cloud

Let Us Revise

- *Data visualisation refers to the process of representing data visually or graphically, by using visual elements like charts, graphs, diagrams and maps etc.*
- *Data visualisation is important and useful for understanding the information stored in data.*
- *An XY (scatter) chart either shows the relationships among the numeric values in several data series or plots two groups of numbers as one series of XY coordinates.*
- *A bubble chart is primarily used to depict and show relationships between numeric variables with marker size as additional dimension. Bigger marker means bigger value.*
- *A line chart shows trends in data at equal intervals. Line charts are useful for depicting the change in a value over a period of time.*
- *A pie chart shows the proportional size of items that make up a single data series to the sum of the items.*
- *A bar chart illustrates comparisons among individual items, mainly of number types.*
- *A histogram is used to summarize discrete or continuous data by showing the number of data points that fall within a specified range of values (called "bins").*
- *Choropleth maps are used with statistical data (numeric, processed data) attached to enumeration units (textual data e.g., counties, provinces, states etc.) to depict data for geographic regions.*
- *A heat map is a graphic representation of data in which values are represented by colours.*
- *A Timeline Chart shows a series of events in chronological order.*
- *A node-link diagram shows how things are interconnected through the use of nodes/vertices and link lines to represent their connections and the type of relationships between a group of entities.*
- *The word cloud data visualisation technique represents the frequency of a word within a body of text with its relative size in the cloud.*

SESSION 8

Modelling

- Types of AI Technologies
- AI Modelling
- Decision Trees
- Identifying Patterns/Images

8.2 TYPES OF AI TECHNOLOGIES

There are many AI technologies and systems. But most used of these are **Machine Learning** and **Deep Learning**.

- **Machine Learning (ML)** is a branch of AI that enables machines to automatically learn and improve at tasks with experience and by the use of data. ML based machines undergo lots of repetitions of taking data and testing it; these then keep track of when things went wrong or right, and keep improving their results.

> Machine Learning
>
> **Machine Learning (ML)** is a branch of AI that enables machines to automatically learn and improve at tasks with experience and by the use of data.

- **Deep Learning (DL)** is a subset of machine learning where learning takes place through examples. Deep Learning computer-models filter the input data using layers and rules-based algorithms to predict and classify information. Tasks like speech and image recognition are performed through deep learning systems. Driver-less cars are being developed using deep learning technologies.

> Deep Learning
>
> **Deep Learning (DL)** is a subset of machine learning where learning takes place through examples by filtering the input data using layers and rules-based algorithms to predict and classify information.

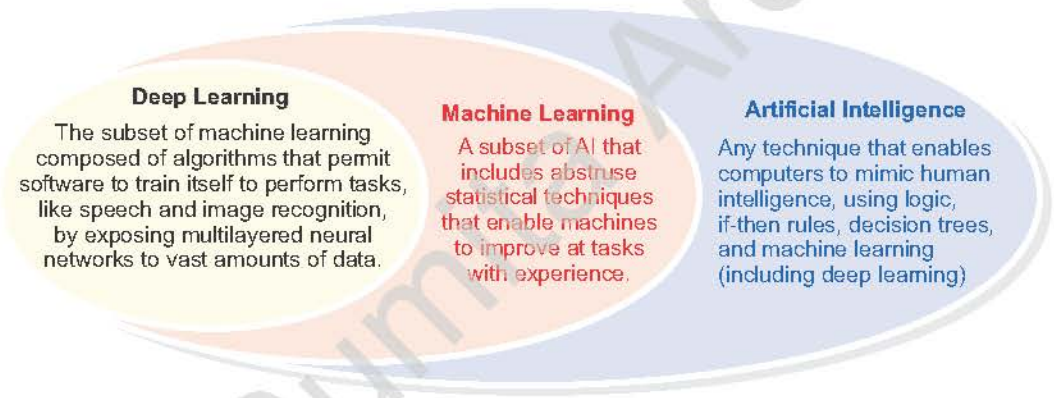


Figure 8.1 AI and its Subsets

8.3 AI MODELLING

The process of developing AI based algorithms (AI model), is called *AI modelling*. An AI model is a program or algorithm that utilizes a set of data that enables it to recognize certain patterns. This allows it to reach a conclusion or make a prediction when provided with sufficient information, *i.e.*, give intelligent outcomes.

AI Modelling

The process of developing an AI algorithm (AI model) is called **AI modelling** where code is developed to produce intelligent outcomes using a set of data.

There are many types of AI models which are popular today :

(*i*) ***Linear Regression.*** Linear regression models use statistics and supervised learning. These are widely used in banking, healthcare, insurance, etc.

(*ii*) ***Deep Neural Networks (DNN).*** These are based on interconnected units known as *artificial neurons* (inspired by the neural network of the human brain). DNN models find applications in *speech recognition, image recognition, natural language processing* and so on.

(*iii*) ***Logistic Regression.*** This is similar to the Linear regression model, except that it is only used in solving classification-based problems.

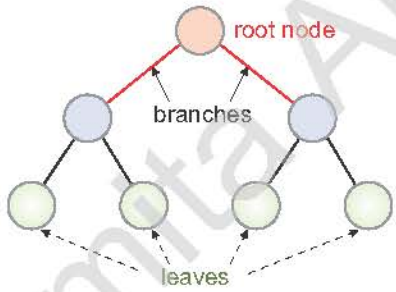
(*iv*) ***Decision Trees (DT).*** The Decision Tree (DT) model is used to arrive at a conclusion based on the data from past decisions. In this model, the data is divided into smaller portions resembling the structure of a tree (hence the name). This model can be applied for both regression and classification problems.

8.4 DECISION TREES

A Decision Tree is a structure where root is at the top. The top node, the **root node**, asks a question, which can have multiple answers. From here, the tree splits into multiple subtrees, based on each answer, with the help of arrows called **branches**. Each of these branches again asks a decision question and based on its answers, more branches diverge. The decisions are termed as the **leaves** of the tree. The final decision is available on the final nodes, known as **leaves**.

Decision Tree Model

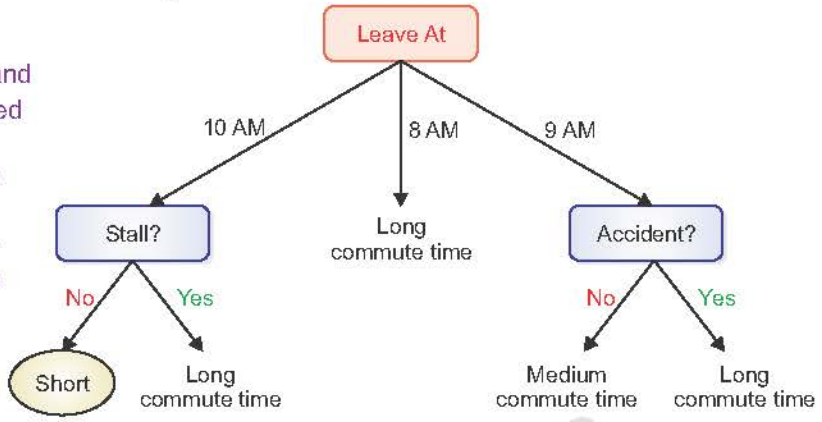
The **Decision Tree model (DT model)** is a machine learning predictive model based on a branching series of decision-making questions.



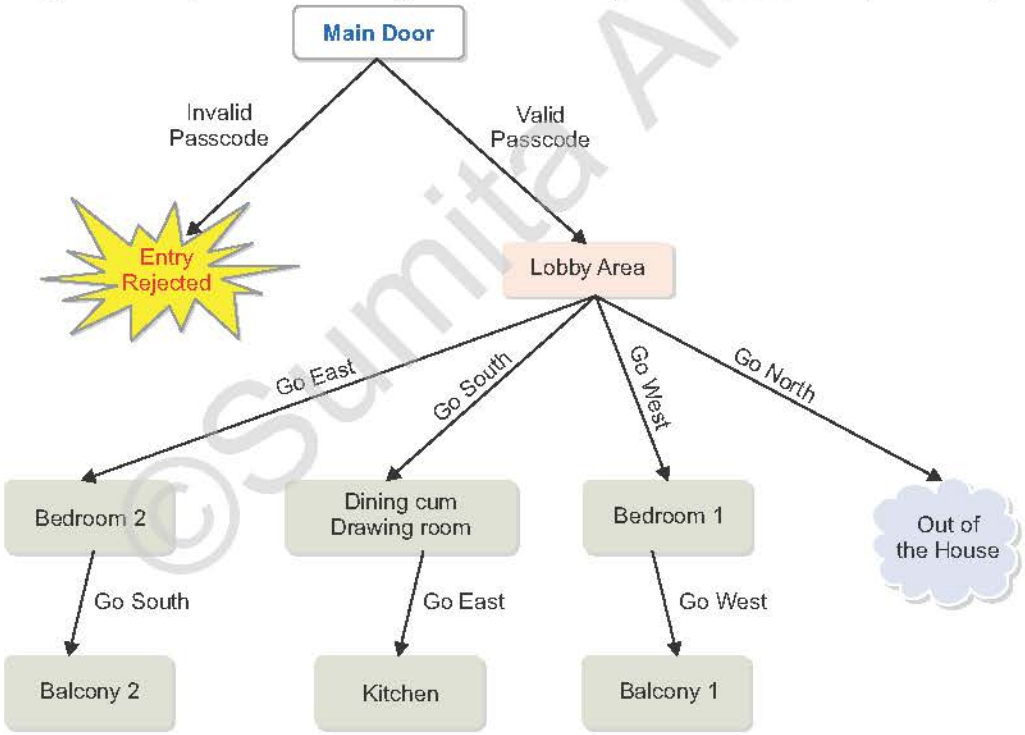
For example, consider the following decision tree :

If we leave at 10 AM and there are no cars stalled on the road, what will our commute time be?

Before 10 am, there is too much traffic on the road.



As you can see from the above example that the *Decision Tree model* is a **rule-based ML model** which helps a machine in predicting or reaching an outcome with the help of various decisions (or rules) fed to it. Recall our smart home navigation story (unit 1, session 2). Using a decision tree, you can depict how to navigate to different parts of your home (*see* below).

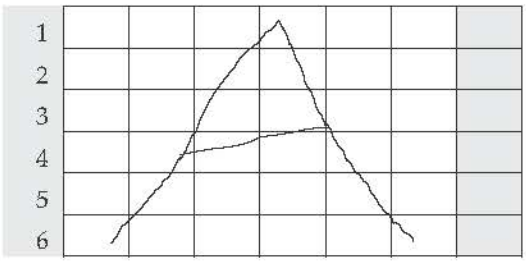


Some key terms related to a decision tree are being given below :

Root Node	This is the top node from where the decision tree begins.
Branch or Sub-Tree	A part of the entire decision tree is called a branch or sub-tree.
Splitting	It is the act of dividing a node into two or more sub-nodes based a decision question.
Decision Node	It is a node (of decision tree) that splits based on a decision question.
Leaf or Terminal Node	This is the end of the decision tree where it cannot be split into further sub-nodes. It holds an outcome value.

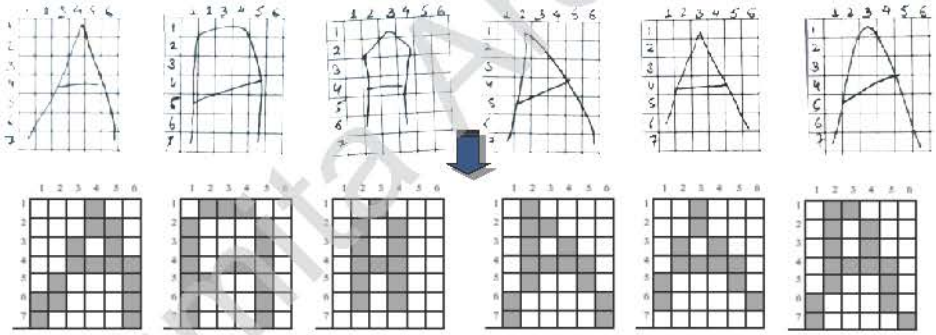
Activity PIXEL IT – LEARN HOW A COMPUTER LEARNS TO IDENTIFY ALPHABETS

1. Distribute a blank printed 6 x 6 matrix with row numbers marked to all the students in your class and ask them to write a capital letter on the matrix, something as shown here.

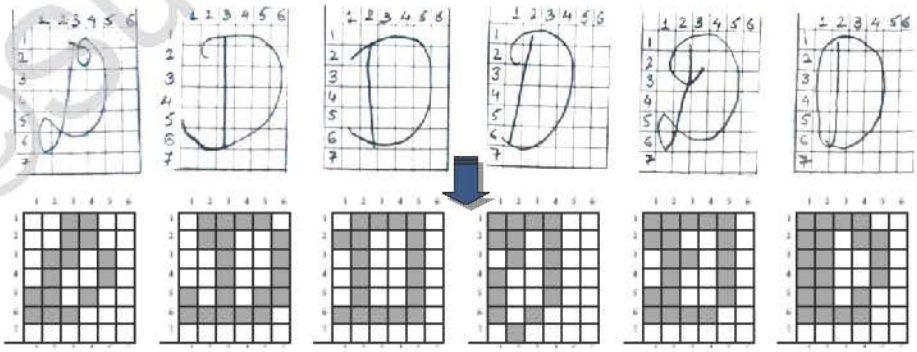


2. Collect the sheets and make stacks of papers having the same letters.
3. Then for each stack of same letters, shade the boxes that have some line on them, *e.g.*,

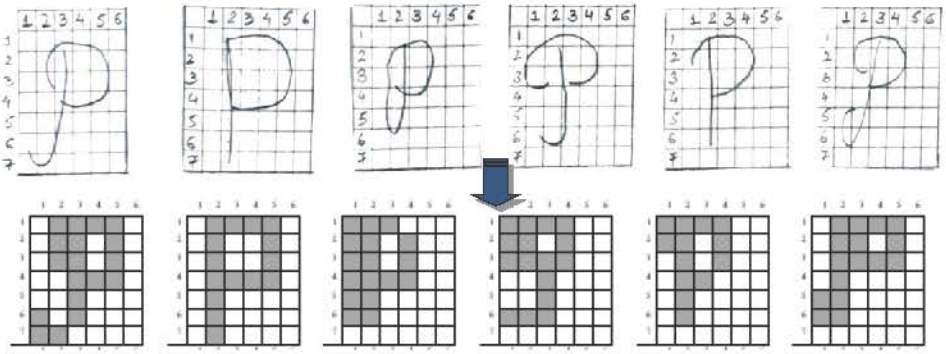
Letter A's stack



Letter D's stack

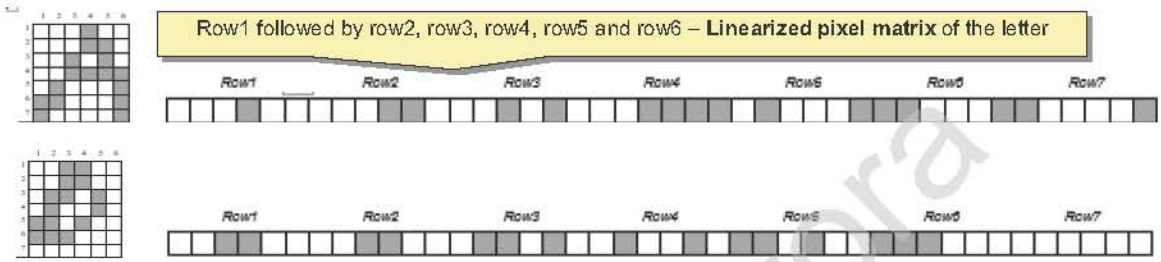


Letter P's stack



That is, you need to pixelate each of the letter. As you can see above, for each box having a part of the letter's shape is filled with a colour. Similarly, Computer creates pixelated image of the letter/image internally by filling the pixels lying on the shape of the image/letter.

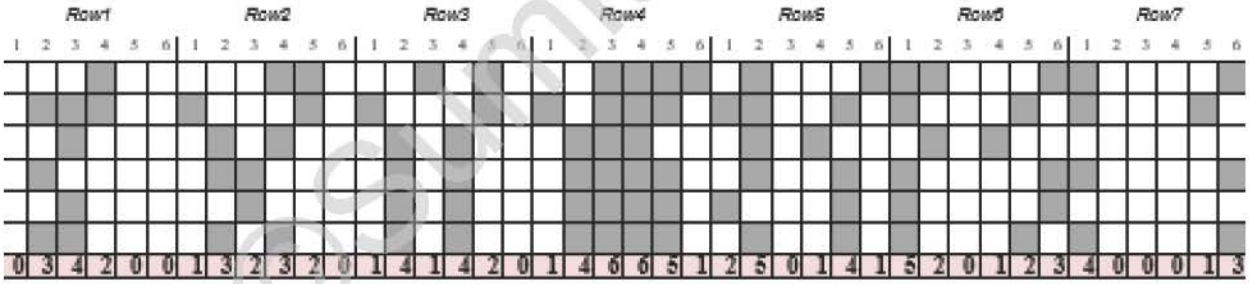
4. In the next step, cut each row of the pixelated letter and put them in the form of connected rows, *i.e.*, row 1's pixels, then row 2's pixels, then row 3's pixel and so on. There should not be any overlapping or gaps. (*see* below – two letters are shown for your reference).



5. Repeat the above step for each of the letters, *i.e.*, linearize all the letters.

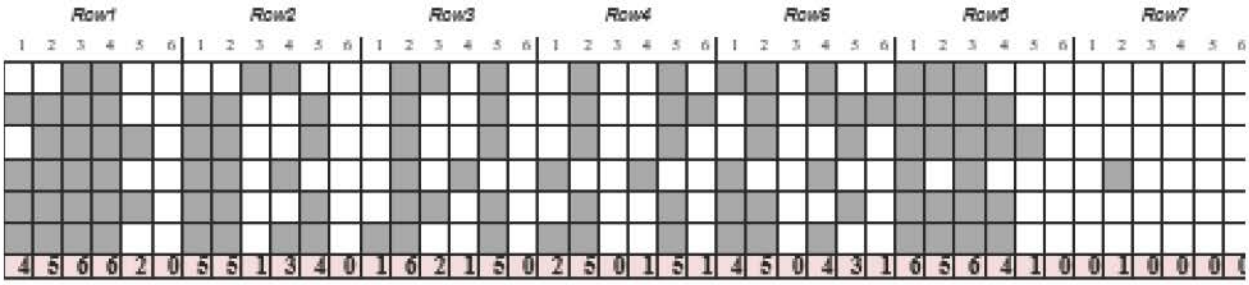
6. Put linearized pixel matrices of same letters one below another and write below each column the total number of shaded pixels in that column.

All letter A's (six in count)



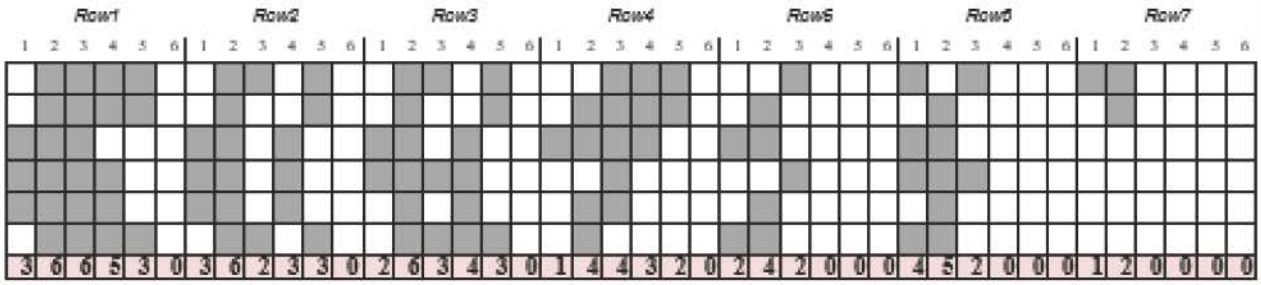
Number obtained with ***six A*** letters : **0342001323201414201466512501415201234000 13**

All letter D's (six in count)



Number obtained with ***six D*** letters : **4566205513401621502501514504316564100100 00**

All letter P's (six in count)



Number obtained with six ***P*** letters : **366530362330263430144320242000452000120000**

7. See, with six input pixel matrices for a letter, computer reached to a number. If the input number of letters are huge in number, say 60000 letters or 600000 letters, then computer would similarly reach to a number internally for an image/pattern or a letter. This input data acts as training data, which a computer used to create a model.

8. Now every time you show an image/pattern or a letter, computer would create its pixelated multiple copies with little variations and reach to a number. Then that number would be compared with the number computer prepared with training data. So computer would reach to a decision in terms of probability, such as 70% it is letter *P*, 30% it is *D* and so on (as you saw it yourself in the previous activity).

LET US REVISE

- *Artificial Intelligence (AI) refers to ability of modern machines and computers to mimic human intelligence. AI machines work through data, algorithms and models.*
- *Most used of AI technologies are Machine Learning and Deep Learning.*
- *Machine Learning (ML) is a branch of AI that enables machines to automatically learn and improve at tasks with experience and by the use of data.*
- *Deep Learning (DL) is a subset of machine learning where learning takes place through examples by filtering the input data using layers and rules-based algorithms to predict and classify information.*
- *AI can either be Data driven (also called learning-based AI) or model driven (also called Rule based AI).*
- *Rule Based AI (Model driven AI) refers to the branch of AI where models are developed using the algorithms having pre-defined labels, rules, patterns and relationships.*
- *The Learning based AI (data driven AI) refers to that branch of AI where models are trained to learn by inputting them tons of data. Here there are no patterns, rules and relationships predefined by the developer, rather machine learns with each new input and comes up with own algorithm.*
- *The process of developing an AI algorithm (AI model) is called AI modelling where code is developed to produce intelligent outcomes using a set of data.*
- *The Decision Tree model (DT model) is a machine learning predictive model based on a branching series of decision-making questions.*
- *Decision trees are rule based AI models, which are suitable for smaller data sets. For bigger data sets, making decision trees is not feasible.*
- *Training data refers to a set of data that is used to train a computer to perform a task.*
- *Testing data refers to a set of data that is used to test the outcome of developed model (which was developed using the training data).*
- *Training data is comparatively very large than testing data.*

UNIT 3

Neural Network

IN THIS UNIT

SESSION 1

Introduction to Neural Network

What are Artificial Neural Networks (ANN) ?

Artificial Neural Network

An **Artificial neural network** (ANN) refers to a collection of ***connected computational units*** or ***nodes*** (called ***neurons***) along with ***biologically inspired computer programs*** designed to simulate the way in which the natural neural networks of human brain process information.

Since Artificial Neural Networks mimic human brains, it would be beneficial to know how natural(biological) neural networks work. So, let us first talk about that.

1.2.1 How Natural Neural Networks Work ?

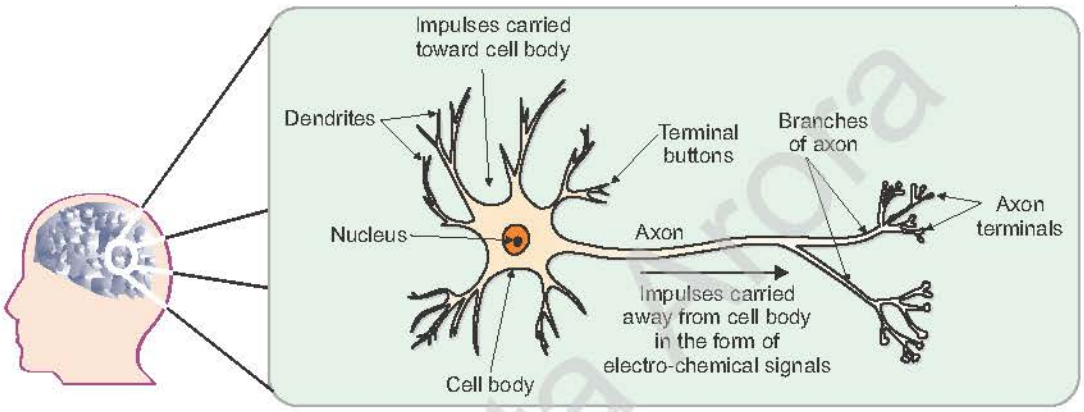
A human brain is a network of *neurons* where parts of each *neuron* (*see* Fig. 1.1) play an important role. A *neuron* broadly is comprised of *three* parts (Fig. 1.1) :

(*i*) ***Dendrites.*** It is a tree branch which receives information from other cells. They have leaf like structure on them called *spines*.

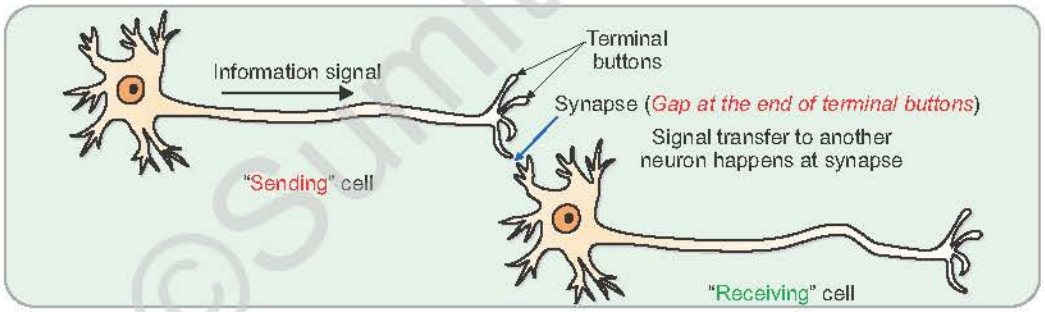
(*ii*) ***Soma/Cell Body.*** It contains the nucleus where the protein is made which is transmitted through *dendrites* and *axons*.

(*iii*) ***Axons and Axon Terminals.*** It is tree root and takes output from nucleus and transfers it to another neuron. Axon terminals at the end of axons are connected to the dendrites of another neuron through a small gap called **synapse**. Information signal transfers through synapse to the connected neuron's dendrites.

(*iv*) ***Terminal Buttons and Synapse.*** The **terminal buttons** are located at the end of neurons and are responsible for sending the signal on to other neurons. At the end of the terminal button is a gap known as a **synapse** [Fig. 1.1(*b*)] through which *neurotransmitters* carry the signal across to other neurons.



(*a*) Structure of human brain neurons



(*b*) Information transfer happens at synapse

Figure 1.1

The average human brain has about one hundred billion (100,000,000,000) neurons and each neuron has up to ten thousand (10,000) connections via the *dendrites*.

The *information signals* are passed via *electro-chemical processes* and can be thought of as simple electrical impulses that travel from *axon* to *dendrite*. The connections from one **dendrite** to **axon** are through gaps called **synapses** and these are the basic signal transfer points.

The components of an artificial neural network (ANN) are :

Neuron	equivalent of neuron (also called **node** or **neurode**)
Weighted inputs	equivalent to dendrites
Activation function	equivalent to soma (It defines how the weighted sum of the inputs is transformed into an output from a node or nodes in a layer of the network.)
Synapse	connection from a neuron to another that carry the information
Axon	output

In an ANN, a **node** (or **neurode**) is the artificial equivalent to a neuron. It consists of a set of weighted inputs (**dendrites**), an activation function (**soma**) and one output (**axon**). Information signal travels through multiple layers of connecting neurons before it is transformed in the form on an output.

Bias is an additional parameter in the Neural Network which is used internally as per some hidden rules and algorithm to adjust the output along with the *weighted sum of the inputs* to the neuron.

Structure of ANN

As you can see [Fig. 1.2(*b*)] that a Neural Network is divided into multiple layers. Each layer of an ANN consists of several ***artificial neurons*** called **nodes**. Each node has to perform a specific task and pass the information to the next layer.

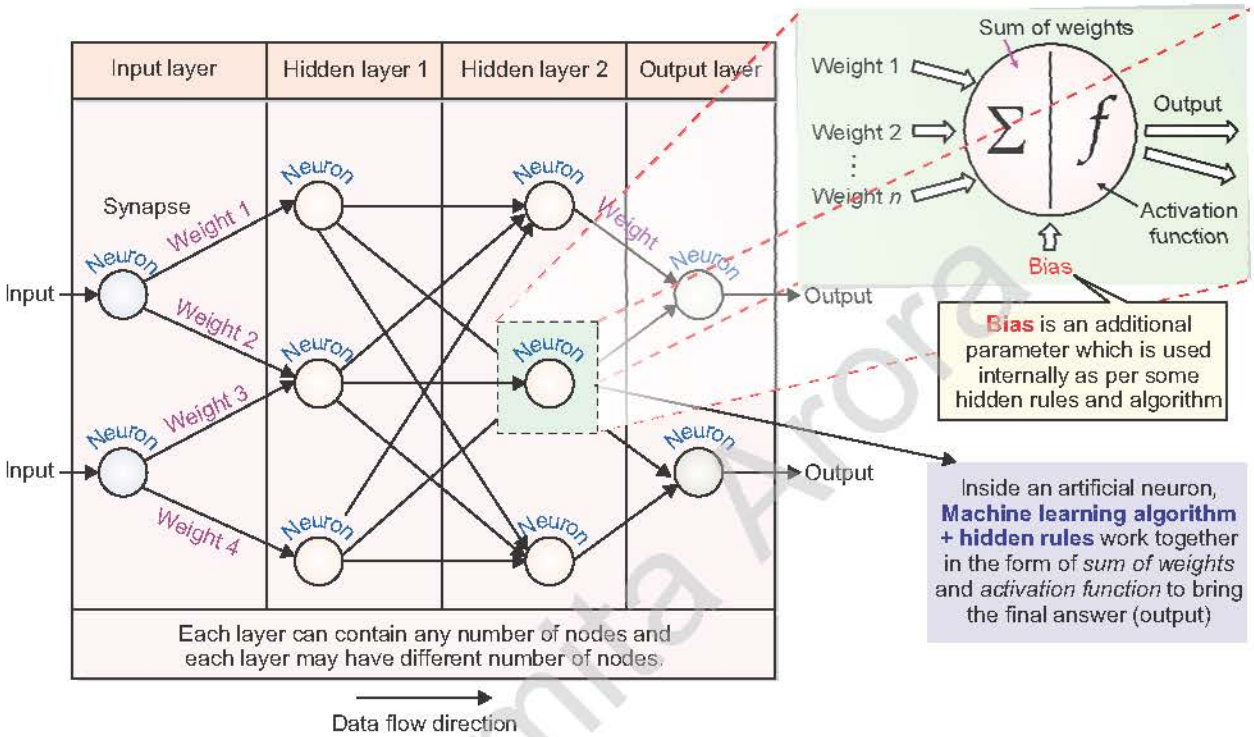


Figure 1.2 (*b*) Artificial Neural Network

There are *three* types of layers in an ANN :

(*i*) Input Layer

The first layer of a Neural Network is called the *input layer*, whose role is to acquire data and feed it to the Neural Network. The input layer carries out no processing, it just takes the input data and passes it on to the next connected layer.

(*ii*) Hidden Layer

Input layer is connected to a *hidden layer*, which is further connected to other hidden layers or to the final *output layer*. The role of hidden layers is to process the inputs and carry out a task. The processing at the hidden layers is carried out as :

Sum of weighted inputs
+ activation function (*i.e.*, machine learning algorithm)
+ hidden rules (such as getting additional parameters such as a bias)

There can be multiple hidden layers in an ANN, depending upon the complexity of the task(s) being performed. Hidden layers are not visible to the user. The processed output of a hidden layer is then fed to the subsequent hidden layer of the network.

Note

Each layer of an ANN can contain any number of nodes and each layer may have different number of nodes.

(iii) Output Layer

After processed data travels through multiple hidden layers, it (the final processed data) is finally fed to the final layer known as the **output layer**. The output layer simply provides the final output to the user. At the output layer also, no processing takes place; it only provides user-interface for the output.

This point onwards in this session, the term **"neural networks"** will refer to *Artificial Neural Networks*.

Back Propagation

Providing feedback about the difference from the correct output is known as **back propagation**.

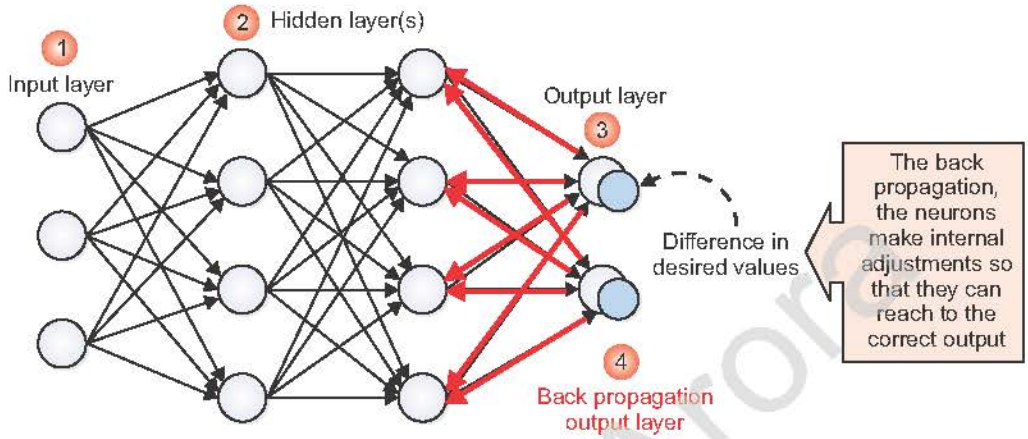


Figure 1.3 Back Propagation Helps Neural Networks Learn

Note

After initial training, Neural networks keep updating their internal mechanism of reaching to an output with every new feedback received.

1.2.4 Features of Neural Networks

From the above discussion, we can list the features of neural networks as follows :

(*i*) Neural networks have been developed to mimic the structure and working of human brain.

(*ii*) Neural networks evolve and automatically learn with each input and each new attempt.

(*iii*) Neural networks can work with big data sets.

(*iv*) The neural networks employ machine learning techniques to function and evolve.

1.2.5 Neural Networks and AI Models

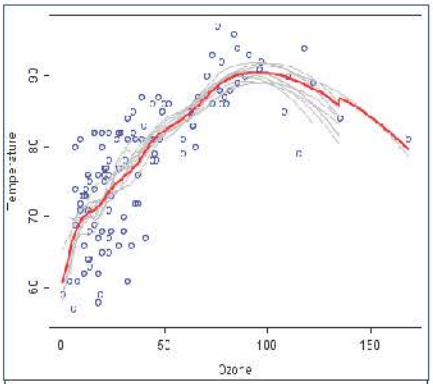
The neural networks have become so popular as these can be used with various types of AI models, as discussed below :

- Regression
- Classification
- Clustering

1. REGRESSION

This is a type of ***Rule-based AI model***. The goal of regression model is to build a mathematical equation from the available set of input data to produce the output. The mathematical equation is called the *mapping function*, which with the help of new input can predict the

possible output. See figure, where the blue dots represent the data, and the red line depicts the **mapping function** (mathematical equation). For example, if we have the data available about the drainage system, sanitation and cleanliness in an area and the number of dengue cases, we can use this data to train an AI model and in future we can predict the possibility of dengue cases using the data of an area.



Regression AI model

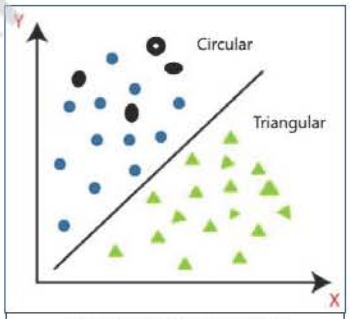
Regression models use continuous datasets.

Note

Neural networks can be applied to all popular types of AI models such as Regression model (a rule based model), Classification model (a rule based model), Clustering model (a machine learning approach model) etc.

2. CLASSIFICATION

This is also a type of *Rule-based AI model*. This AI model uses rules and labelled datasets to classify datasets into categories. A classification AI model involves assigning a class label to input examples and thus assigning them to a category. For example, if we want to train a model to identify if an image is of a specific alphabet, we need to train it with multiple images of alphabets along with their labels. The machine will then classify images on the basis of the labels and predict the correct label for testing data. Classification models use non-continuous, *i.e.*, discrete datasets.

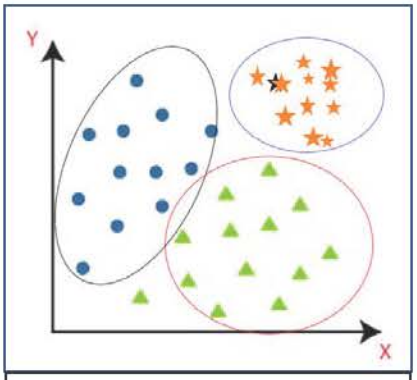


Classification AI model

Note **Regression** and **Classification**, both are **supervised learning models**.

3. CLUSTERING

This is a learning based (**Machine Learning approach**) model. Clustering is a way to group a set of data points in a way that similar data points are grouped together. Therefore, clustering algorithms look for similarities or dissimilarities among data points. Clustering is an **unsupervised learning method** so there is no label associated with data points. Clustering algorithms try to find the underlying structure of the data. For example, a clustering model can be input a dataset of customers' buying details in past one year ; it can then identify the customers who shopped the most; the customers who preferred buying certain type of things; customers who only bought essential items and so on.



Clustering AI model

Let Us Revise

- *Artificial neural network (ANN) refers to a collection of connected computational units or nodes (called neurons) along with biologically inspired computer programs designed to simulate the way in which the natural neural networks of human brain process information.*
- *The message transfer from a neuron is called **action potential**.*
- *In an ANN, a node (or neuron) is the artificial equivalent to a neuron. It consists of a set of weighted inputs (dendrites), an activation function (soma) and one output (axon). Information signal travels through multiple layers of connecting neurons before it is transformed in the form on an output.*
- *A Neural Network is divided into multiple layers (hidden layers in between input and output layers).*
- *Each layer of an ANN consists of several artificial **neurons** called **nodes**.*
- *Each **neuron** node has to perform a specific task and pass the information to the next layer.*
- *An activation function in a neural network defines how the weighted sum of the input is transformed into an output from a node or nodes in a layer of the network.*
- *Providing feedback about the difference from the correct output is known as back propagation.*
- *After initial training, Neural networks keep updating their internal mechanism of reaching to an output with every new feedback received.*
- *Neural networks can be applied to all popular types of AI models such as **Regression model** (a rule based model), **Classification model** (a rule based model), **Clustering model** (a machine learning approach model) etc.*
- *In supervised learning, ANNs learn through labelled data of tasks with known outcomes.*
- *In unsupervised learning, ANNs learn through unlabelled data of tasks with unknown outcomes.*
- *Some popular applications of neural networks are Face recognition, Speech recognition, Self-driven vehicles, medical diagnosis and many others.*
- *Deep learning is the most advanced form of machine learning where there are many hidden computational layers in a neural network (deep neural network) enabling an algorithm to powerfully analyse the input data.*

UNIT 4

Introduction to Python

IN THIS UNIT

SESSION 1

Introduction to Programming

- Activity 1 : Solve the Maze Puzzles
- Activity 2 : Play a Game on Code Combat

Programming

A **program** is a set of instructions given to a computer to carry out a task and the act of writing and executing programs is called **programming**.

SESSION 2

Introduction to Python

- Working in Python
- Getting Simple Input

2.2 WORKING IN PYTHON

Once you have Python installed on your computers, you are ready to work on it. You can work in Python in following different ways :

(*i*) in Interactive mode (*ii*) in Script mode

2.2.1 Working in Interactive Mode

Interactive mode of working means you type the command – one command at a time, and the Python executes the given command there and then and gives you output. In interactive mode, you type the command in front of the *Python command prompt* >>> .

For example, if you type **2 + 5** in front of the Python prompt, it will give you result as 7 :

```
>>> 2 + 5
7
```

Result returned by Python → 7

>>> 2 + 5 ← *command/expression given here*

To work in interactive mode, follow the process given below :

(*i*) Click **Start button** → **All Programs** → **Python 3.x** → **IDLE (Python GUI)** [see Fig. 2.1(*a*)]

Or

Click **Start button** → **all Programs** → **Python 3.x** → **Python** (command line)

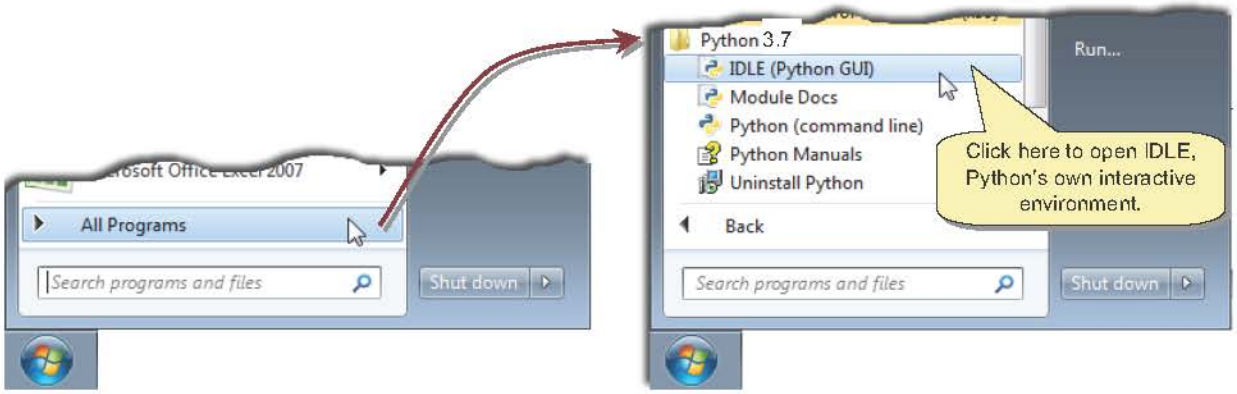


Figure 2.1 (*a*) Starting Python Shell

(*ii*) It will open Python Shell [*see* Fig. 2.1(*b*)] where you'll see the Python prompt (three '>' signs *i.e.*, >>>).

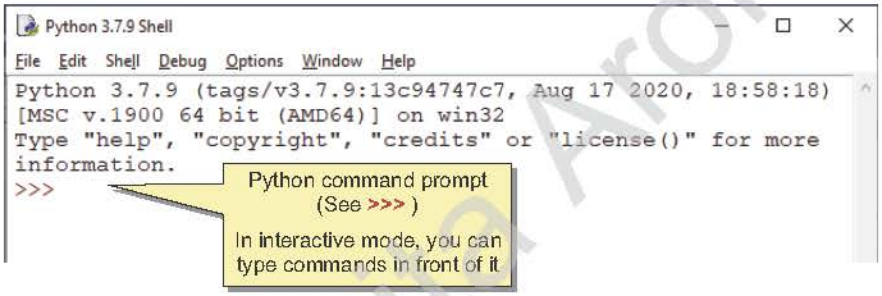


Figure 2.1 (*b*) Python's Interactive Interpreter – Python Shell.

Note

The interactive *interpreter of Python* is also called **Python Shell**.

(*iii*) Type commands in front of this Python prompt and Python will immediately give you the result [*see* Fig. 2.1(*c*)].

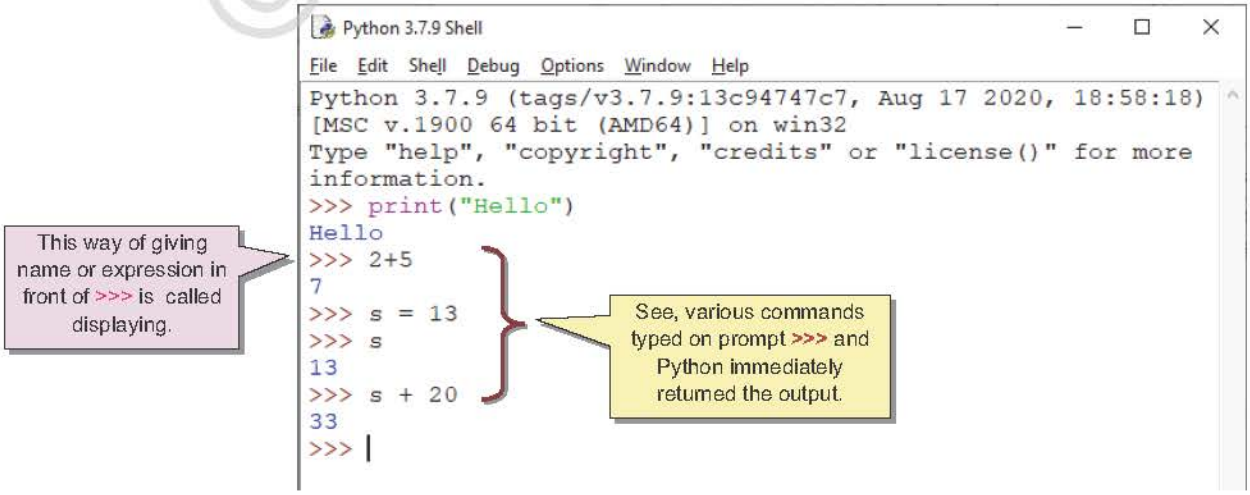


Figure 2.1 (*c*) Interactive Commands and their Output in Python Shell

For example, to print string "Hello" on the screen, you need to type the following in front of Python prompt (>>>)

```
>>> print ("Hello") ↵
```

And Python interpreter will immediately display string **Hello** below the command. To display, you just need to mention name or expression [Fig. 2.1(*c*)] in front of the prompt.

Note

Interactive mode proves very useful for testing code ; you type the commands one by one and get the result or error one by one.

Figure 2.1(*c*) shows you some sample commands that we typed in Python shell and the output returned by Python interpreter.

2.2.2 Working in Script Mode

What if you want to save all the commands in the form of a program file and want to see all output lines together rather than sandwiched between successive commands ? With interactive mode, you cannot do so, for these reasons :

- Interactive mode does not save the commands typed by you in the form of a program.
- The output is sandwiched between the command lines (*see* Fig. 2.2).

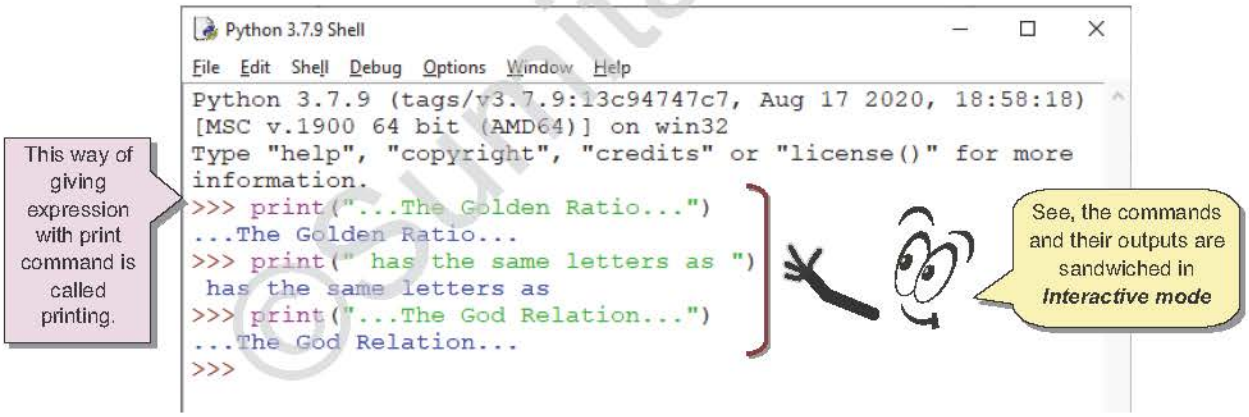


Figure 2.2 Output Sandwiched between Commands in Interactive Mode

The solution to above problems is the **Script mode**. To work in a script mode, you need to follow the steps given below.

Step 1 : Create Module / Script / Program file

Firstly, you have to create and save a module / Script / Program file.

To do so, follow these instructions :

(*i*) Click **Start button** → **All Programs** → **Python 3.x** → **IDLE.** [Fig. 2.3(*a*)]

(*ii*) Click **File** → **New** in IDLE Python Shell [Fig. 2.3(*a*)].

(*iii*) In the new window that opens, type the commands you want to save in the form of a program (or script) [Fig. 2.3(*b*)].

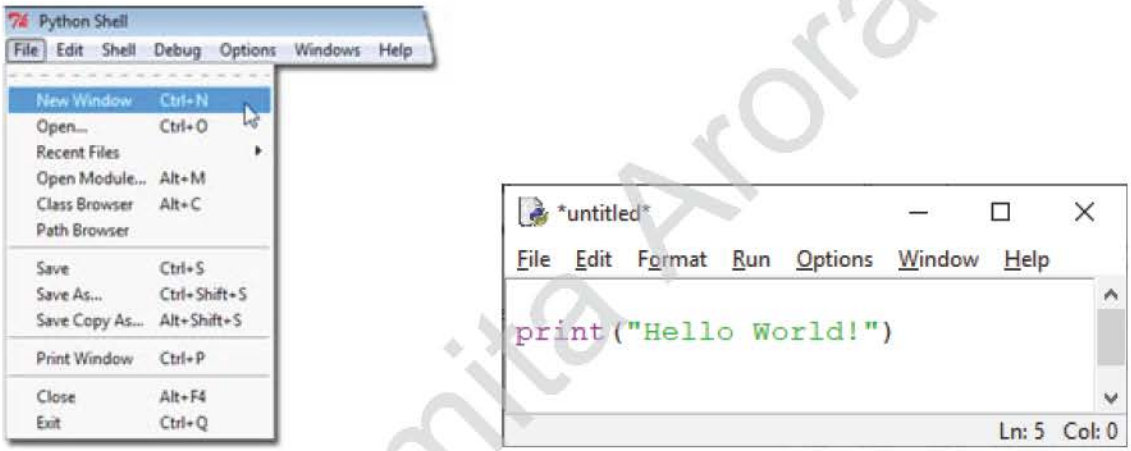
For instance, for the simple Hello World *program*, you'll need to type following line :

```
print (" Hello World ! ")
```

Note

You can display as well as print values in interactive mode, but for script mode, only **print( )** command is used to print results.

(*iv*) Click **File** → **Save** and then save the file with an extension **.py**. The Python programs has **.py** extension [Fig. 2.3(*c*)]. For instance, we gave the name to our program as **Interesting.py**.



(*a*) **File** → **New** Command in Python Shell

(*b*) Type Commands in New Blank File (Script mode)

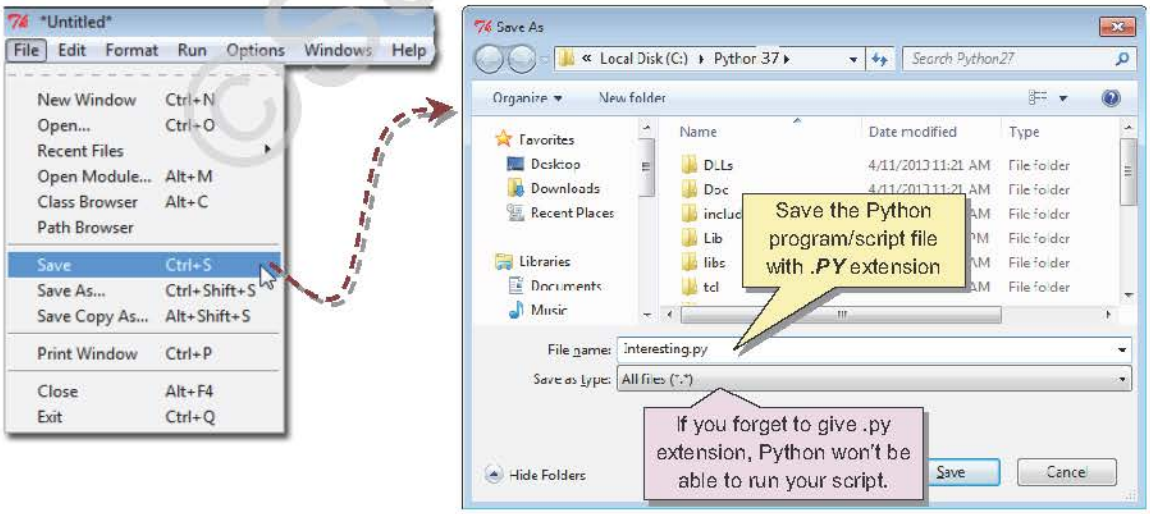


Figure 2.3 (*c*) Save File with .py Extension with File → Save Command (Script mode)

Now your program would be saved on the disk and the saved file will have **.py** extension.

Step 2 : Run Module / Script / Program File

After the program/script file is created, you can run it by following the given instructions :

(*i*) Open the desired program/script file that you created in previous Step 1 by using IDLE's **File → Open** command.

If the program/script file is already open, you can directly move to next instruction.

(*ii*) Click **Run → Run Module** command [Fig. 2.4(*a*)] in the open program/script file's window.

You may also press **F5 key**.

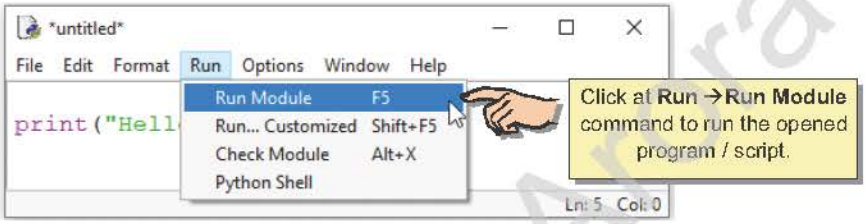


Figure 2.4 (*a*) Run → Run Module Command (Script mode)

(*iii*) And it will execute all the commands stored in module/program/script that you had opened and show you the complete output in a separate Python Shell window. [Fig. 2.4(*b*)]

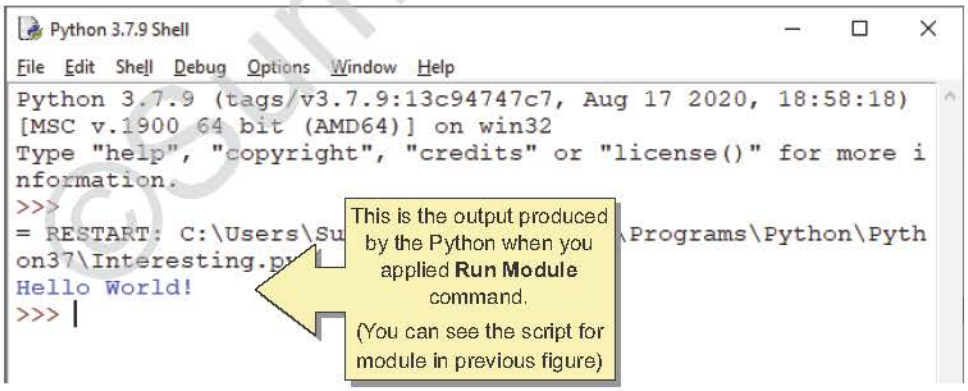


Figure 2.4 (*b*) Output of a Module-run is shown in the Shell Window

As you can see that with script mode, you can store all commands together in the form of a module/program/script and can get all output lines together. (No more command-output sandwiching :).

2.3 GETTING SIMPLE INPUT

Just like you used **print( )** function above, there is another function used for taking input. This function is **input( )**. You can use input as :

```
<name to store input value> = input(<message to be displayed (in quotation marks)>)
```

You can then use the name in which you stored the input value for printing. For example,

```
Studentname  = input("What is your name ?")
```

It will display the message and ask you to type the input (*see* below) :

The value that you type will be stored in the name you mentioned — in our case, it will be

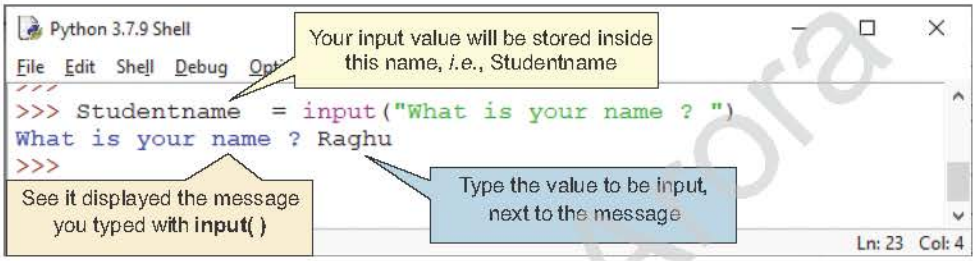


Figure 2.5

stored in **Studentname** as we used this name with input( ). You can now print this name with **print()** and it will show your input value as it stored your input value in that name (*see* Fig. 2.6 below)

In the same way you can take any input, *e.g.*,

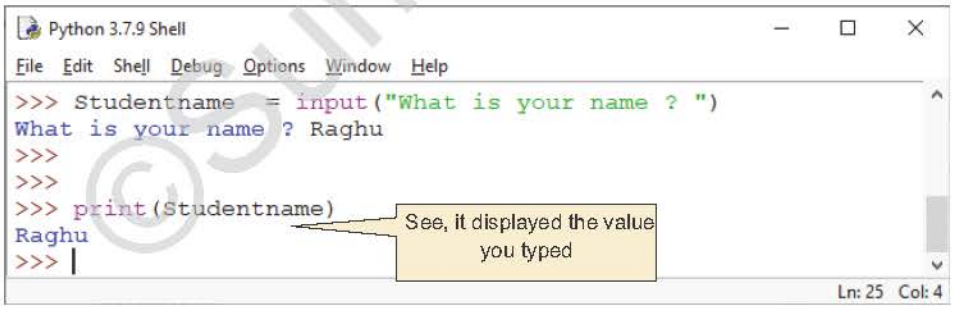


Figure 2.6

LET US REVISE

- *Python was developed by Guido Van Rossum in February 1991.*
- *In Python, one can work in two different ways : (i) Interactive mode, (ii) Script mode.*
- *Interactive mode does not save commands in the form of a program and also, output is sandwiched between commands.*
- *Script mode is useful for creating programs and then run the programs later and get complete output.*
- *Python is a case sensitive programming language.*

SESSION 3

Python Basics

- Variables and Expressions
- Data Types
- The input( ) Function Revisited
- Statements

This section/content is not part of teacher-support PDF

PART B

3.2 VARIABLES AND EXPRESSIONS

Variables

Variables are the named places to hold some values in a program.

3.2.1 Variables

Variables are the named places to hold some values in a program. A computer program uses variables to hold values. These values can be changed when needed.

For example, in above story, the ***four piggybanks*** and the ***envelope*** are acting like variables.

Variables are places to hold values

You can create a variable in Python by assigning a value to its name *e.g.*,

```
Age = 10
print(Age)
```

variable Age is created

Names given to variables are called **identifiers**. A variable name (**identifier**) can be used only after you have created it *e.g.*, consider the following Python code :

```
Age = 15
print(Age)
print(marks)
```

this statement will give error as marks variable is not defined.

To correct the above error, you may write :

```
Age = 15
marks = 75
print(Age)
print(marks)
```

Note

While creating and naming variables, make sure that it starts with a letter only. It can only contain letters, digits and underscore, *e.g.*, some valid variable names are : **File1**, **Total**, **Anvi_PB**. No other character or symbol should be used.

A variable's value can be changed as many times as required in a program *e.g.*,

```
marks = 75
print(marks)
marks = 82
print(marks)
```

*value of variable **marks changed.***

The above code will run without any error and produce the output as :

```
75
82
```

Example 3.1 *Declare variables (in Python) for the piggy banks used in above story.*

Solution.

```
AnvisPB = 172
CharusPB = 190
VasusPB = 196
VanshsPB = 142
```

3.2.2 Arithmetic Operators

The arithmetic operators are the symbols which depict some arithmetic calculation, such as + (*addition*), – (*subtraction*), * (*multiplication*), / (*division*). You used the plus (+) operator to perform addition of values of *variables* **AnvisPB**, **CharusPB**, **VasusPB** and **VanshsPB** and store in variable **Total**.

Arithmetic Operators

The **arithmetic operators** are the symbols which depict some arithmetic calculation.

To do arithmetic, Python uses arithmetic operators. Python provides operators for basic calculations, as given below :

+	addition	/	division	%	remainder
–	subtraction	//	floor division	* *	exponentiation
*	multiplication				

1. Addition operator +

The arithmetic binary operator + adds values of its operands and the result is the sum of the values of its two operands. *For example,*

4 + 20 results in 24.
a + 5 (a = 2) results in 7.
a + b (a = 4, b = 6) results in 10.

2. Subtraction operator –

The – operator subtracts the second operand from the first. *For example,*

14 – 3 evaluates to 11.
a – b ($a = 7$, $b = 5$) evaluates to 2.
x – 3 ($x = -1$) evaluates to – 4.

The operands may be of number types.

3. Multiplication operator *

The * operator multiplies the values of its operands. *For example,*

3 * 4 evaluates to 12.
b * 4 ($b = 6$) evaluates to 24.
p * 2 ($p = -5$) evaluates to –10.
a * c ($a = 3$, $c = 5$) evaluates to 15.

The operands may be of *integer* or *floating point number* types.

4. Division operator /

The / operator divides its first operand by the second. *For example,*

100/5 evaluates to 20.
a/2 ($a = 16$) evaluates to 8.
a/b ($a = 15.9$, $b = 3$) evaluates to 5.3.

The operands may be of *number* types. The behaviour of this operator depends upon the operands' data types (in *Python 2.x*) ; *'How a data type affects the functioning of division operator /'* will be clear to in the next section – *Expressions*.

5. Floor Division operator //

Python also offers another division operator //, which performs the floor division. The floor division is the division in which only the whole part of the result is given in the output and the fractional part is truncated.

To understand this, consider the third example of division given in division operator / , *i.e.*,

$a = 15.9, \ b = 3,$

a/b evaluates to 5.3.

Now if you change the division operator / ; with floor division operator // in above expression, *i.e.*,

If $a = 15.9$, $b = 3$,

$a//b$ will evaluate to 5.0

See, the Fractional part 0.3 is discarded from the actual result 5.3

Consider some more examples :

7//3 evaluates to 2

6.5//2 evaluates to 3.0

Note

Floor division (//) truncates fractional remainders and gives only the whole part as the result.

The operands may be of number types.

6. Modulus operator %

The % operator finds the modulus (*i.e.*, remainder but pronounced as *mo-du-lo*) of its first operand relative to the second. That is, it produces the remainder of dividing the first operand by the second operand. *For example*,

19 % 6 evaluates to 1, since 6 goes into 19 three times with a remainder 1.

Similarly,

7.2 % 3 will yield 1.2

6 % 2.5 will yield 1.0

The operands may be of *number* types.

7. Exponentiation operator **

The exponentiation operator ** performs exponentiation (power) calculation, *i.e.*, it returns the result of a number raised to a power (exponent). *For example*,

4 ** 3 evaluates to 64 (4^3)

a ** b ($a = 7, b = 4$) evaluates to 2401 (a^b *i.e.*, 7^4).

x ** 0.5 ($x = 49.0$) evaluates to 7.0. ($x^{0.5}$, *i.e.*, $\sqrt{x}$, *i.e.*, $\sqrt{49}$)

27.009 ** 0.3 evaluates to 2.68814413570761. ($27.009^{0.3}$)

The operands may be of number types.

3.2.3 Expressions

Expressions refer to a combination of variables/values and operators that generates another value. For example, adding the values of variables **AnvisPB**, **CharusPB**, **VasusPB** and **VanshsPB** and storing in variable **Total** can be written as the following expression :

```
Total = AnvisPB + CharusPB + VasusPB + VanshsPB
```

Make sure that the case of a variable name is not changed from its declaration as Python is case-sensitive.

Expressions

Expressions refer to a combination of variables/values and operators that generates another value.

A program uses **variables, operators, expressions** and **statements** to carry out tasks.

Example 3.2 *What will be the output produced by the following code ?*

```
A = 9.2
B = 2.0
C =  4
D = 21
print(A/4)
print(A//4)
print(B ** C)
print(D//B)
print(A % C)
```

Solution.

```
2.3
2.0
16.0
10.0
1.2
```

Aimed with this knowledge, let us now talk about these in Python's context and create some programs. But before that you should also know about *two* more things — **data types** and **statements**.

3.3 DATA TYPES

Data can be of many types *e.g., character, integer, real, string* etc. Anything enclosed in quotes represents **string data** in Python. Numbers without fractions represent **integer data**. Numbers with fractions represent **real data**.

3.3.1 Integers

Integers are whole numbers such as 5, 39, 1917, 0 etc. They have no fractional parts. Integers are represented in Python by numeric values with no decimal point. Integers can be positive or negative, *e.g.*, + 12, – 15, 3000 (missing + or – symbol means it is positive number).

3.3.2 Floating Point Numbers

A number having fractional part is a floating-point number. *For example*, 3.14159 is a floating-point number. The decimal point signals that it is a floating-point number, not an integer. The number 12 is an integer, but 12.0 is a floating-point number.

3.3.3 Strings

Strings are the values that hold characters in quotation marks. Both single and double[1] quotation marks can be used, *e.g.*, following are valid strings :

```
'Navin', "Kevin", "Gurjyot"
```

Strings can hold digits in quotes too, *e.g.*, following are also valid strings :

```
"File1", "R125", "63", "43.5"
```

Note

Anything enclosed in quotation marks whether a *number*, *alphabet* or a *symbol*, is a string value in Python.

Please note that :

"63" is a string while 63 is an *integer value*.

"43.5" is a string while 43.5 is a *floating point value*.

To type apostrophe in a string, use double quotes, *e.g.*,

```
"Sania's"
```

A single quote works as apostrophe inside a string of double quotes.

1. Python also allows triple-quotes ''' ... ''', but we are not covering that here as it is beyond the scope of the book.

But **input( )** has a property, which you must be aware of. The **input( )** function **always returns a value of string type**. Notice carefully in the above code – while displaying both values ***name*** and ***age***, Python has enclosed both the values in quotes *i.e.*, as '**Simar**' and '**16**'. This is because whatever value you enter through **input( )** function is treated as a string.

Now what are its consequences, especially when you are entering a number ? In order to understand this, consider the following code :

```
>>> age = input('Your age :')
Your age : 17
>>> age + 1
Traceback (most recent call last):
   File "<pyshell#7>", line 1, in <module>
       age + 1
TypeError: cannot concatenate 'str' and 'int' objects
>>>
```

*Notice the error raised when try to add a number to variable **age**.*

See, Python raised an error when you tried to add 1 to variable *age* whose value you entered as 17. The reason is obvious and clear — **Python cannot add on integer to a string**. Since variable **age** received value ***17*** through **input( )**, it actually had '***17***' in it *i.e.*, string value '***17***' ; thus you cannot add an integer to it.

Note

The input( ) always returns a value of string type.

Then what is the way out ? Don't worry. Python offers two functions **int( )** and **float( )** to be used with **input( )** to convert the values received through **input( )** into *integer* and *float* types.

Let us see how.

int( ) around input( ) function converts the received value in integer type

```
>>> age = int(input( "Your age :" ) )
Your age : 17
>>> age + 1
18
```

Now the age variable will store an integer type value

float( ) around input( ) function converts the received value in floating point type

```
>>> amount = float(input( "Amount Payable is :" ) )
Amount Payable is : 2300.50
>>> amount + 200
2500.5
>>>
```

Now the amount variable will store a floating-point type value

PART B

3.5 STATEMENTS

A statement refers to a command given to the computer.
For example, following are some statements in Python :

You can add comment in code by putting a # sign.

```
name = input("Enter your name")          # a statement that takes input
total = marks1 + marks2 + marks3         # a statement that computes
print(name)                              # a statement that prints
print(total)                             # a statement that prints
```

As you can see that statements given above have some text after a # sign. These are **comments**. Python ignores comments, *i.e.*, does not execute them. Comments are only for the reader, for explanation of the code.

Program 3.1 *Program to obtain three numbers and print their sum.*

```
# to input 3 numbers and print their sum

num1 = int(input("Enter number 1 :"))
num2 = int(input("Enter number 2 :"))
num3 = int(input("Enter number 3 :"))
```

While printing the values of variables, make sure that the variable names given without quotation marks, otherwise they will be treated as string values.

```
sum  = num1 + num2 + num3

print("Three numbers are :", num1, num2, num3)
print("Their sum is :", sum)
```

The output produced by above program is as shown below :

```
Enter number 1 : 15
Enter number 2 : 30
Enter number 3 : 25
Three numbers are :  15 30 25
Their sum is :  70
```

Program 3.2 *Program to obtain length and breadth of a rectangle and calculate its area.*

```
# to input length & breadth of a rectangle and calculate area

length = float(input("Enter length of the rectangle :") )
breadth = float(input("Enter breadth of the rectangle :") )
area = length * breadth

print("Rectangle Specifications")
print("Length =", length)
print("Breadth =", breadth)
print("Area =", area)
```

The output produced by above program is as shown below :

```
Enter length of the rectangle : 3.5
Enter breadth of the rectangle : 2.5
Rectangle specifications
Length = 3.5
Breadth = 2.5
Area = 8.75
```

LET US REVISE

- *Variables are the named places to hold some values in a program.*
- *The arithmetic operators are the symbols which depict some arithmetic calculation.*
- *Expressions refer to a combination of variables/values and operators that generates another value.*
- *The values/variables used along with an operator, are called operands.*
- *By default, the input( ) takes the input value as string type.*
- *You can use int( ) and float( ) with input( ) function to convert the input value to integer and floating point values respectively.*
- *A line following # sign till the end of the line becomes a comment in Python code. Python does not execute comments ; these are only for explanations.*

SESSION 4

Introduction to Lists

- Introducing Lists
- What are Python Lists ?

This section/content is not part of teacher-support PDF

4.3 WHAT ARE PYTHON LISTS ?

Python lists are sequence of values somewhat like list of activity 1*B* above. In other words, a Python list is simply a sequence of values stored in a specific order with each value identified by its index number in that order. The index numbers of Python lists start 0 onwards and go like 0, 1, 2, ...

Python List

A **Python list** is simply a sequence of values stored in a specific order with each value identified by its index number in that order.

4.3.1 How to Create Python Lists ?

To create Python lists, you can put a list of values with comma in between the items, enclosed in square brackets, *e.g.*,

```
>>> mylist = [2, 4, 6, 8, 10]
```

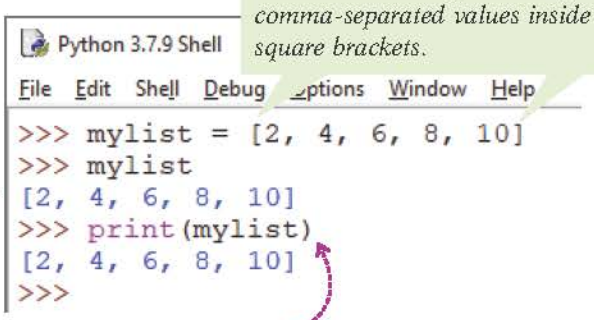
It will create a list with five numbers and name it as **mylist**.

You can display the list items by simply typing its name on the prompt in the interactive mode, *e.g.*, (*see* figure)

```
>>> mylist
[2, 4, 6, 8, 10]
```

You can also print the list by using the list name with **print()** (*see* figure)

Lists are created by enclosing comma-separated values inside square brackets.



Python lists can store any type of values in them – *integers, floating point-numbers, strings,* and even a combination of values, such as :

```
Python 3.7.9 Shell
File  Edit  Shell  Debug  Options  Window  Help
>>>
>>> mylist1 = [ 2, "Rudy", 93.5, "A"]
>>> mylist1
[2, 'Rudy', 93.5, 'A']
>>>
```

A Python list can store a combination of values of different types.

If you use the **type( )** with list name, Python will show its type as list :

```
Python 3.7.9 Shell
File  Edit  Shell  Debug  Options  Window  Help
>>>
>>> type(mylist)
<class 'list'>
>>>
```

The **type()** function when used with a list name

4.3.2 Checking the Length (Number of Items) of the List

The **length** or the **size** of a list means the number of items in the list. To check the length (number of items in the list), you can use the ***len( )*** function with list name as :

```
Python 3.7.9 Shell
File  Edit  Shell  Debug  Options  Window  Help
>>> mylist
[2, 4, 6, 8, 10]
>>> len(mylist)
5
>>> mylist1
[2, 'Rudy', 93.5, 'A']
>>> len(mylist1)
4
```

The **len()** function when used with a list name gives its length , *i.e.*, no. of items in the list

Refer figure 1

Note

The **len(<listname>)** function returns the size or length of the list, *i.e.*, the number of elements in the list.

4.3.3 Accessing Individual Items of a List

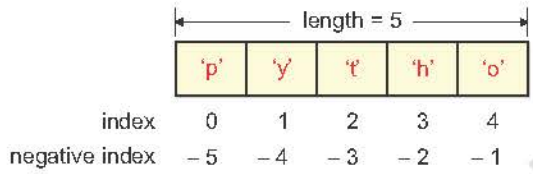
Python list's items are internally numbered as 0, 1, 2, 3...(Positive Indexes). That is,

the 1st item has the index number as 0
the 2nd item has the index number as 1
the 3rd item has the index number as 2
:
The last item has index number as length-1 or size-1

Python sequences such as lists have positive indexes as well as the negative indexes. That is,

```
the last item has the index number as -1
the 2nd last item has the index number as -2
the 3rd last item has the index number as -3
:
The first item has index number as -length or -size)
```

Graphically, it can be depicted as :



Positive index for the first item in the list	= 0
Positive index for last item of a list	= size –1
Negative index for the first item in the list	= –size
Negative index for last item of a list	= –1

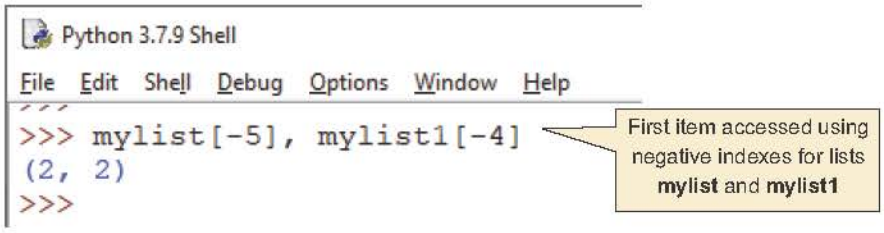
To access an element, just give its index (positive or negative, any) in the square brackets along with the list name, *i.e.*, as :

```
<Listname>[<index of element>]
```

For example, consider our lists **mylist** and **mylist1** as created above :

```
Python 3.7.9 Shell
File Edit Shell Debug Options Window Help
>>> mylist
[2, 4, 6, 8, 10]
>>> mylist1
[2, 'Rudy', 93.5, 'A']
>>>
```

You may also use the negative index for the first item. For **mylist**, the negative index for the first item is **–5** while for **mylist1**, the negative index for the first item is **–4** as the lengths of **mylist** and **mylist1** respectively are 5 and 4.



4.3.4 Accessing all Items of a List

To access all items of a list one by one, you can use a **for** statement of Python, which picks individual items from a sequence and lets you use them as you want. It can be used as :

```
for <picking name> in <listname>:
    using <picking name> as individual item
```

For example, consider the list **mylist** containing item as [2, 4, 6, 8, 10]. If you use **for** statement with it to print individual items, you need to write :

```
for oneitem in mylist :
    print (oneitem)
```

It will print the items of the mentioned list (mylist) one by one as :

```
2
4
6
8
10
```

4.3.5 Modifying an Item in the List

To modify an item in a list, just assign the new value to its accessing name, *i.e.*,

```
<listname>[<index>] = <new value>
```

For example, to change the 3rd item of mylist1 to 95.5, you may write :

```
mylist1 [2] = 95.5
```

```
>>> mylist1
[2, 'Rudy', 93.5, 'A']
>>> mylist1[2] = 95.5
>>> mylist1
[2, 'Rudy', 95.5, 'A']
```

Note

Changing the value of an item in a list is the same as changing the value of a variable, *i.e.*, assign a new value to item's accessing name.

4.3.6 Adding an Item in the List

To add an item to a list, use **append(<item>)** as :

```
<listname>.append(<newitem>)
```

For example, to add element 12 to list **mylist [2, 4, 6, 8, 10]**, you may write :

```
mylist.append(12)
```

```
>>> mylist
[2, 4, 6, 8, 10]
>>> mylist.append(12)
>>> mylist
[2, 4, 6, 8, 10, 12]
```

LET US REVISE

- *A Python list is simply a sequence of values stored in a specific order with each value identified by its index number in that order.*
- *The* ***len(<listname>)*** *function returns the size or length of the list , i.e., the number of elements in the list.*
- *Positive index for the first item in the list is* ***0****.*
- *Positive index for last item of a list is* ***size–1****.*
- *Negative index for the first item in the list is* ***–size****.*
- *Negative index for last item of a list is –1.*
- *Using a for loop all items from a list can be accessed one by one.*
- *Using append(), a new item is added to a list.*