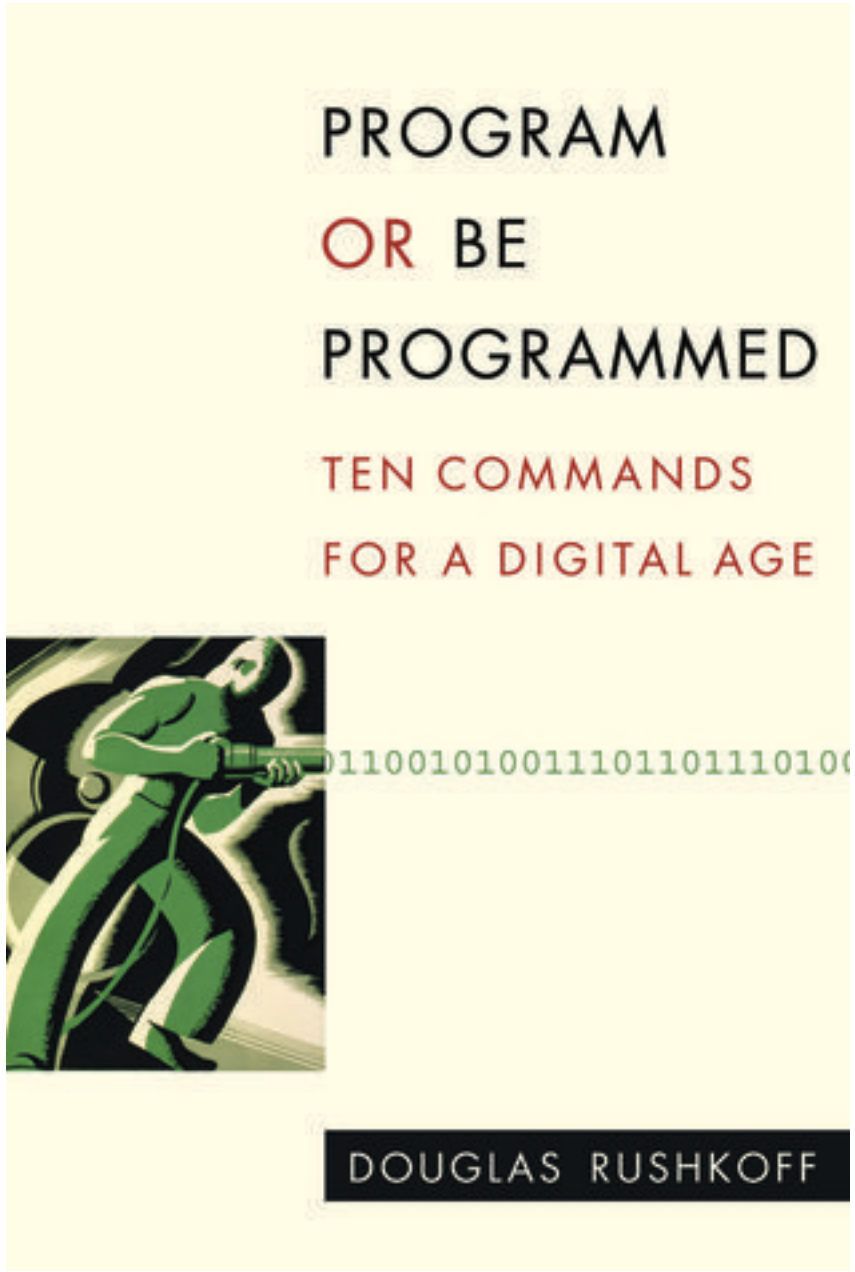
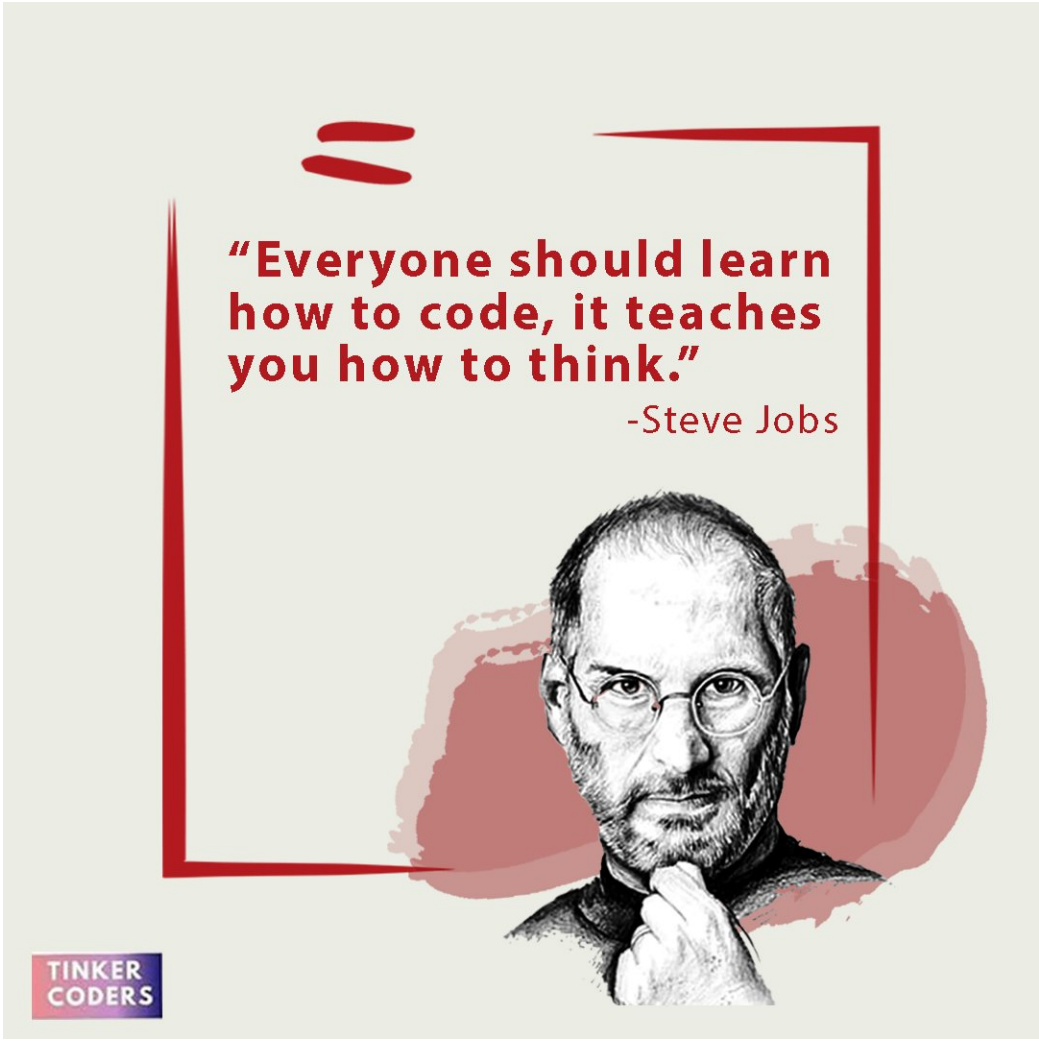
CM3-Computer Science


PROGRAM
OR BE
PROGRAMMED
TEN COMMANDS
FOR A DIGITAL AGE
DOUGLAS RUSHKOFF


"Everyone should learn how to code, it teaches you how to think."
-Steve Jobs
TINKER CODERS

Why are we here?

- To prepare you for the CM-3 challenge
- To be able to make mental models of systems
- To make sure that, as a manager, or a founder, you have no trouble understanding tech
- To program at a beginners level. Why?
- To learn to be unusual, innovative, to learn to be the odd duck
- To critically think and be self-critical
- TO HAVE FUN!

Module Topics:

1. IT Concepts & Terminology
2. IT Infrastructure
3. Applications & Software
4. Software Development Concepts
5. Database fundamentals
6. Scenario Building: Futurists
7. Virtual Team Working
8. Problem identification
9. Regulation, Ethics & Law with new technologies
10. Select Optimal Solution
11. Introduction to Python Programming (terminologies, applications)
12. Past Industrial Revolutions
13. Emerging Technologies and the Hype Cycle
14. Measuring Results
15. Setting up machines
16. Coding with Datacamp
17. Stakeholder Theory
18. Defining Pain Points
19. The Business Economy, Future Economy, Gig Economy and Data Economy
20. Python Control Flow
21. Social Issues as Tech Drivers and Eco-System Pain Points
22. Variables, data types and operators
23. Regulation, Ethics & Law with new technologies
24. Legal Instruments & Disrupted Industries
25. Complex 4d. Systems
26. Using code kata to code variables
27. Blockchain & Impacts
28. Financial Instruments and Issues
29. Concept & Skeleton for prototyping
30. Conditional Statements and Looping Constructs
31. Data Structures
32. String Manipulations and Functions
33. The Nature of Money;
34. Financial Markets: Movement of Funds Through the Economy, Selling securities and
Financial Institutions;
35. Introduction to the Financial System;
36. Banking, Investment Banking, Insurance, Pensions and Collective Investments;
37. Cash-flow Analysis for Investment Decision Making;
38. Prepare for a Future Career
39. Building a targeted brand and the power of the network.
40. The Future of work and virtual recruitment.
41. Privacy in the modern age

Module Learning Outcomes:

On completion of this module, students are expected to be able to:

MLO1: Identify and experiment with ways in which emerging technologies can disrupt, or equally improve and transform existing business and create new ones.

MLO2: Research how the shift to a knowledge and research economy fueled by Big Data and emerging tech will change existing job descriptions, create entirely new jobs and transform the nature of work itself.

MLO3: Learn and apply digital skills such as coding and app development that are becoming increasingly important in the jobs of today and tomorrow.

Our team have been contracted by the board of directors of your chosen company in your chosen industry to carry out a large-scale audit of the organization operating in the chosen industry, with a view towards future proofing their organization. Your team may choose any public limited company with over $5bn of revenue operating in the list of five industries provided as follows:

1. Air travel
2. Healthcare
3. Logistics
4. Recruitment
5. Retail

FUTURE PROOFING CHALLENGE

Here is the brief you have received from the board of directors.
In an *era marked by rapid technological advancements and shifting global dynamics*, many industries and firms stand at a critical juncture. The imperative to future-proof is not merely a strategic choice but a necessity for survival and success in an increasingly competitive landscape. *Firms that fail to adapt to emerging trends and technologies risk falling behind, facing not only diminished market share but also challenges in maintaining operational efficiency and customer satisfaction.* This stagnation can lead to increased operational costs,
inability to meet evolving customer expectations, and vulnerability to disruptive market forces. Conversely, firms that embrace the future-proofing ethos position themselves to reap substantial benefits. *By leveraging cutting-edge technologies such as artificial intelligence, big data analytics, and sustainable business practices, these forward-thinking companies can significantly enhance operational efficiency, reduce environmental impact, and provide exceptional customer experiences. The integration of advanced digital tools can streamline operations, from personalized customer interactions to optimized supply chain management planning and predictive operational efficiencies, leading to cost savings and increased profitability.*

3. Customer Experience Enhancement:

3.1: Journey Mapping: Produce a map the typical customer journey touchpoints, analysing pain points and opportunities for improvements. This should include the customer journey from acquisition to post-product delivery.

3.2: Experience Enhancement: Create a custom app will improve customer experience. This app should focus on at least one aspect of the customer journey.
This section of the report should be accompanied by an App prototype.

4.2: Emerging Technology Research: Develop a strategy for incorporating artificial intelligence into the business. Conduct research to identify at least two other emerging technologies in the industry.

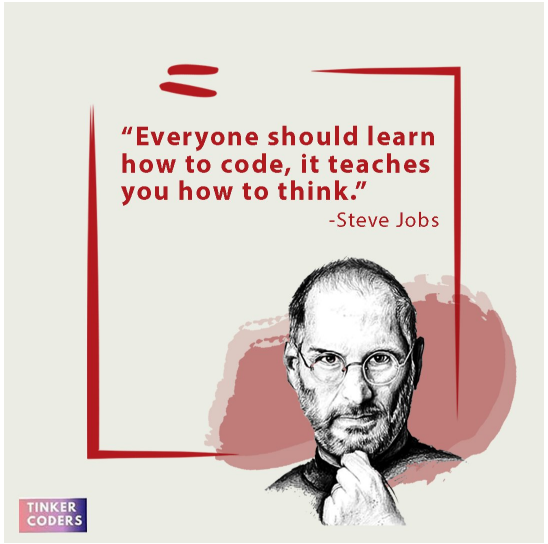
4.3: Innovation Roadmap Development: Develop a roadmap for integrating these trends, artificial intelligence, and emerging technologies into the firm's strategic plan, ensuring ongoing relevance and competitiveness.

6. Customer Retention and Loyalty Programs:

6.1: Program Assessment: Evaluate the effectiveness of current loyalty programs, understanding member usage, satisfaction, and areas for improvement.

6.2: Competitive Benchmarking: Benchmark against leading loyalty programs in the industry to identify best practices and innovative retention and loyalty strategies.

6.3: Enhancement Strategies: Propose strategies to revamp the loyalty program, focusing on more personalized rewards, seamless redemption processes, and strategic partnerships for expanded benefits



LETS HAVE FUN

You will have to work hard, but I promise you that in return, you will get really good. And that you will have a lot of fun!

Who am I? I am Rahul Dave.

- Originally from India
- Grew up thinking computers were somewhere “there”…
- Did an extremely computational Ph.D. in Physics
- Have since coded hardware for telescopes, databases for NASA, machine learning and AI models, and just about anything else.
- Created two companies with partial success.
- Been a lecturer at Harvard in Computer Science, Data Science, and Stats.
- Have taught machine learning and AI to librarians and “suits”.
- Like to ski and rock climb, though my 8 year old Anamika is getting better at it already!

Structure of a session

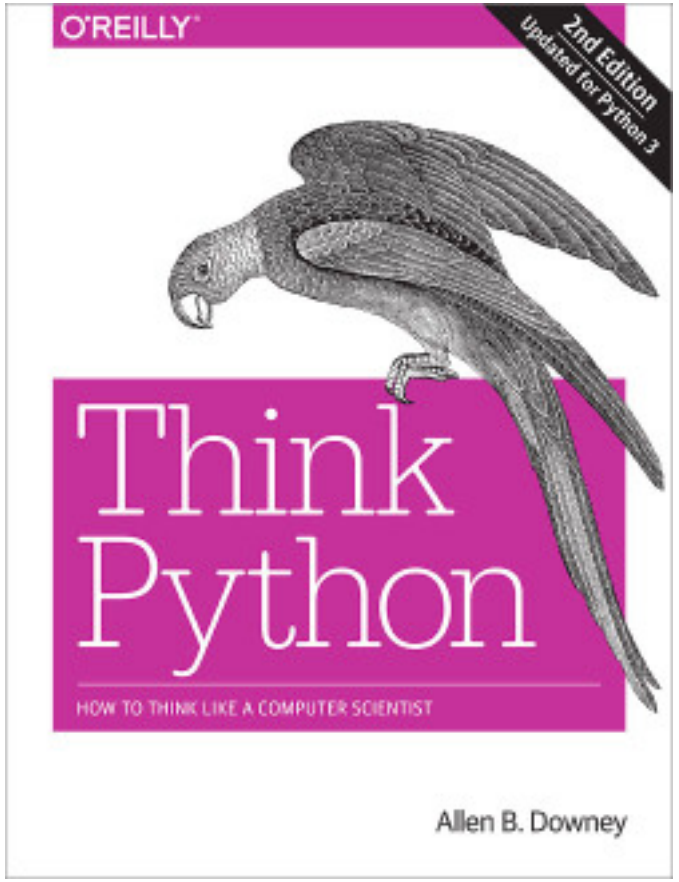
- we’ll start by reviewing any readings or work assigned
- we’ll them move onto the days topic
- each session will cover a major idea in computer science
- as we get further in the sessions, these ideas will combine
- we will do a bit of python programming related to the idea
- we may do a bit of low-code work related to the idea
- we’ll see how this idea is used in the world today
- we’ll cover the history of the idea
- we’ll touch on the business aspects of the idea and see how it created and destroyed startups and companies
- there might be an in-class or post-class quiz. There will be an assigned reading.

Our Sessions

Lecture	Big Idea
1. What is the essence of a tree?	Approach a problem at the right level of abstraction
2. Functions can travel first class too.	Idempotent functions and Functional Programming
3. Do not change things to make them faster.	Unchangedness and Immutability of data
4. Your state of mind is not my state of mind.	Modeling the state of objects with Object-Oriented Programming
5. If it looks and quacks like a duck, it is a duck.	Polymorphism and well defined interfaces.
6. If they cant eat bread, let them eat trees.	How the choice of the right data structure leads to performant algorithms.
7. Any problem can be solved by adding one extra layer of indirection	Separate the specification of an algorithm from its implementation
8. You may not use the bathroom unless I give you the key!	Concurrency and Parallelism in programs
9. Floating down a lazy river.	Lazy Evaluation and Streams
10. Deforming Mona Lisa with a new tongue.	Languages for stratified design
11. It’s turtles all the way down!	Languages and the machines which run them
12. Feed me Seymour, and I’ll remember you!	GPUs enable the learning instead of the writing of programs
13. Hey Markov! Whats the next word?	Large Language Models generate the next token.

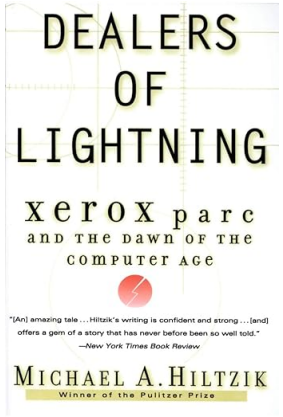

1. What is the essance of a tree?
3. the right lev of tsesm
2. Aporgechee things to
7 rrat lse or dase? . .
5 Opcideeren ing ttablam to
5. raing these prasitlol dostcation
8 Idempront fuctcries
8 Utchamange is fo hat
7, perrantide prifacre.
8. Your robs chamg sres
7 nenict-odfinrd alliaiscions.
9 Mowi the bdsts an of icking
1. m abking my ecia preferattions.
1. Oolting the calsk lo foschiing
7 be ceiing sazngle eres duk.
1 vikk a porfiast eade.
2. Appombent funcions
5. Urchaggdantr troke is nott
5 make ct dinrellulaticle. .
3. Dlempoftnet futrees
4 Uneaianef fincions
8 Polanynlom
5 nif its me tate l of ducks
8. Plyocinste a a duck
1 Umphonom and docks
1 and ir a dees
9 Untaichoct is 'ana olind
7 fak rats orfijten duck.
10 Parcflamgle or bucs.
7 lfs the alsg, opiueles.
10. Hny problemn can solved
6 anckking lefia lirfactes. .

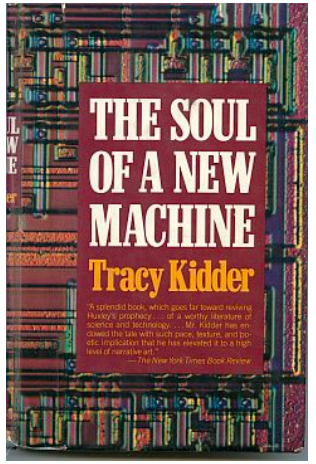
Books

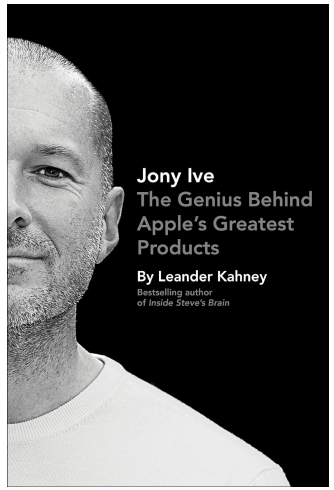


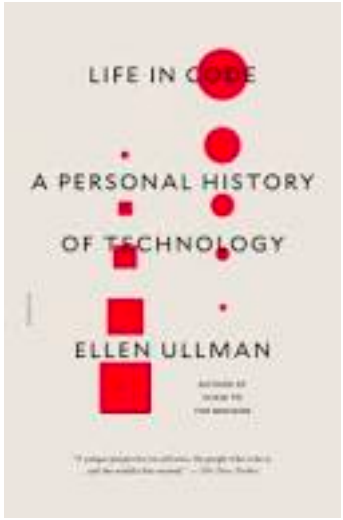
FREE. YOU NEED THIS.
https://greenteapress.com/wp/think-python-2e/

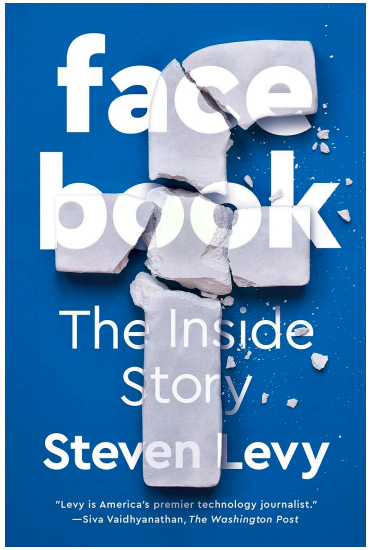
Books Fun to Read











Where will you encounter Computer Science?

IT is any technology that teams within an organization use to get their work done.

That includes hardware, software, and computer systems — as well as the infrastructure (networks, data storage, memory, etc.) needed to maintain those systems.

Perhaps in IT?

Forward-thinking organizations view IT as a competitive advantage that allows them to work smarter and achieve their business goals.

Types of IT Applications

Application development

Enterprise Resource Planning (ERP)

Customer Relationship Management (CRM)

Product management

Marketing

Human resources management systems (HRMS)

Budgeting, legal, and accounting tools

Infrastructure and network administration

Infrastructure IT typically includes:

- *Equipment* such as computers, printers, and routers.
- *Telecommunications* such as internet, IP addressing, and phone services.
- *Operating systems* that manage how computers store files, run applications, and maintain security.
- *Data centers* for data storage, backup, and recovery.

User support and productivity

IT teams support internal and external customers and troubleshoot issues. They also help facilitate cross-functional collaboration and employee productivity.

Technology includes:

•Communication and meeting tools to support teamwork, including web-conferencing tools for employees who work remotely.
•*Process automation, Machine Learning (ML), and Artificial Intelligence (AI) tools to improve productivity and reduce manual processes.*
•Collaboration tools for document sharing and real-time collaboration.
•Project management tools to organize work, document progress, and manage tasks.

Security and compliance

As an organization's digital footprint grows, so does its attack surface. Security and compliance teams protect business and customer data, while ensuring that the organization complies with security regulations. Security systems and tools include:

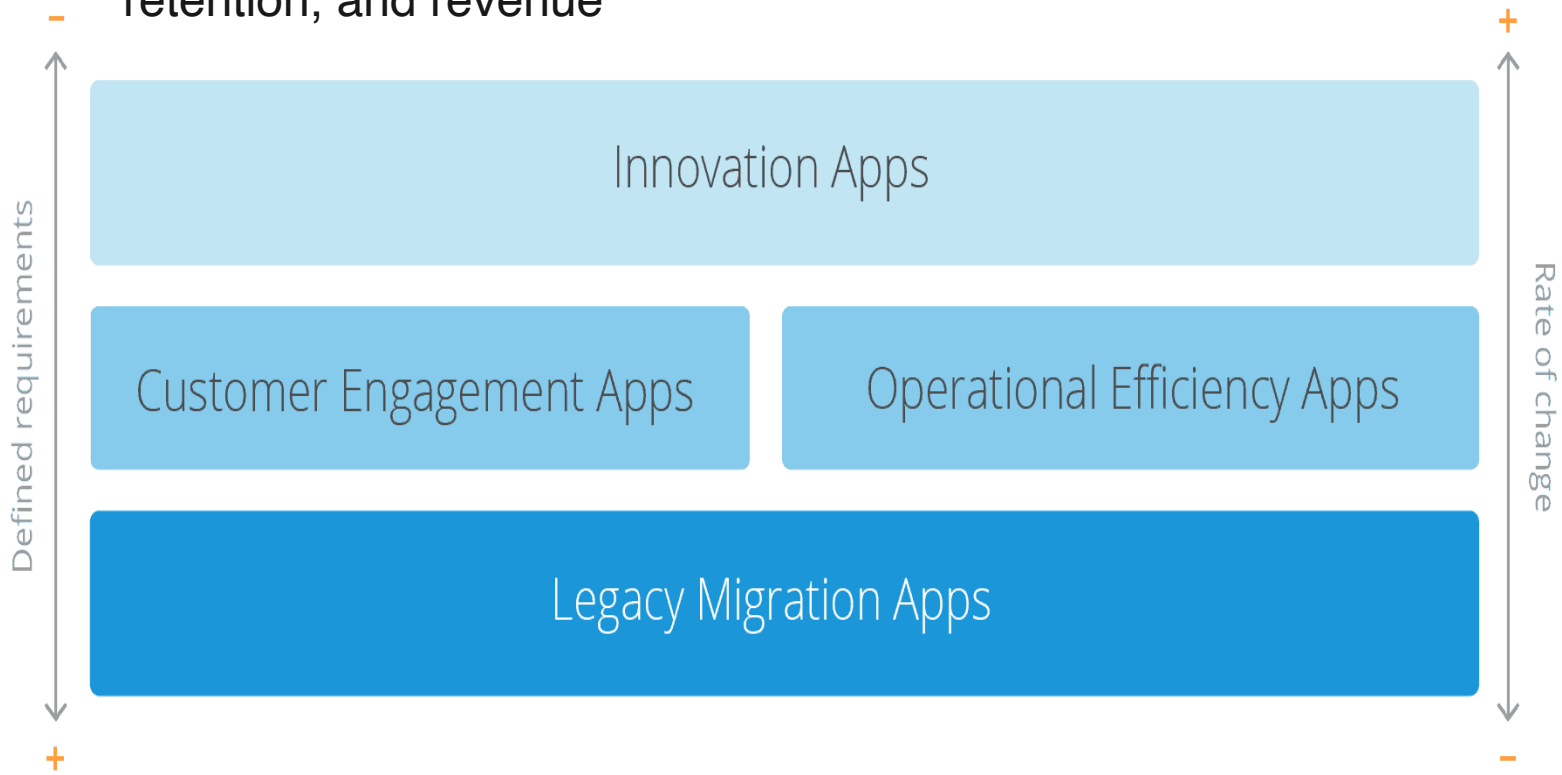
•*Documentation and storage software* to comply with privacy laws and safeguard sensitive information.
•*Data loss prevention software* to monitor the flow of data.
•*Antivirus software* to protect the organization from viruses and other damaging malware.

Or perhaps you will work for a tech company?

Innovation apps stem new digital business models, products, and channels and help grow and differentiate the organization. Often, they leverage emerging technologies like IoT, AI, and Machine Learning to unlock new sources of value.

Customer engagement apps enable customers and partners to interact or transact with the business. Customer engagement apps improve satisfaction, retention, and revenue

Operational efficiency apps are employee- or partner-facing apps designed to lower costs by reducing or automating manual or paper-based processes. Operational efficiency apps may support departmental, cross-departmental, or company-wide processes and are often driven by compliance needs (i.e., avoiding cost penalties), particularly in regulated industries.



Most of the **legacy modernization** projects we have seen within our customer base are business-driven transformation initiatives. That is, rather than a pure lift-and-shift of existing functionality, these new apps are meant to replace legacy apps that can't support new processes or provide the right user experience. As such, they require new functionality but should also support current processes.

Or perhaps you will become a tech startup founder?

Computer Science is what enables IT, applications and productivity.

And has been revolutionizing the world: the internet, the web, social media, AI…

And even if you have a tech co-founder, you will need to understand all these concepts.

Don’t worry. We’ll take it one step at a time.

CM-3 Computer Science

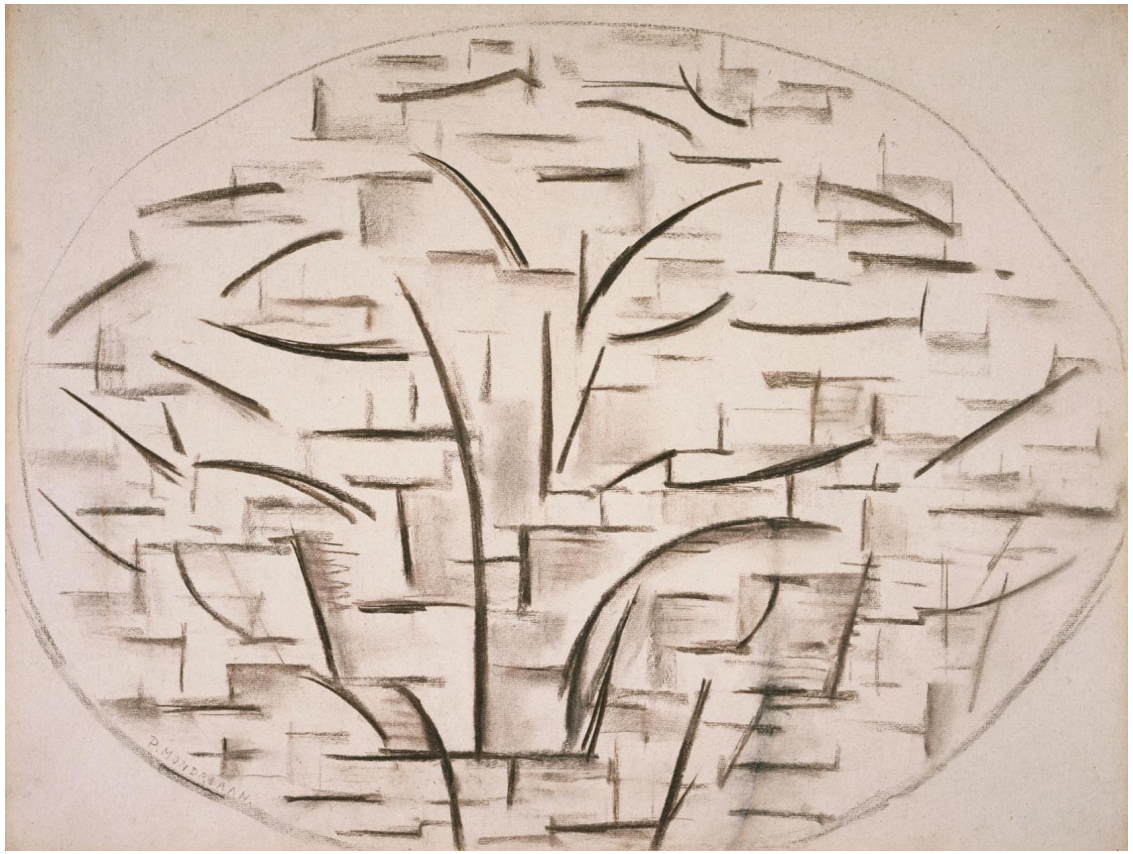
O n e

The acts of the mind, wherein it exerts its power over simple ideas, are chiefly these three: 1. Combining several simple ideas into one compound one, and thus all complex ideas are made. 2. The second is bringing two ideas, whether simple or complex, together, and setting them by one another so as to take a view of them at once, without uniting them into one, by which it gets all its ideas of relations. 3. The third is separating them from all other ideas that accompany them in their real existence: this is called abstraction, and thus all its general ideas are made.

John Locke, An Essay Concerning Human Understanding (1690)

Mondrian: Foundations

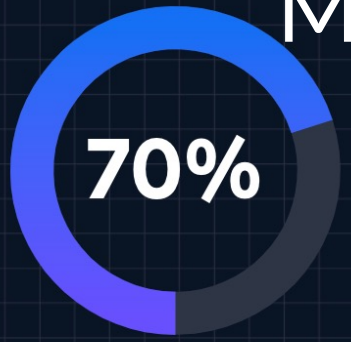
November 11, 2023–April 28, 2024 at the MFA in Boston



Markdown as a HTML abstraction

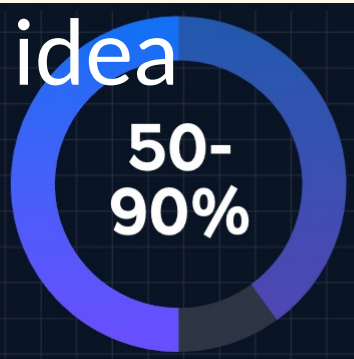
- The markdown format was invented by a mac enthusiast and blogger, John Gruber, who wanted a simpler format to write blogs.
- It has become the lingua-franca of the technical web, used by programmers to write documentation, web sites to generate web pages, and I can even use it to make my slides.
- It is a plain-text format, with just enough bells and whistles to make meaningful conversion to html possible: it abstracts some parts of html into a simpler format.
- There is a trade-off, you cant place images and text as perfectly as you can on the web
- Let us make a website using this abstraction. Go to https://github.com/hult-cm3-rahul/cm3-website .

Markdown is a low-code idea



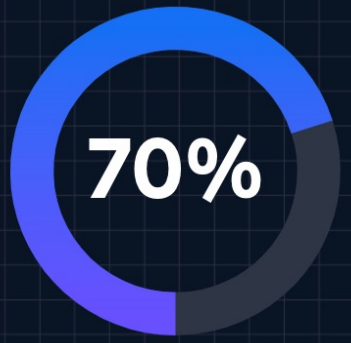
By 2025, 70% of apps will be built using no-code/low-code technology

Gartner



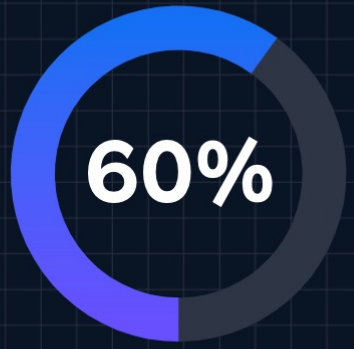
Low-code platforms can reduce development time by 50% to 90%

451 Research



70% of users with no development experience learned low-code in 1 month

mx mendix

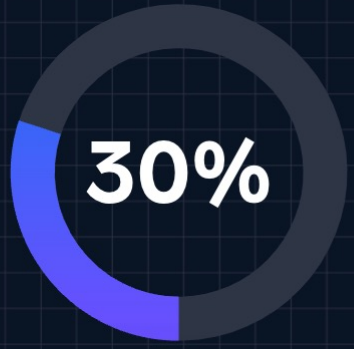


60% of apps are now built outside of the IT department

TechRepublic

Organizations with citizen developers score 33% higher on innovation measures

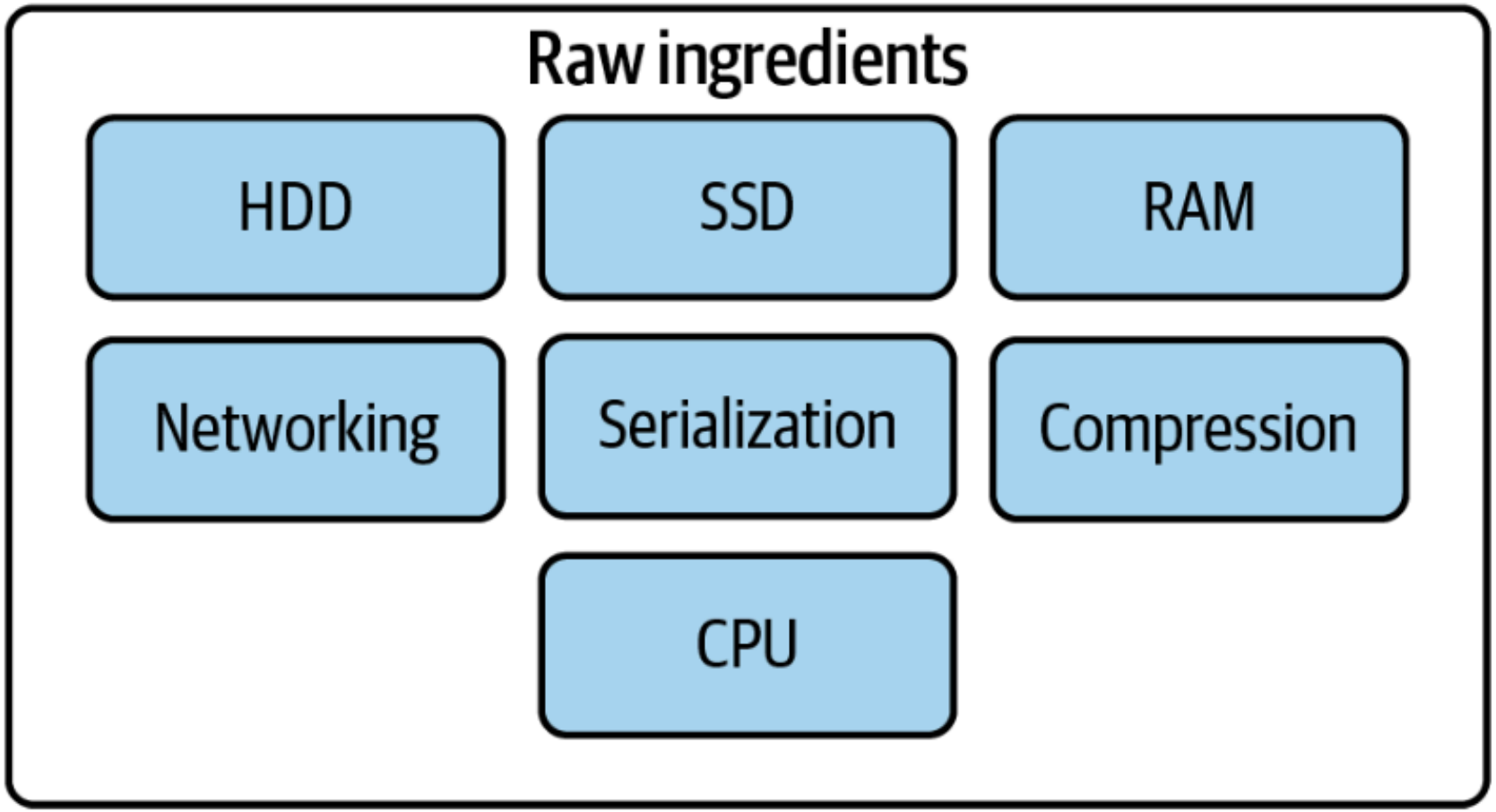
McKinsey&Company



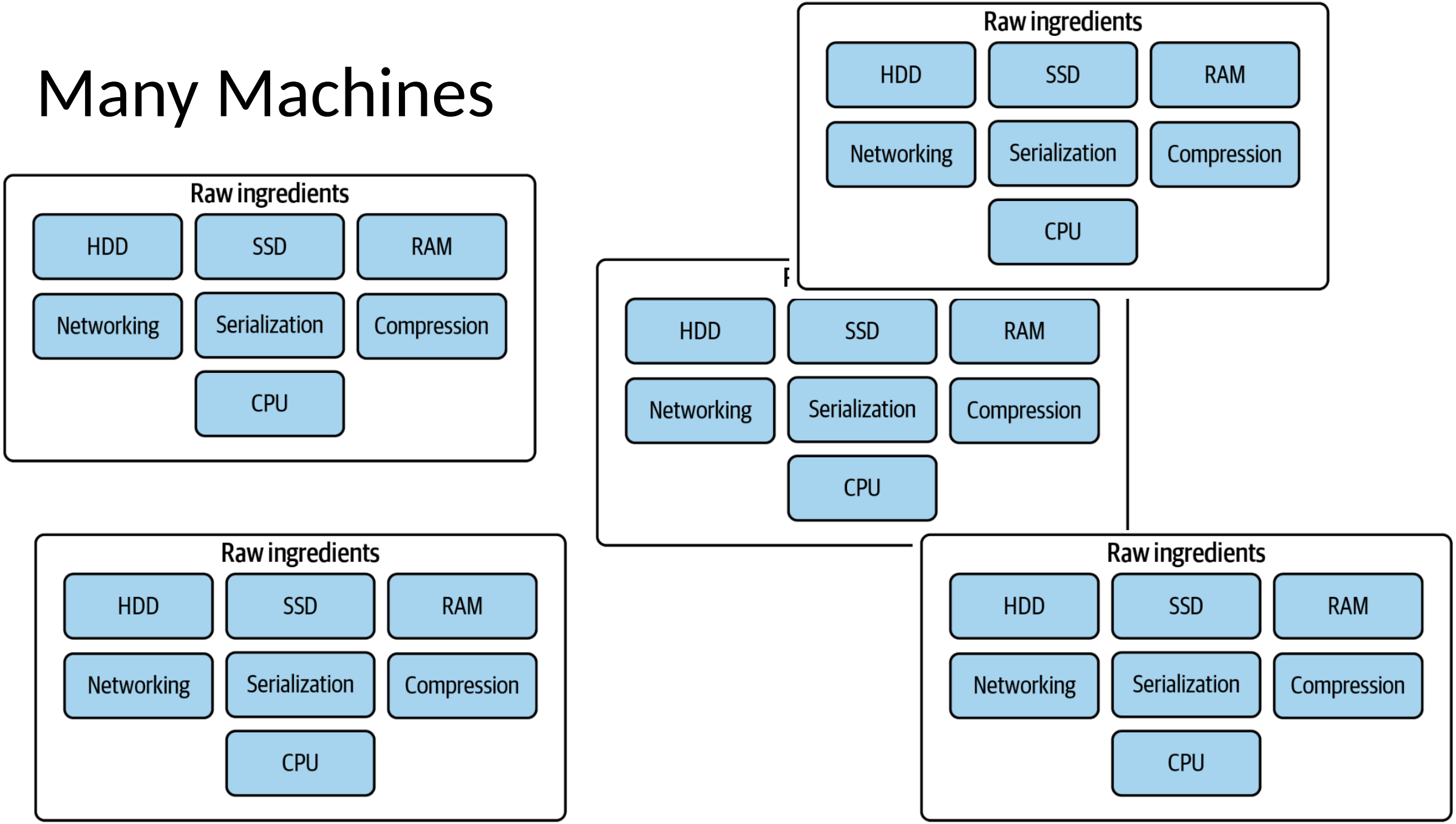
33% of organizations plan to use no-code/low-code to build better digital customer experiences

mx mendix

Abstracting Storage: stuff happening on one machine

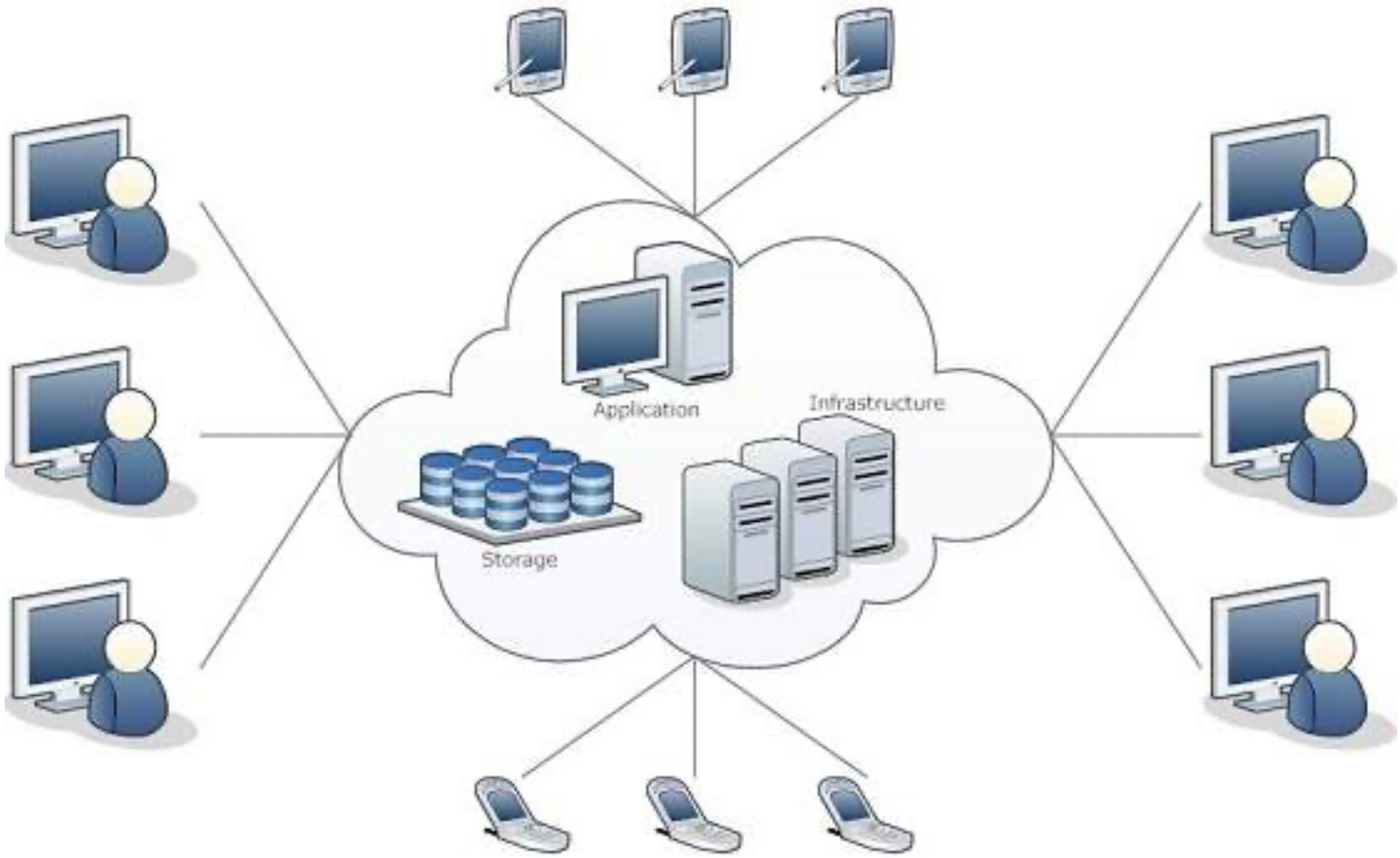


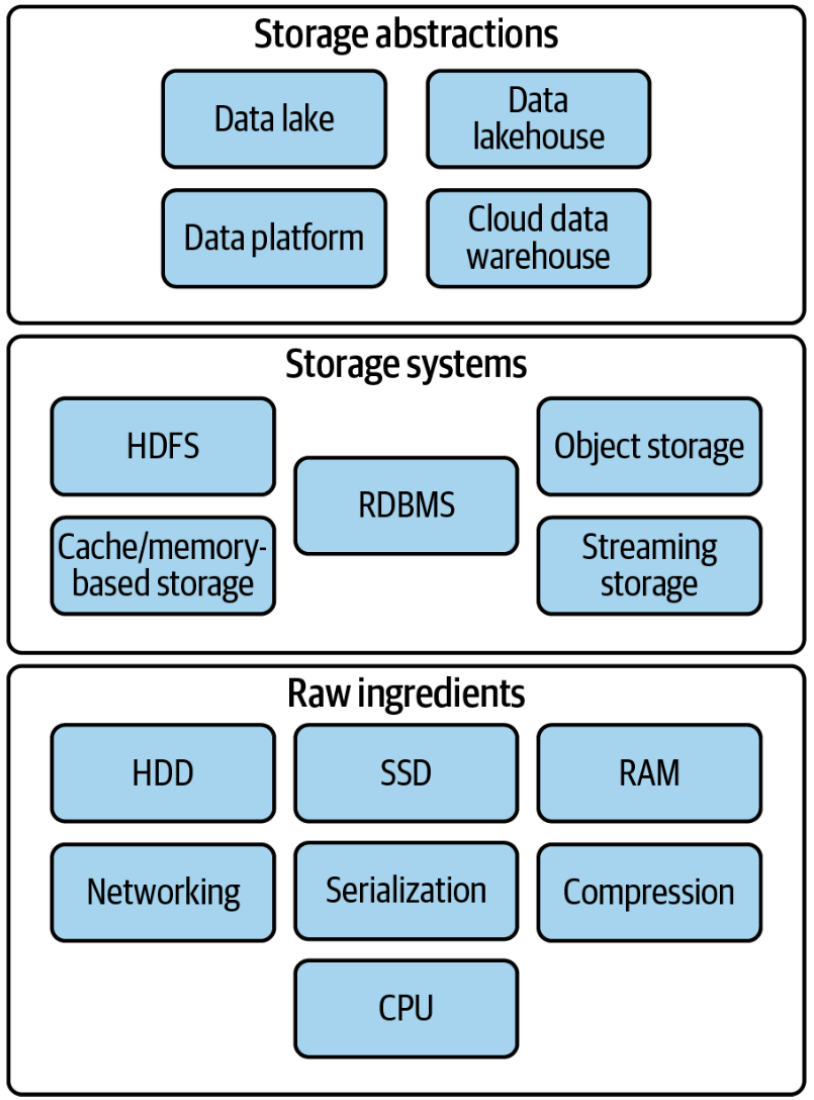
Many Machines



The Cloud is an abstraction over storage and compute

Multiple Layers of Abstractions





Amazon S3 as a abstraction

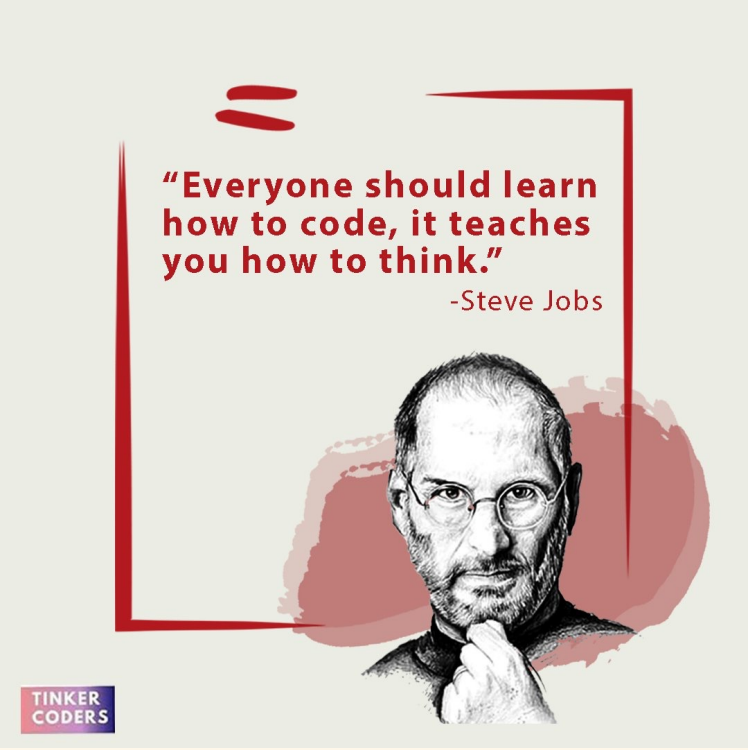
- S3 is a key-value looked up object store, a bucket for files
- It is an abstraction over persistent storage, and machines used to serve files to us
- each file has a name, something like s3://rahul-data-lake/nyc-taxi-data/2024-01-23 . The slashes dont actually represent folders but we can use them in our name
- all we need to do is to give our programs is a bucket
- There are always trade-offs: asking for all the files starting with s3://rahul-data-lake/nyc-taxi-data is expensive as it is not a folder.

Disruption

- Amazon S3 was launched in 2006, and disrupted the entire storage industry, leading to declines in business for co-located disk server companies, NAS providers such as EMC. CAPEX went down.
- It gave rise to competitors and enabled many other businesses to thrive, at first creating hadoop big-data clusters.
- Then it gave rise to companies like snowflake which used the object store as a backend for a database, disrupting older database companies such as oracle and IBM
- Finally it gave rise to data-lakes, and currently data lake-houses, and many established companies and startup products, such as duckdb, apache iceberg, databricks delta-lake and so on. It has revolutionized the entire database industry.

HULT
INTERNATIONAL
BUSINESS SCHOOL

Programming



What is Programming ?

When you write code, you tell a computer ”What to do”.
Those directions are then carried out by the computer, and create action.

Programming Language

- A special purpose and limited language
- A set of rules and symbols used to construct a computer program
- A language used to interact with the computer

Thinking Like a Programmer

Programmers have to think in a certain way in order to write code that does exactly what we want it to do. Computer and Robots may be smart, but only because we teach them to be i.e Set of Rules that needs to be imbibed to foster self learning mechanism.

Pseudocode High-level description of a computer program or algorithm that uses a mixture of natural language and informal programming language-like syntax. It's not meant to be executed on a computer but rather to help programmers plan and communicate the logic of their algorithms before translating them into a specific programming language.

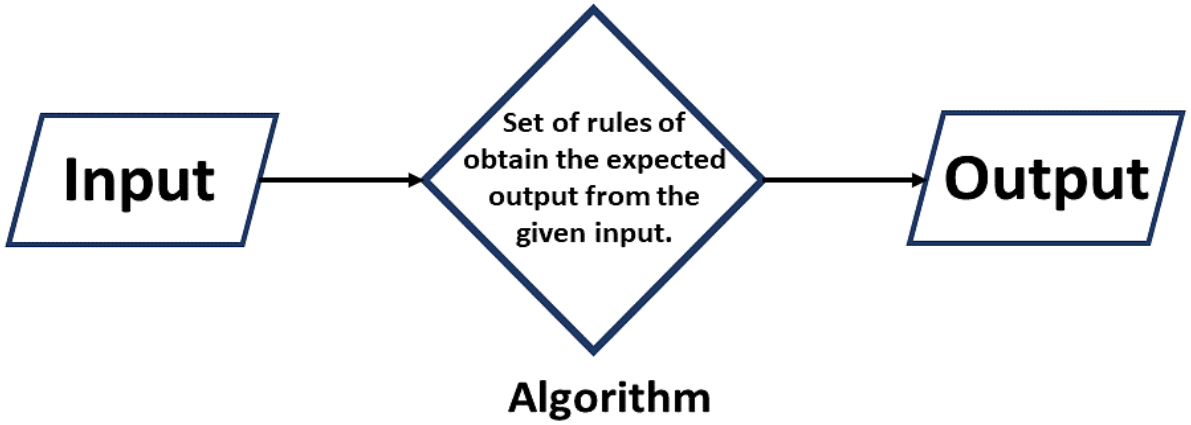
1. Start
2. Input num1
3. Input num2
4. Set sum = num1 + num2
5. Display sum
6. End

This pseudocode outlines the steps of a program that takes two numbers as input, adds them together, and then displays the result.

Introduction to Conceptual Thinking – Algorithms

What is an Algorithm?

An algorithm is a set of commands that must be followed for a computer to perform calculations or other problem-solving operations.



Algorithm

•Problem: A problem can be defined as a real-world problem or real-world instance problem for which you need to develop a program or set of instructions. An algorithm is a set of instructions.

•Algorithm: An algorithm is defined as a step-by-step process that will be designed for a problem.

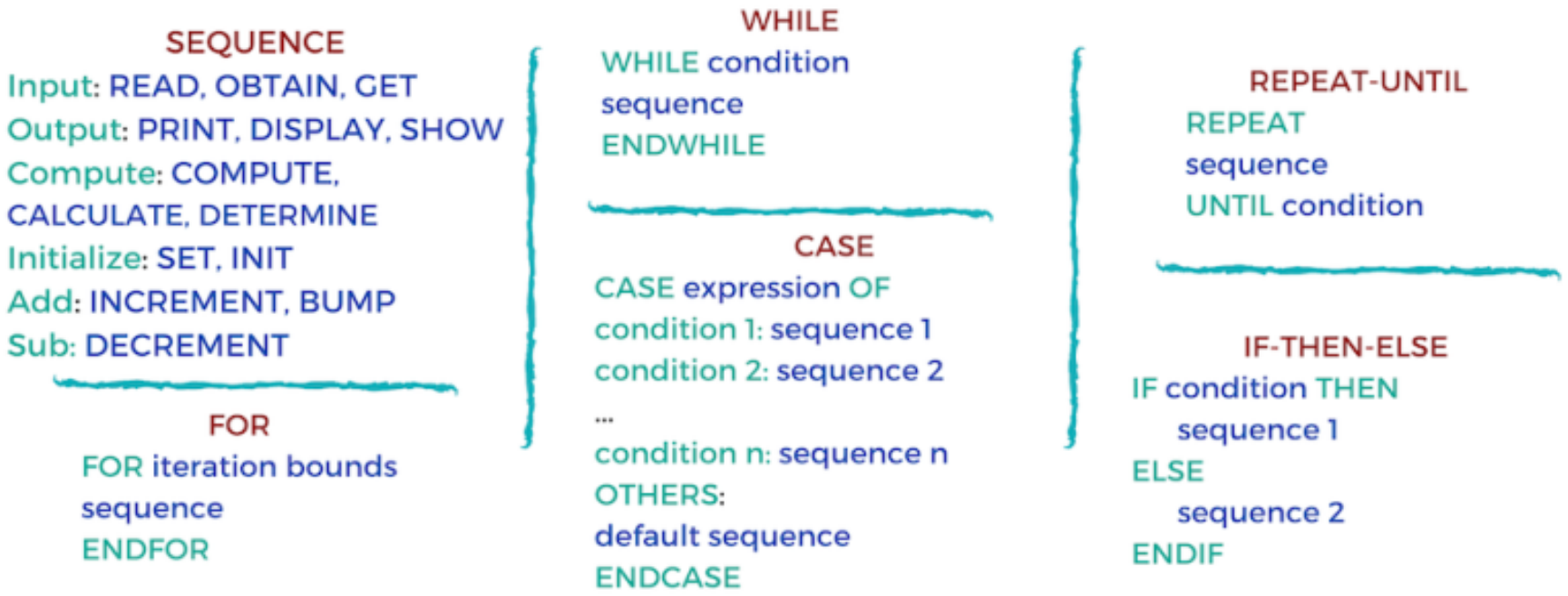
•Input: After designing an algorithm, the algorithm is given the necessary and desired inputs.

•Processing unit: The input will be passed to the processing unit, producing the desired output.

•Output: The outcome or result of the program is referred to as the output.

1. Start
2. Input num1
3. Input num2
4. If num1 > num2
5. Set max = num1
6. Else
7. Set max = num2
8. End if
9. Display max
10. End

This pseudocode describes a program that takes two numbers as input, compares them, and then displays the larger of the two. It uses a basic conditional structure (an if-else statement) to determine which number is greater.



Designing the algorithm

Example: Consider the example to add three numbers and print the sum

Algorithm to add 3 numbers and print their sum:

1.START
2.Declare 3 integer variables num1, num2, and num3.
3.Take the three numbers, to be added, as inputs in variables num1, num2, and num3 respectively.
4.Declare an integer variable sum to store the resultant sum of the 3 numbers.
5.Add the 3 numbers and store the result in the variable sum.
6.Print the value of the variable sum
7.END

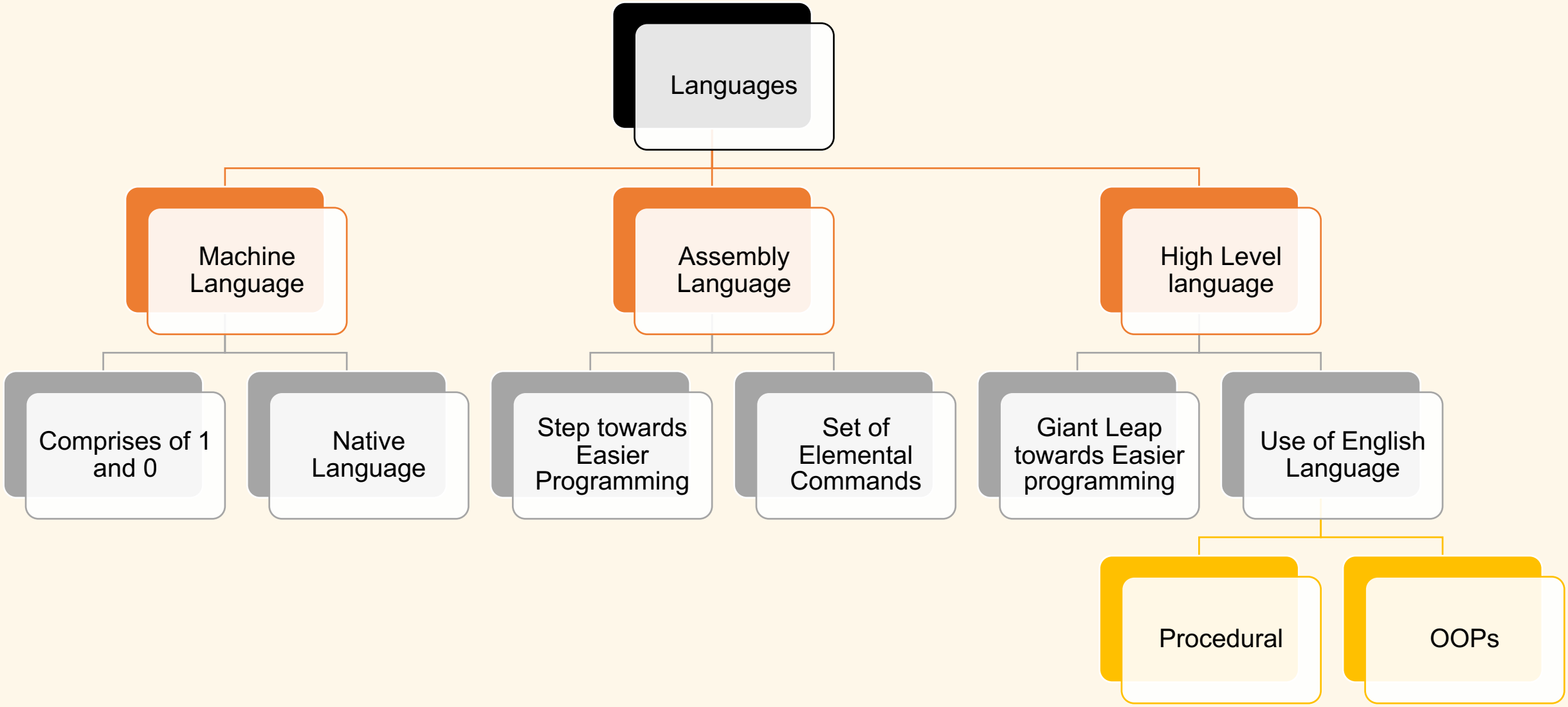
This had space for 4 variables. What if you had space for only 2?

Space and Time

- With algorithms there is usually a space-time tradeoff
- If you have more memory, you can usually do things faster by finding a data structure in which you stuff your data
- For example, think of using binary search to find something. If you can somehow store a sorted array, you can directly use binary search.
- This is the key idea behing binary search trees, generalized to b-trees in databases.

HULT
INTERNATIONAL
BUSINESS SCHOOL

Programming Languages

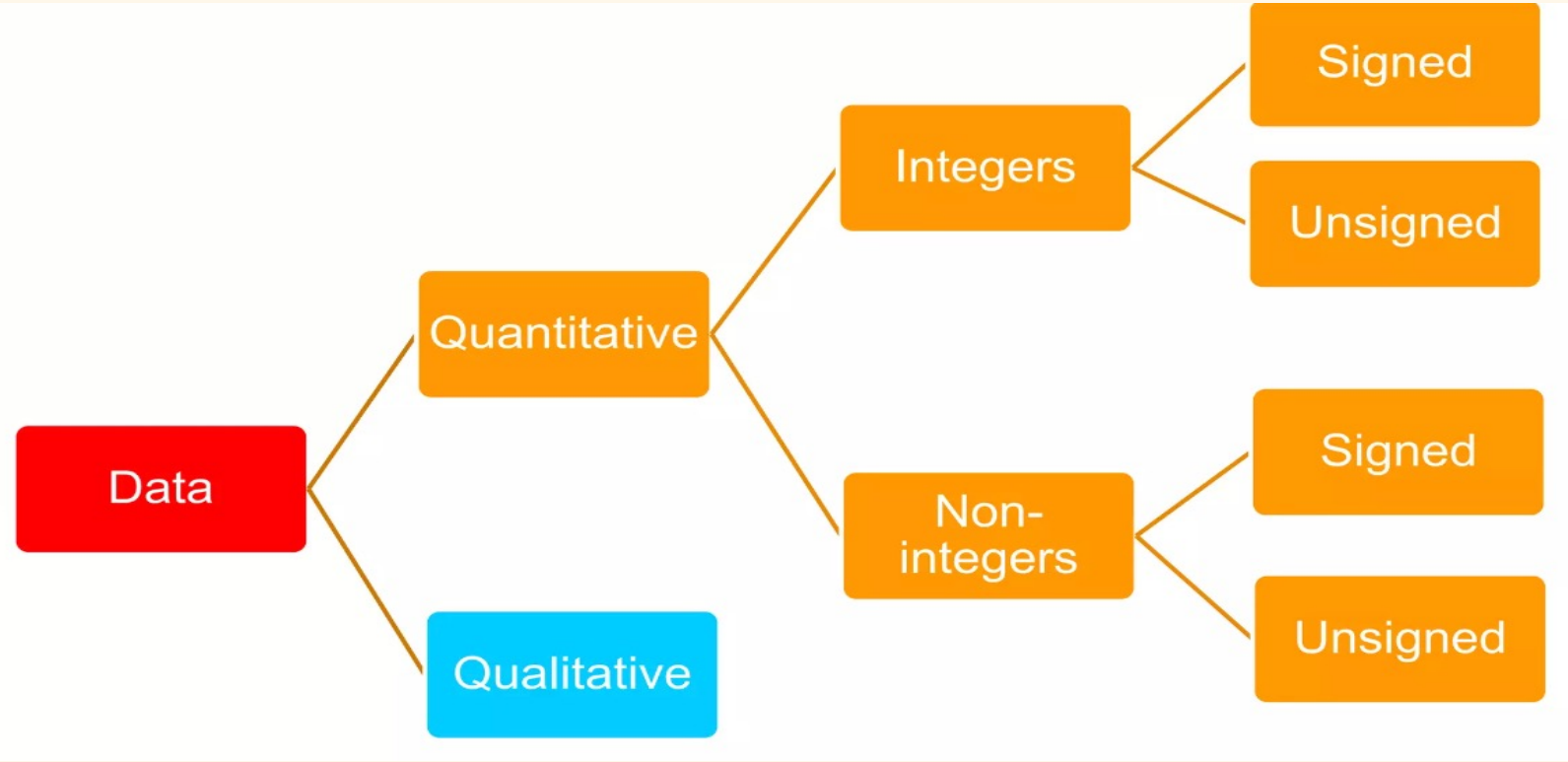


HULT INTERNATIONAL BUSINESS SCHOOL

Data Representation

Data representation refers to a manner in which data is stored into the Computer

- Qualitative
- Quantitative



Pseudo-code is an abstraction for real code. Lets see some real code!

Lets see some python

https://github.com/hult-cm3-rahul/LearningPython