# What is Artificial Intelligence? (Real Definition)

Artificial Intelligence (AI) is the field of computer science that studies and builds machines capable of performing tasks that require intelligence when done by humans, by perceiving, reasoning, learning, and acting autonomously in complex environments.

| AI is **NOT**: | AI **IS**: |
|---|---|
| • Just automation<br>• Just programming rules<br>• Just neural networks<br>• Just ChatGPT or robots | • **Decision-making under uncertainty**<br>• **Learning from data or experience**<br>• **Generalization to unseen situations**<br>• **Goal-oriented behavior**<br>• **Reasoning and inference** |

## What Makes a System "Intelligent"?

1. **Perception**

   Understand the world (images, speech, text, signals)

2. **Representation**

   Store knowledge in a usable internal form

3. **Reasoning**

   Draw conclusions, plan steps, solve problems

4. **Learning**

   Improve performance with experience

5. **Decision Making**

   Choose actions to maximize reward or minimize cost

6. **Adaptation**

   Handle unseen situations

A calculator does none of these so it's not AI.

A chess engine with search + evaluation is AI but its week.

A self-driving car is an advance type of applied AI.

# What is Machine Learning?

### Learning patterns from data instead of explicit rules

Subtypes:

- Supervised Learning (classification, regression)
- Unsupervised Learning (clustering, dimensionality reduction)
- Semi-supervised Learning
- Self-supervised Learning

Examples:

- Spam detection
- Price prediction
- Face recognition

# What is Deep Learning?

### Neural networks with many layers

Deep learning is a type of machine learning using multi-layered artificial neural networks, inspired by the human brain, to learn complex patterns from vast amounts of data, enabling tasks like image recognition, natural language processing, and self-driving cars with minimal human intervention by automatically extracting features through its deep structure.

Used when:

- Data is large
- Patterns are complex

Subfields:

- CNNs → vision
- RNNs / LSTMs → sequences
- Transformers → language, vision, audio

Examples:

- ChatGPT
- Image generation
- Speech recognition

# What is Natural Language Processing?

### Understanding and generating human language

Natural Language Processing (NLP) is an AI field where computers learn to understand, interpret, and generate human language (text and speech) by combining computer science, linguistics, and machine learning, enabling applications like chatbots, translation, and sentiment analysis by processing unstructured text into structured data. It bridges the gap between human communication and computer understanding, allowing machines to perform tasks like recognizing voice commands, summarizing documents, and detecting spam.

Tasks:

- Text classification
- Translation
- Question answering
- Summarization
- Chatbots

# What is Computer Vision?

### Making machines “see”

Computer Vision (CV) in AI enables machines to "see," interpret, and understand visual data (images, videos) like humans do, using techniques like machine learning, deep learning (especially CNNs), and pattern recognition to identify objects, extract info, and make decisions from visual inputs.

Tasks:

- Object detection
- Face recognition
- Medical imaging
- Autonomous driving

# What is Reinforcement Learning?

### Learning by interaction and reward

Reinforcement Learning (RL) in AI is a trial-and-error learning method where an "agent" learns to make optimal decisions by interacting with an environment, receiving rewards for good actions and penalties for bad ones, aiming to maximize cumulative rewards over time.

Core idea:

Agent + Environment + Actions + Rewards

Used in:

- Robotics
- Games (AlphaGo)
- Autonomous systems

# What is Knowledge Representation and Reasoning?

### Logic-based intelligence

Knowledge Representation and Reasoning (KRR) is a core AI field focused on structuring information about the world so computers can use it to solve complex problems, involving both **representing knowledge** (using formalisms like ontologies, graphs, logic) and **reasoning** (drawing conclusions, making inferences, solving problems from that knowledge)

Includes:

- Rules
- Ontologies
- Expert systems
- Inference engines

Used in:

- Medical diagnosis
- Legal reasoning
- Planning systems

# What is Robotics and Embodied AI?

### AI acting in the physical world

Robotics integrates AI into physical systems, allowing them to perceive, reason, and act in the real world, with Embodied AI specifically focusing on this fusion where AI learns through direct physical interaction, moving beyond digital-only realms to understand and manipulate environments, making robots more adaptive and human-like in their capabilities. This field combines machine learning, computer vision, and NLP to create intelligent agents (robots) that can learn by doing, collaborate with humans, and handle complex, unstructured tasks.

Includes:

- Sensors
- Motion planning
- Control systems
- Perception + RL

# What is AI Ethics and Safety?

### Ensuring AI is safe, fair, explainable, and controllable

AI ethics involves moral principles guiding responsible AI development and use (fairness, privacy, transparency), while AI safety focuses on technical measures to prevent accidents, misuse, and unintended consequences, ensuring AI reliably does what's intended, with both aiming to build beneficial, trustworthy systems that benefit humanity and mitigate risks from bias to existential threats.

Includes:

- Bias mitigation
- Interpretability
- Alignment
- Governance

# Python Execution Flow

## How Python runs line by line?

Source Code (.py)

↓

Lexing & Parsing

↓

Abstract Syntax Tree (AST)

↓

Bytecode (.pyc)

↓

Python Virtual Machine (PVM)

↓

Execution (instruction by instruction)

### Step 1: Lexing & Parsing (Before Any Code Runs)

Python first **reads the entire file** and checks:

- Syntax correctness
- Indentation rules
- Structure (blocks, loops, functions)

### Step 2: Abstract Syntax Tree (AST)

- Python converts your code into a tree structure.

x = 10 + 20

Becomes something like:

Assign

├— Name(x)

└— Add

├— Constant (10)

└— Constant (20)

### Step 3: Bytecode Compilation

Python converts AST → bytecode

### Step 4: Python Virtual Machine (PVM)

Now comes the actual execution.

The PVM:

- Reads bytecode
- Executes one instruction at a time
- Uses a stack-based architecture

Python does NOT execute lines

It executes bytecode instructions

## Execution Flow:

1. Python enters file
2. Executes top-level statements from top to bottom
3. Each statement may contain multiple bytecode instructions
4. Once a statement finish → moves to next

    - **Top to bottom**
    - But **instruction by instruction**, not text lines

# AI Thinking Practice

## Task:

Imagine a simple AI system that predict whether a student will pass or fail.

**Q: What is the Data?**

Ans: All relevant data against student

- Student ID
- Student Name
- Total marks
- Obtain marks
- Passing and failing criteria
- Previous Grades or marks in previous classes

**Q: What are the features?**

Ans: Features are values (usually numbers) derived from data. The *process* is called feature engineering. Features are what the model actually learns from.

- Student ID to track his previous result or grades from data.
- Student to find the exact student
- Total marks and obtain marks to get percentage and average of marks to make predictions.
- Previous results are important to make prediction and that will make lot easier to get average of their grades to check pass or fail.

**Q: What could be the Output?**

Ans: Output depend on the input and previous data in the data base, if results and average satisfy the criteria of passing then student is passed otherwise, he is failed.

**Q: Where does python help?**

Ans: Python can help to take input of current data if needed or it can get data from database and we can create functions in python that can get previous data of student, and after performing certain function we can predict from average that the student will pass or not.

**Data:**

- Student academic records (raw)

**Features:**

- Percentage
- Attendance %
- Previous average score
- Improvement trend

**Output (Label):**

- Pass (1) / Fail (0)

**Python's Role:**

- Load data
- Clean data
- Create features
- Train model
- Predict outcome

# THE UNIVERSAL AI PIPELINE

This pipeline is **more important than any algorithm**.

Problem

↓

Data

↓

Cleaning

↓

Features

↓

Model

↓

Prediction

↓

Evaluation

↓

Improvement

## STEP 1: PROBLEM

Before data, before Python, before ML:

**Ask:**

- What decision do I want the machine to make?
- Is this **classification**, **regression**, or something else?

**Example:**

"Predict whether a student will pass or fail"

This tells you:

- Output type → **binary classification**
- Evaluation → accuracy, precision, recall

If the problem is unclear → AI will fail.

## STEP 2: DATA (RAW REALITY)

Data answers:

"What information do I have about the problem?"

**Types of Data:**

- Structured → tables (CSV, Excel, DB)
- Unstructured → text, images, audio

Key idea:

Data is NOT learning. Data is just evidence.

Bad data → bad AI
Even the best model can't fix bad data.

## STEP 3: DATA CLEANING (THE MOST TIME-CONSUMING STEP)

Real data is:

- Missing
- Noisy
- Inconsistent

Cleaning includes:

- Removing duplicates
- Handling missing values
- Fixing wrong entries
- Normalizing formats

**Industry truth**:

70% of AI work is data cleaning.

No cleaning → model learns garbage.

## STEP 4: FEATURES (THE BRAIN FOOD)

This is where **intelligence starts**.

**Remember:**

- Data = raw facts
- Features = **useful signals**

Good features:

- Capture patterns
- Are numerical
- Relate directly to the outcome

Bad features:

- IDs
- Random text
- Constant values

**Key insight**:

Models don't understand meaning — only patterns in numbers.

So feature engineering = translating reality into numbers.

## STEP 5: MODEL (JUST A TOOL)

A model is: A mathematical function that maps features → output

Important truth:

- Model does NOT "think"
- Model does NOT "understand"
- Model only finds patterns

Examples:

- Linear Regression → straight-line relationships
- Decision Tree → rule-based splits
- Neural Network → layered transformations

Beginners over-focus here.
Experts don't.

## STEP 6: PREDICTION (OUTPUT)

Prediction means:

- Using trained model on unseen data

Examples:

- Pass / Fail
- Price = 120,000
- Spam / Not spam

Important concept:

Prediction ≠ truth
It's a **probability-based guess**.

## STEP 7: EVALUATION (REALITY CHECK)

Ask:

- How good is the model?
- Is it better than random guessing?
- Is it biased?

Metrics depend on problem:

- Classification → accuracy, precision, recall
- Regression → MAE, RMSE

If you don't evaluate, you're **not doing AI**.

## STEP 8: IMPROVEMENT (THE LOOP)

AI is **iterative**, not linear.

You improve by:

- Better data
- Better features
- Better cleaning
- Sometimes better models

**Important mindset**:

If model is bad → fix data first, not model.