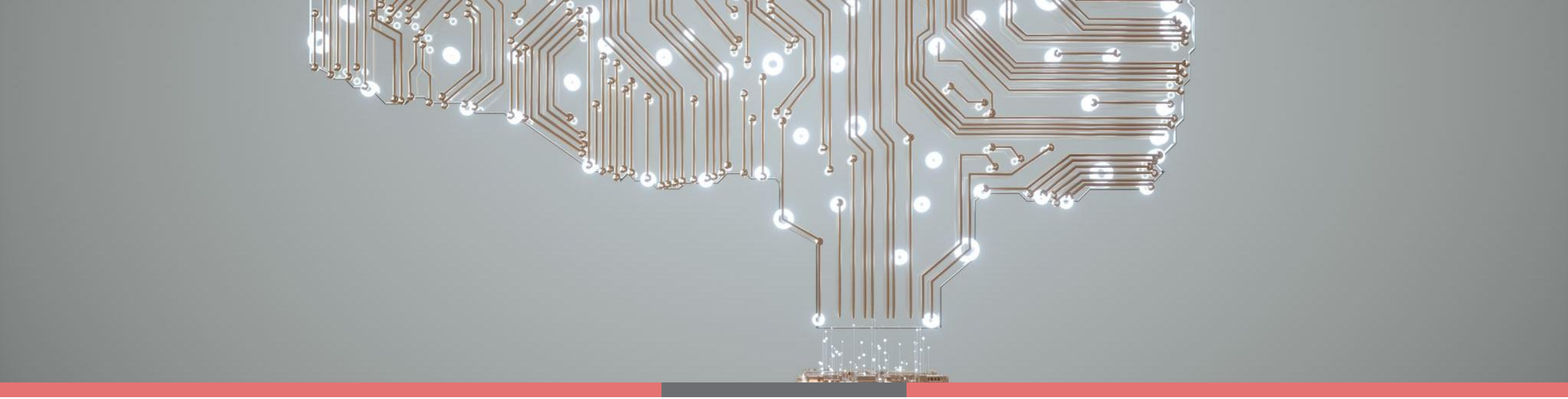


Artificial Intelligence

Mohamad Elnomrossie

April - 2025

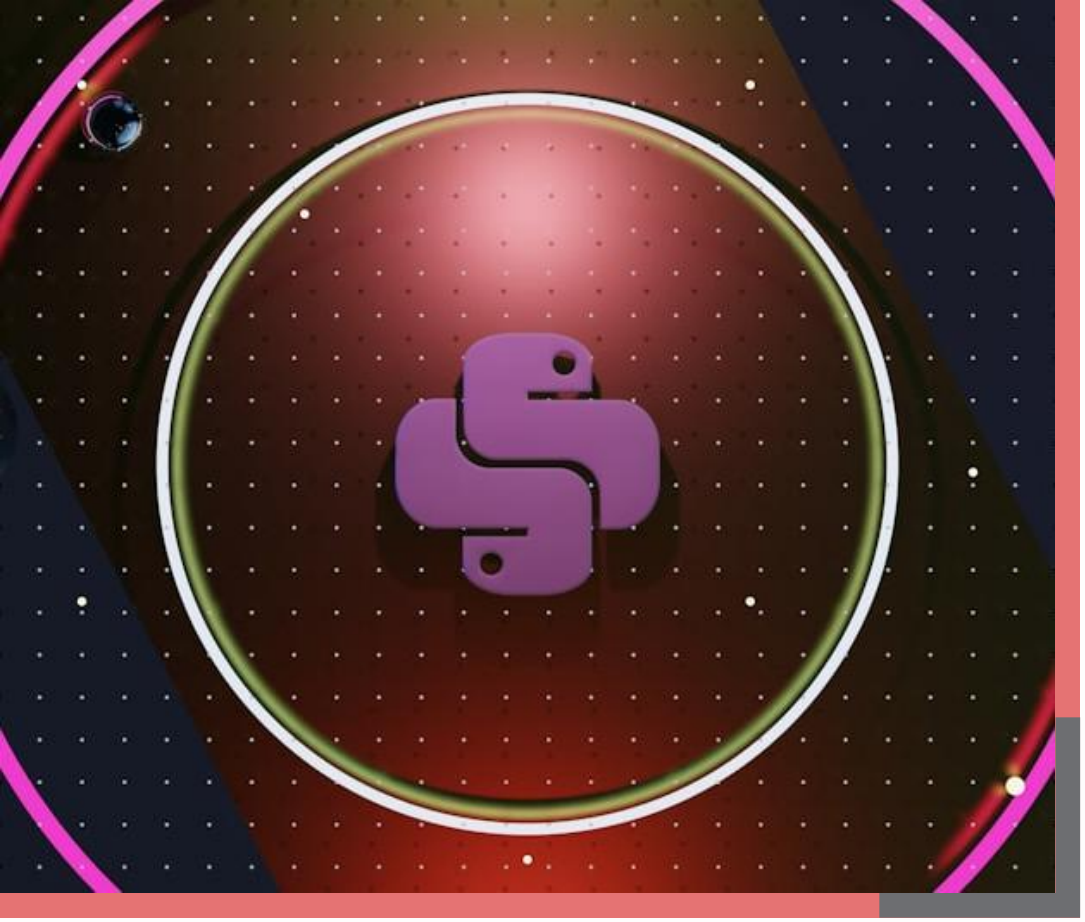
Agenda

- AI Overview
- Python in AI
- Machine Learning Basics
- Deep Learning Explained
- AI Applications

AI Overview

UNDERSTANDING AI

AI is the simulation of human intelligence in machines designed to think and act like humans. It encompasses problem-solving, speech recognition, learning, and planning.



Python in AI

WHY PYTHON?

Python offers simplicity, readability, and a rich ecosystem of libraries, making it ideal for AI development.

KEY LIBRARIES

Popular libraries include TensorFlow, PyTorch, and Scikit-learn, which provide powerful tools for machine learning and neural networks.

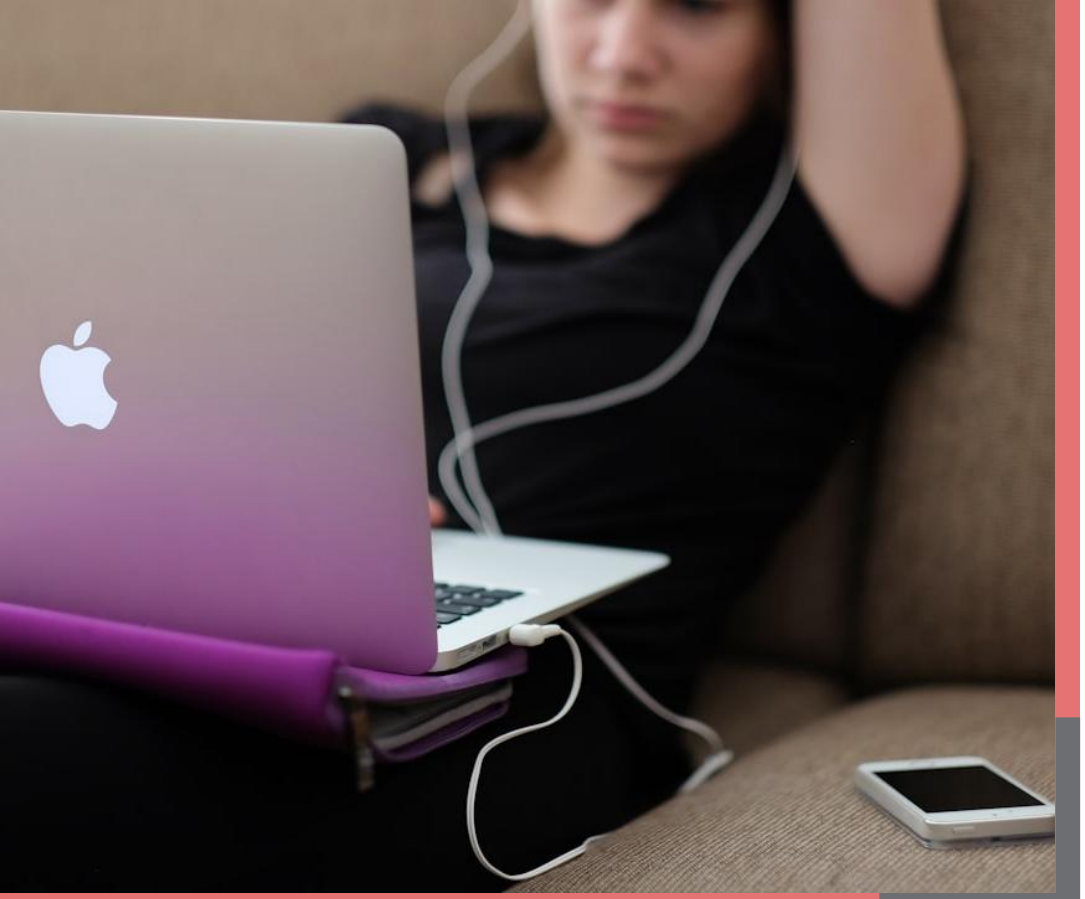
Machine Learning

BASICS OF ML

Machine Learning is a subset of AI that enables systems to learn from data and improve their performance without being explicitly programmed.

TYPES OF LEARNING

Supervised, unsupervised, and reinforcement learning are the three main types, each serving different purposes in data analysis.



Deep Learning

WHAT IS DEEP LEARNING?

Deep Learning is a specialized form of Machine Learning using neural networks with many layers, allowing for complex pattern recognition.

SIGNIFICANCE

It drives advancements in fields like computer vision and natural language processing, powering applications such as voice assistants and image recognition.

AI Applications

HEALTHCARE

AI aids in diagnosis, treatment recommendations, and personalized medicine.

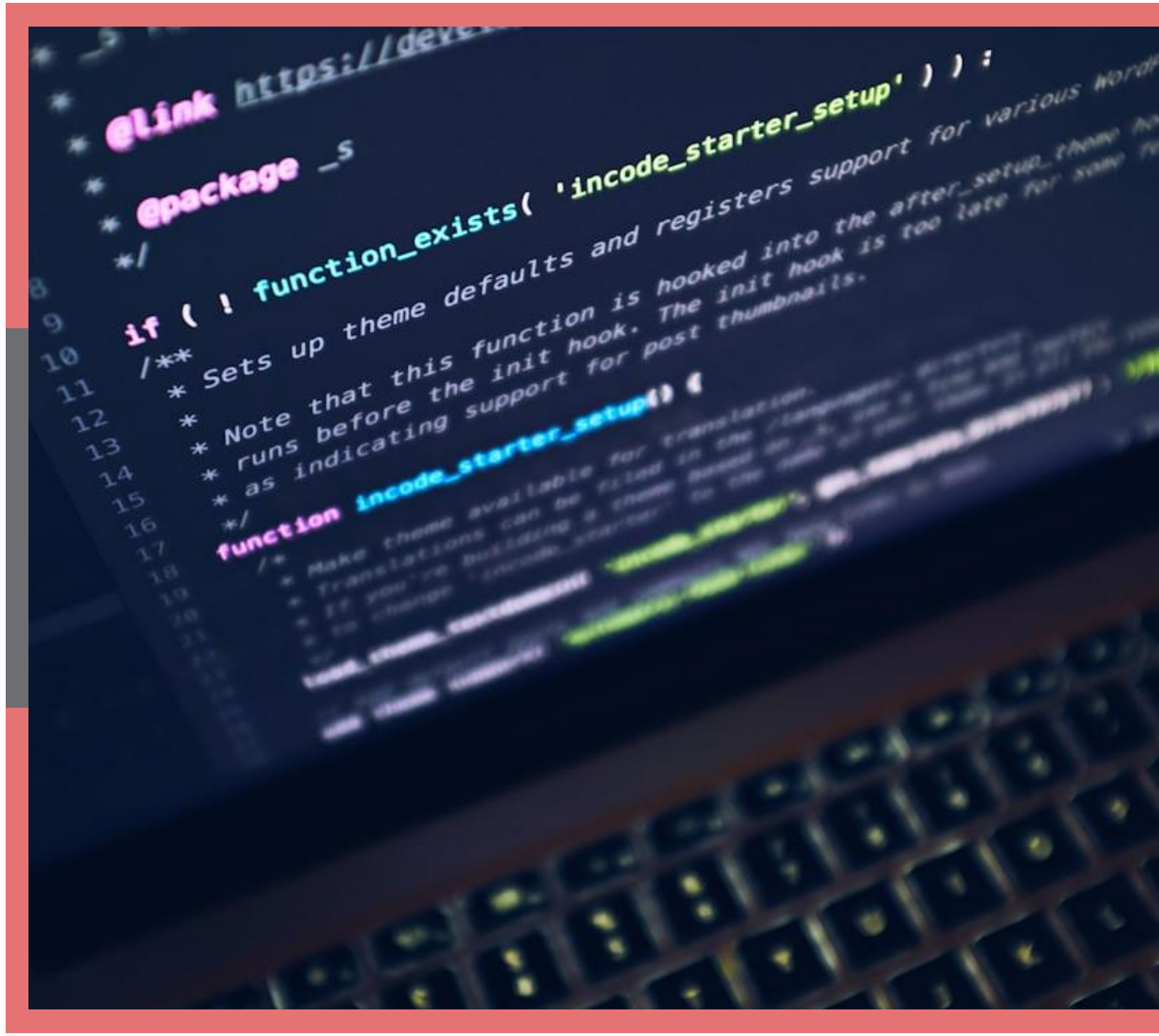
FINANCE

Used for fraud detection, risk assessment, and automated trading.

TRANSPORTATION

Enables autonomous vehicles and optimizes logistics and supply chains.

Challenges in AI

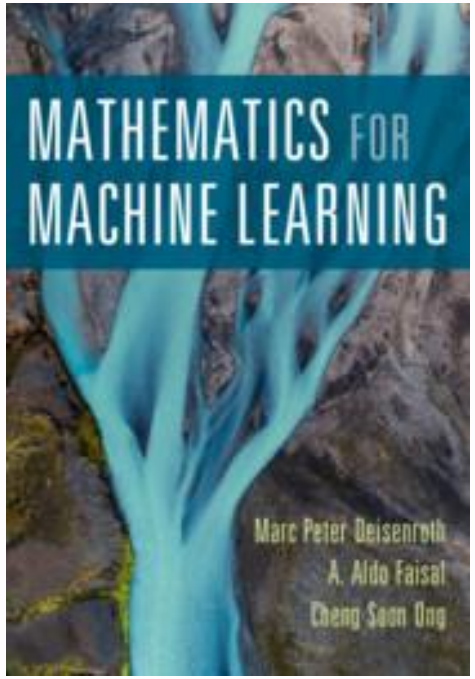


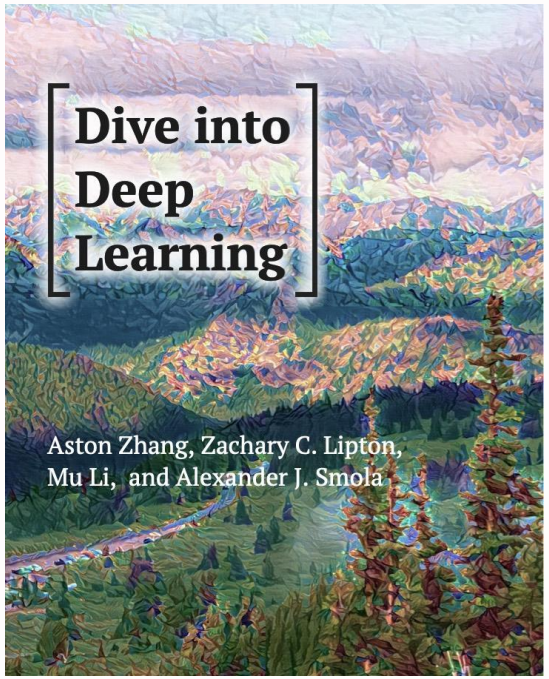
KEY CHALLENGES

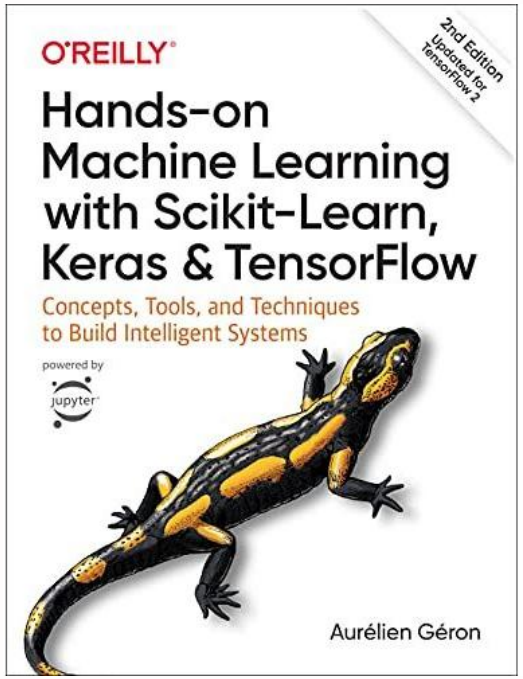
Ethical concerns, data privacy, and algorithm bias pose significant challenges to AI development and deployment. Addressing these is critical for responsible AI use.

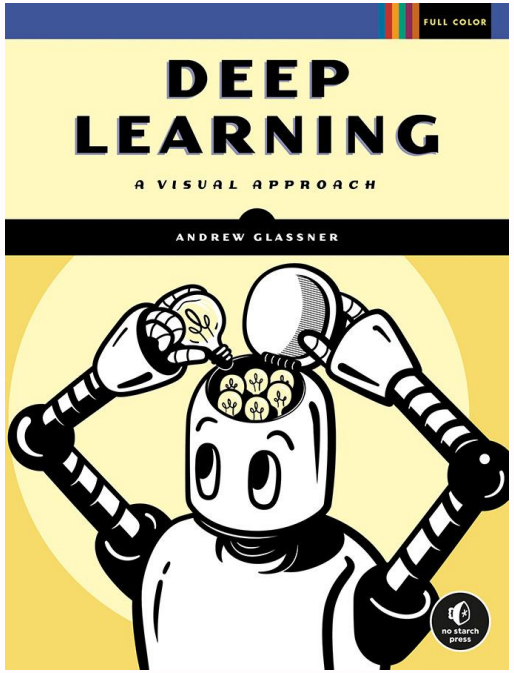
Course outline
Introduction to AI
Module 2: Programming Fundamentals
Module 3: Mathematics for AI
Module 4: Data Analysis
Module 5: Introduction to Machine Learning
Module 6: Deep Learning
Module 7: Computer Vision
Module 8: Natural Language Processing
Module 9: Capstone Project

Coursebook


MATHEMATICS FOR MACHINE LEARNING
Marc Peter Deisenroth
A. Aldo Faisal
Cheng Soon Ong


Dive into Deep Learning
Aston Zhang, Zachary C. Lipton,
Mu Li, and Alexander J. Smola


O'REILLY
2nd Edition
Updated for TensorFlow 2
Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow
Concepts, Tools, and Techniques to Build Intelligent Systems
powered by
jupyter
Aurélien Géron


FULL COLOR
DEEP LEARNING
A VISUAL APPROACH
ANDREW GLASSNER
no starch press

kaggle

Hugging Face

DAGsHub

GitHub

git

Google colab

jupyter

Github repo

Prerequisites

➔ Strong hold on mathematics ⇒ Linear Algebra, Calculus and Statistics.

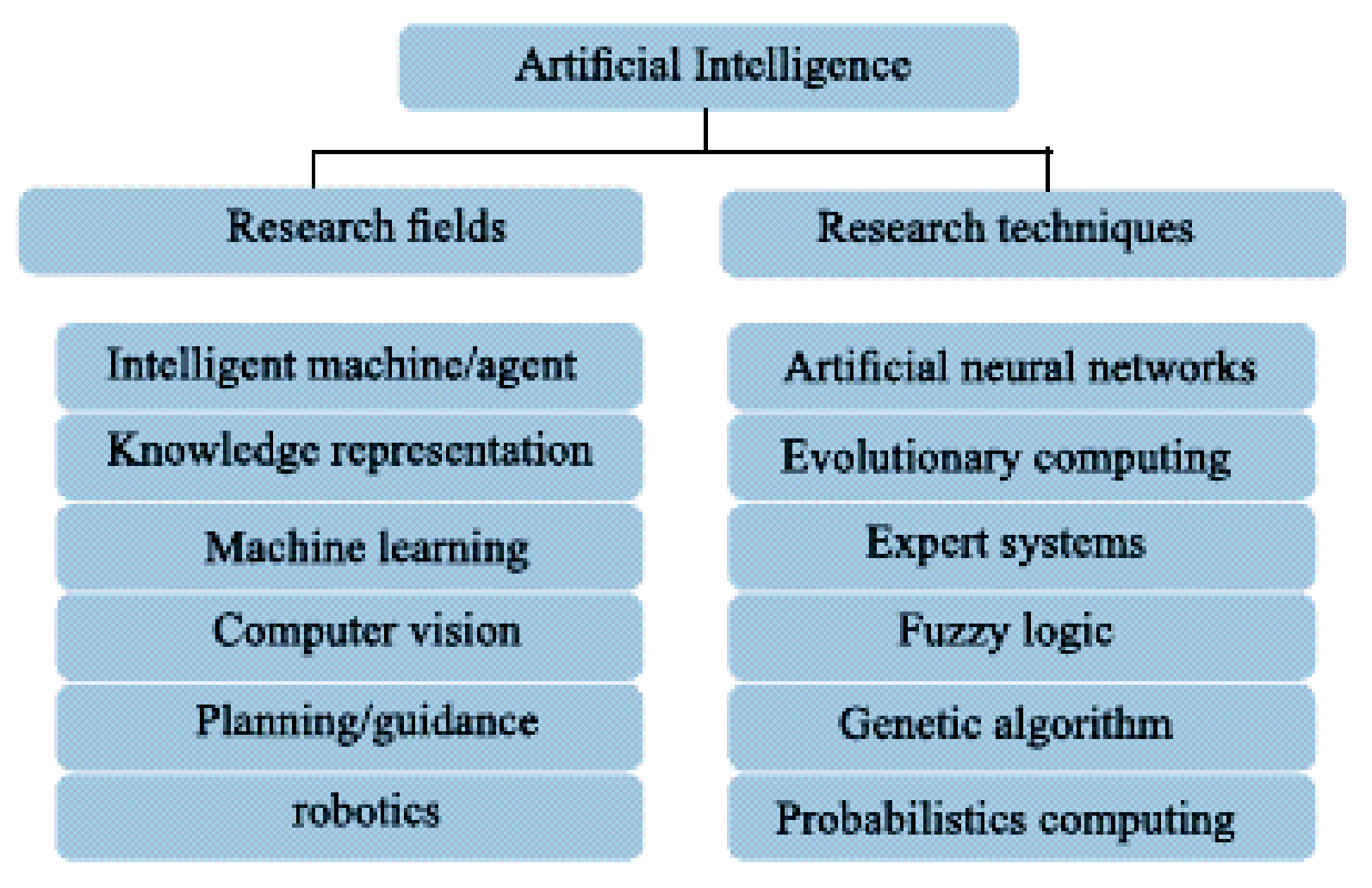
➔ Experience with programming languages as Python and/or C++.

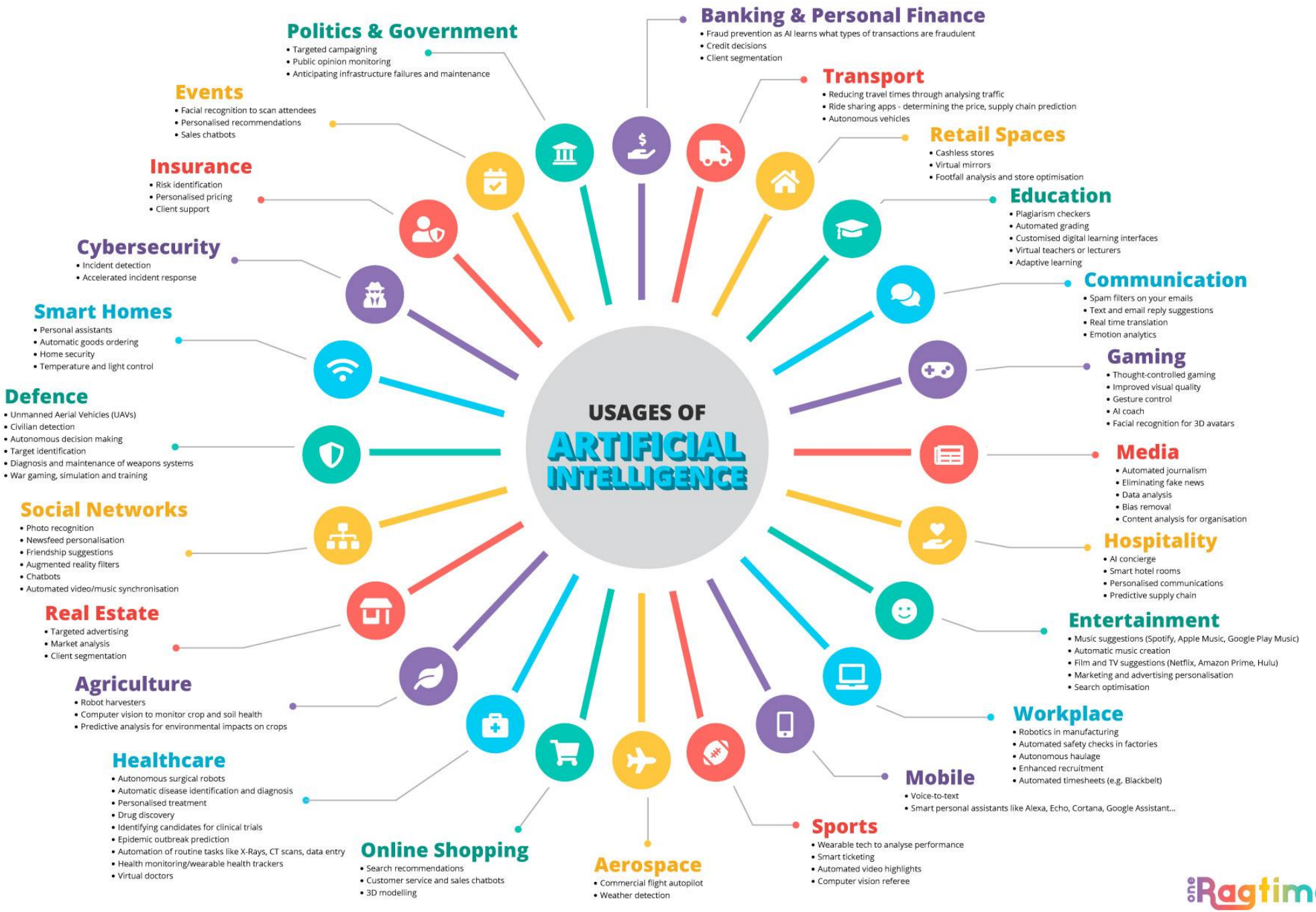
➔ A good knowledge in understanding and creating algorithms.

➔ Technical reading and writing which is essential in any technology. (self learning)

Artificial Intelligence vs. Machine Learning vs. Deep Learning?

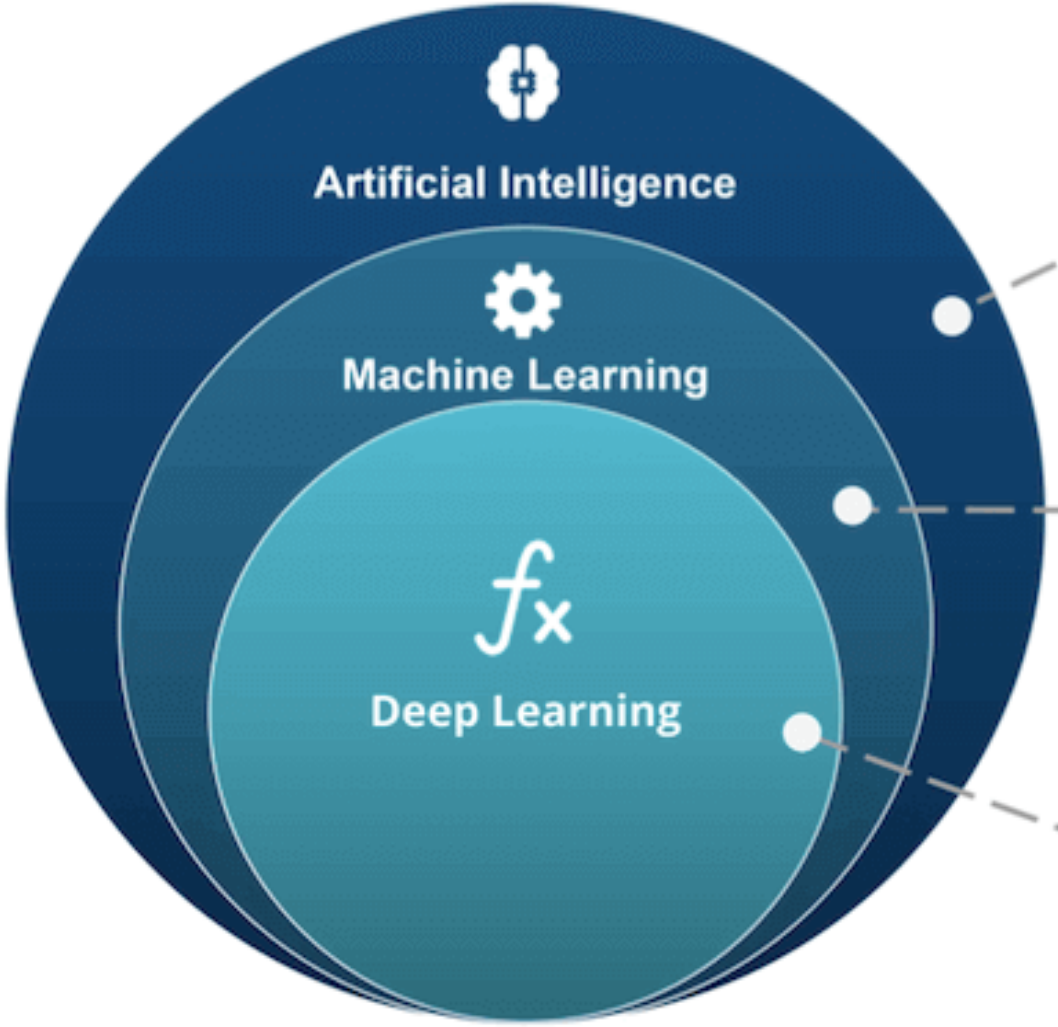
Artificial intelligence (AI) is technology that enables computers and machines to simulate human learning, comprehension, problem solving, decision making, creativity and autonomy.




USAGES OF
ARTIFICIAL INTELLIGENCE
Banking & Personal Finance
• Fraud prevention as AI learns what types of transactions are fraudulent
• Credit decisions
• Client segmentation
Transport
• Reducing travel times through analysing traffic
• Ride sharing apps - determining the price, supply chain prediction
• Autonomous vehicles
Retail Spaces
• Cashless stores
• Virtual mirrors
• Footfall analysis and store optimisation
Education
• Plagiarism checkers
• Automated grading
• Customised digital learning interfaces
• Virtual teachers or lecturers
• Adaptive learning
Communication
• Spam filters on your emails
• Text and email reply suggestions
• Real time translation
• Emotion analytics
Gaming
• Thought-controlled gaming
• Improved visual quality
• Gesture control
• AI coach
• Facial recognition for 3D avatars
Media
• Automated journalism
• Eliminating fake news
• Data analysis
• Bias removal
• Content analysis for organisation
Hospitality
• AI concierge
• Smart hotel rooms
• Personalised communications
• Predictive supply chain
Entertainment
• Music suggestions (Spotify, Apple Music, Google Play Music)
• Automatic music creation
• Film and TV suggestions (Netflix, Amazon Prime, Hulu)
• Marketing and advertising personalisation
• Search optimisation
Workplace
• Robotics in manufacturing
• Automated safety checks in factories
• Autonomous haulage
• Enhanced recruitment
• Automated timesheets (e.g. Blackbelt)
Mobile
• Voice-to-text
• Smart personal assistants like Alexa, Echo, Cortana, Google Assistant...
Sports
• Wearable tech to analyse performance
• Smart ticketing
• Automated video highlights
• Computer vision referee
Aerospace
• Commercial flight autopilot
• Weather detection
Online Shopping
• Search recommendations
• Customer service and sales chatbots
• 3D modelling
Healthcare
• Autonomous surgical robots
• Automatic disease identification and diagnosis
• Personalised treatment
• Drug discovery
• Identifying candidates for clinical trials
• Epidemic outbreak prediction
• Automation of routine tasks like X-Rays, CT scans, data entry
• Health monitoring/wearable health trackers
• Virtual doctors
Agriculture
• Robot harvesters
• Computer vision to monitor crop and soil health
• Predictive analysis for environmental impacts on crops
Real Estate
• Targeted advertising
• Market analysis
• Client segmentation
Social Networks
• Photo recognition
• Newsfeed personalisation
• Friendship suggestions
• Augmented reality filters
• Chatbots
• Automated video/music synchronisation
Defence
• Unmanned Aerial Vehicles (UAVs)
• Civilian detection
• Autonomous decision making
• Target identification
• Diagnosis and maintenance of weapons systems
• War gaming, simulation and training
Smart Homes
• Personal assistants
• Automatic goods ordering
• Home security
• Temperature and light control
Cybersecurity
• Incident detection
• Accelerated incident response
Insurance
• Risk identification
• Personalised pricing
• Client support
Events
• Facial recognition to scan attendees
• Personalised recommendations
• Sales chatbots
Politics & Government
• Targeted campaigning
• Public opinion monitoring
• Anticipating infrastructure failures and maintenance

Artificial Intelligence vs. Machine Learning vs. Deep Learning?

AI vs ML vs DL



ARTIFICIAL INTELLIGENCE
A technique which enables machines to mimic human behaviour

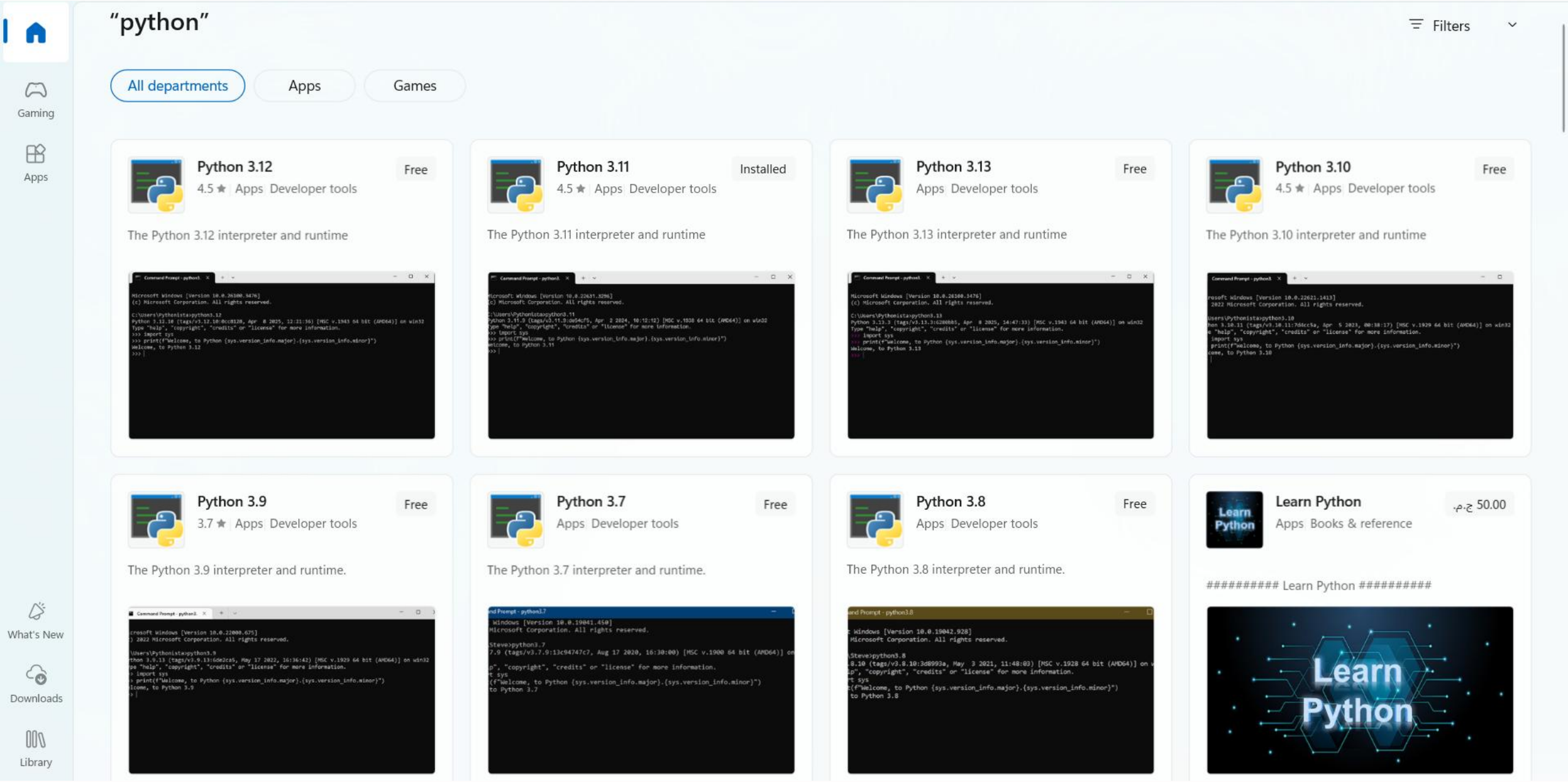
MACHINE LEARNING
Subset of AI technique which use statistical methods to enable machines to improve with experience

DEEP LEARNING
Subset of ML which make the computation of multi-layer neural network feasible

Tools setup

1. Git
2. Python
3. Vs code
4. Jupyter Notebook/ Anaconda

python™


"python"
Filters
Gaming
Apps
All departments
Apps
Games
Python 3.12
Free
4.5 Apps Developer tools
The Python 3.12 interpreter and runtime
Python 3.11
Installed
4.5 Apps Developer tools
The Python 3.11 interpreter and runtime
Python 3.13
Free
Apps Developer tools
The Python 3.13 interpreter and runtime
Python 3.10
Free
4.5 Apps Developer tools
The Python 3.10 interpreter and runtime
Python 3.9
Free
3.7 Apps Developer tools
The Python 3.9 interpreter and runtime.
Python 3.7
Free
Apps Developer tools
The Python 3.7 interpreter and runtime.
Python 3.8
Free
Apps Developer tools
The Python 3.8 interpreter and runtime.
Learn Python
50.00
Apps Books & reference
########## Learn Python ##########
Learn Python
What's New
Downloads
Library

Google

python

All Images Videos Books News Maps Flights More Tools

Python.org
https://www.python.org

Welcome to Python.org

Functions Defined · Get Started. Whether you're new to programming or an experienced developer, it's easy to learn and use **Python**. · Download. **Python** source ...

Downloads
Download the latest version of Python · Licenses. All Python ...

Documentation
>>> Python Needs You. Open source software is made better ...

Python Releases for Windows
Stable Releases · Download Windows installer (64-bit ...

Python For Beginners
Learning. Before getting started, you may want to find out which ...

About
Python is developed under an OSI-approved open source license ...

More results from python.org »

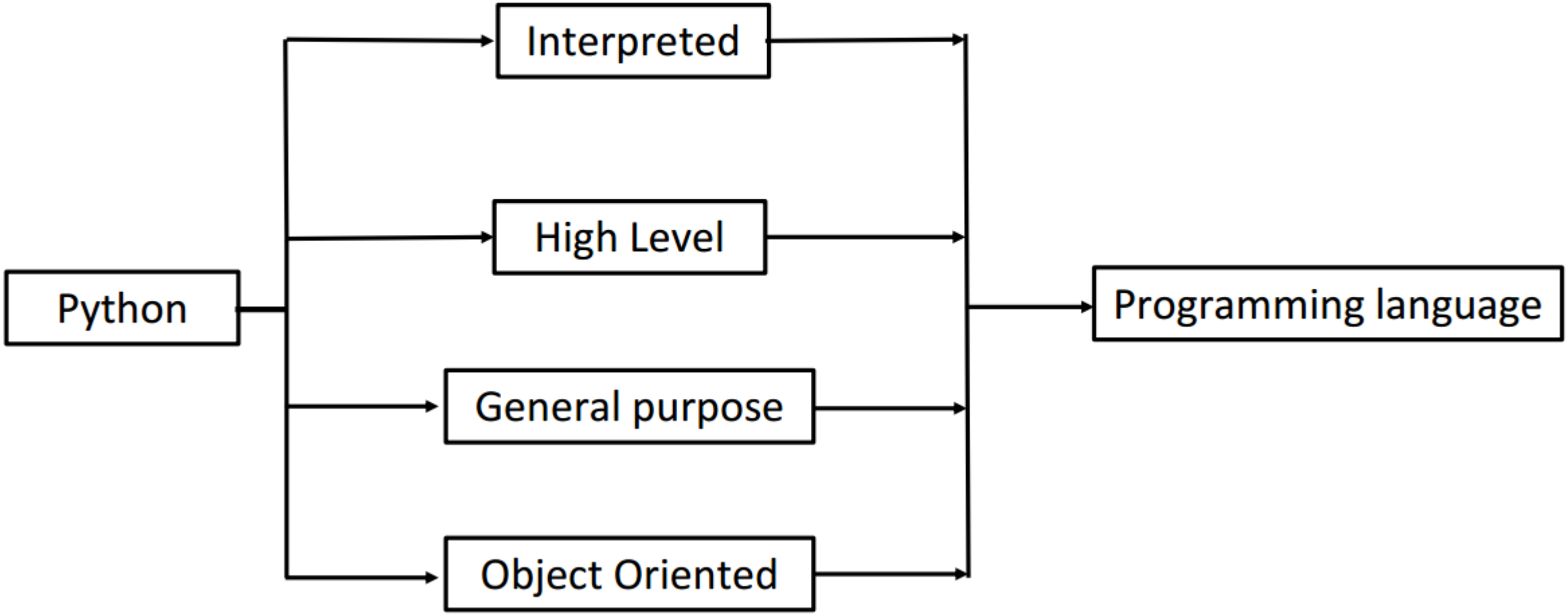
Python
High-level programming language

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically type-checked and garbage-collected. Wikipedia ›

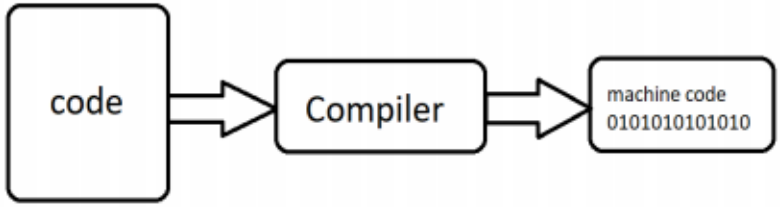
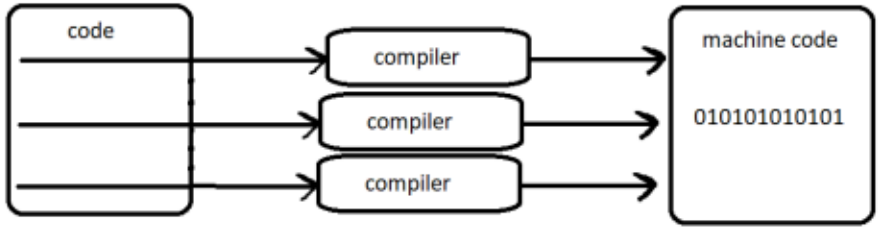
Designed by: Guido van Rossum

Developer: Python Software Foundation

What is Python ?



Types Of Programming languages

	Compiled programming language	Interpreted programming languages
Compilation Process	Code is translated into machine code or an intermediate code by a compiler before execution. 	Code is translated and executed line by line by an interpreter during runtime. 
Execution Speed	Generally faster execution as the entire code is translated into machine code beforehand.	Generally slower compared to compiled languages.
Debugging	Errors are detected during the compilation process, making debugging more challenging.	Errors are identified during runtime, making it easier to pinpoint and fix issues.
Memory Usage	more efficient memory usage	May result in less optimized memory usage compared to compiled languages
Example	C , C++ , C#	**Python** , Ruby

What is Python ?

Programming languages Levels		
	Low Level Language	**High level language**
Definition	- These languages are like talking directly to the computer's hardware. - They're a basic set of instructions that the computer easily understands.	- programming language that's easier for humans to understand. - They use words and structures that resemble everyday language, making programming more user-friendly.
Example	assembly languages and machine code.	**Python**, Java, C++, Ruby, Swift

Programming and Python Basics

Print() Function

The print() function in Python is a **built-in function** that displays information on the screen.

What is functions

```
In [1]: print("Hello World")
```

Compiler

```
Hello World
```

OUTPUT

```
print("my age = ",30)
```

Compiler

```
my age =  30
```

OUTPUT

Types of functions

1. Built-in (print, abs, sum…)
2. User defined

Print() Function

The print() function in Python is a **built-in function** that displays information on the screen.

What is functions

```
In [1]: print("Hello World")
```

Compiler

```
Hello World
```

OUTPUT

```
print("my age = ",30)
```

Compiler

```
my age =  30
```

OUTPUT

Built-in functions

A
abs()
aiter()
all()
anext()
any()
ascii()
B
bin()
bool()
breakpoint()
bytearray()
bytes()
C
callable()
chr()
classmethod()
compile()
complex()
D
delattr()
dict()
dir()
divmod()

E
enumerate()
eval()
exec()
F
filter()
float()
format()
frozenset()
G
getattr()
globals()
H
hasattr()
hash()
help()
hex()
I
id()
input()
int()
isinstance()
issubclass()
iter()

L
len()
list()
locals()
M
map()
max()
memoryview()
min()
N
next()
O
object()
oct()
open()
ord()
P
pow()
print()
property()

R
range()
repr()
reversed()
round()
S
set()
setattr()
slice()
sorted()
staticmethod()
str()
sum()
super()
T
tuple()
type()
V
vars()
Z
zip()
_
__import__()

Python Variables

a = 5 → a=> Variable Name , 5 => value stored in Variable → RAM Memory (5, a)

```
a = 5
print(a)
```

5 → OUTPUT

a = 6 → a=> Variable Name , 6 => value stored in Variable → RAM Memory (~~5~~ Old a, 6 New a)

```
a = 5
a = 6
print(a)
```

→ **overwrite**

6 → OUTPUT

PYTHON IDENTIFIERS

1. Python identifiers are user-defined names for variables, functions, classes, or objects in code.

2. Guidelines for creating identifiers include using letters (uppercase and lowercase), numbers, and underscores.

3. Special characters and operators are not allowed in identifiers.

4. Identifiers should not begin with a number, and certain keywords are reserved and cannot be used as standalone identifiers.

5. Meaningful names for identifiers are encouraged.

6. **Python is case-sensitive,** distinguishing between uppercase and lowercase identifiers.

7. Avoid using 'l', 'I', or 'O' as single-character variable names due to potential font-related confusion.

Class names	Variable names / Methods / Functions / Arguments / Globals	Constants
PascalCase	snake_case	FULLY CAPITALIZED

Python Data Types			
One variable : One Data			
Integer	**Float**	**String**	**Boolean**
whole number without a decimal point.	numeric data type that represents real numbers and can include a decimal point.	sequence of characters, enclosed within single or double quotes, used to represent text data.	binary data type representing either True or False
One variable : Many Data			
List	**Tuple**	**Set**	**Dictionary**
1. Data ordered 2. Changeable Data 3. Allow Duplicate Data 4. List can be represented by [ ] 5. Can be nested among all 6. Convert any datatype to list using List() function	1. Data ordered 2. Unchangeable Data 3. Allow Duplicate Data 4. Tuple can be represented by ( ) 5. Can be nested among all 6. Convert any datatype to tuple using tuple() function	1. Data unordered 2. Unchangeable Data 3. Not allow Duplicate Data 4. Tuple can be represented by { } 5. Can be nested among all 6. Convert any datatype to set using set() function	1. Data ordered 2. Changeable Data 3. Not allow Duplicate for keys 4. Dictionary can be represented by { } 5. Can be nested among all 6. Convert any datatype to dictionary using dict() function