**AGENDA – Module 0 – DAY0 – MDTM29**

* INTRODUCTION TO PROGRAMMING
* WHAT IS DATA SCIENCE AND WHAT IS MACHINE LEARNING
* COMPETITIVE PROGRAMMING WITH PYTHON (CODEKATA)
* INDUSTRY TRENDS & MAIN BOOTCAMP FLOW

**Understanding Computer System**

* Computer System
  + Components required to process and store data.
  + Hardware: Physical equipment of a computer.
  + Software: Instructions telling hardware what to do using programs.
* Main Purposes of Hardware and Software
  + Input data
  + Processing data
  + Output data

**Introduction to Programming**

* Definition of a Program
  + A sequence of instructions in a programming language that a computer executes to perform a specific task.
* Programming
  + Writing software instructions.

**Programming Language**

* A programming language is like a special language that lets you communicate instructions to a computer. It's different from human languages because it's designed for computers to understand and follow precisely.

**Program Development Cycle**

1. Understanding the Problem
   * Grasping end user requirements.
   * Supporting documentation.
2. Planning the Logic
   * Using pseudocodes and flowcharts.
   * Desk checking (verifying logic on paper before coding).
3. Coding the Program
   * Writing code in a chosen programming language.
4. Converting Program to Machine Language
   * Using compilers and interpreters.
5. Testing the Logic and Program
   * Ensuring correctness through test cases.
6. Putting the Program in Production
   * Deploying and refining based on feedback.

**Understanding the Problem**

* Difficult but crucial.
* Involves understanding end user needs and documentation.

**Planning Logic**

* Heart of Programming
  + Tools: pseudocodes and flowcharts.
  + Algorithm: A blueprint of the program.
    - Steps to solve a problem.
    - Can be converted into any programming language.

**Algorithm for Adding Two Numbers**

1. Begin
2. Input two numbers, store them in variables a and b
3. Add the numbers and store the result in a variable Result
4. Output the value of Result
5. End

**Pseudocode for Adding Two Numbers**

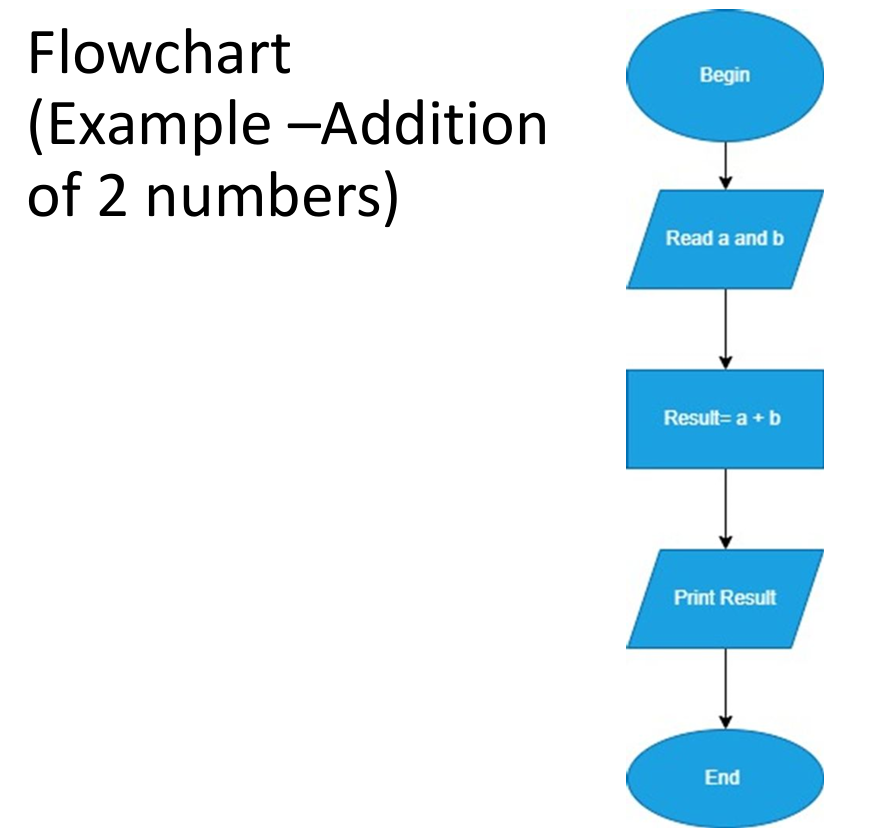
* BEGIN
* READ a, b
* C = a + b
* PRINT Result
* END

**Explanation:**

1. Begin: Start of the algorithm.
2. READ a, b: Input two numbers and store them in variables a and b.
3. Result = a + b: Add the values of a and b, store the sum in Result.
4. PRINT Result: Output the value of Result.
5. End: End of the algorithm.

Flowchart

* Visual Representation of an algorithm or program.



* **Programming Languages**
  + Low-Level Language (Machine Language): Difficult to program (Binary 0,1).
  + High-Level Language (HLL): English-like syntax, readable, efficient, needs translation to machine language (e.g., C++, Java, Python, C#, PHP).

**Role of compiler\interpreter**

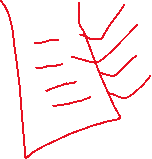
Main role is to translate the code to a high-level language program to a machine level language program

**How does it work?**

* **Read The HLL Code**
* **Check the correctness**
* **Optimization**
* **Translate and generate executable code**

**Role of Compilers and Interpreters**

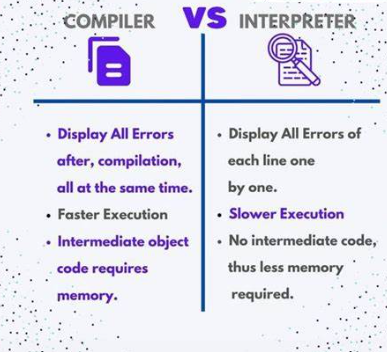
* Compiler
  + Reads entire program.
  + Generates and stores machine code after processing the whole program.
  + Optimizes code.



* + Fast (e.g., C++, Java).



* Interpreter
  + Reads and converts code line by line.
  + Executes code immediately without storing.
  + No optimization.
  + Slower (e.g., Python, Perl).

****

**Testing the Program**

* Check syntax and logic with various test cases.
* Ensure logically correct output.

**Putting Program in Production**

* Deployment can take months.
* Continuous testing and improvement based on feedback.

**Solving Problems with Computers**

* Human vs. Computer Logic
  + Example: Adding multiple numbers.
    - Human: Add digit by digit with carry.
    - Computer: Sequentially add numbers and store the result.

**Conclusion**

* Understanding how computers solve problems is crucial.
* Writing pseudocodes and flowcharts helps in planning and visualizing the logic.

**WHAT IS DATA**

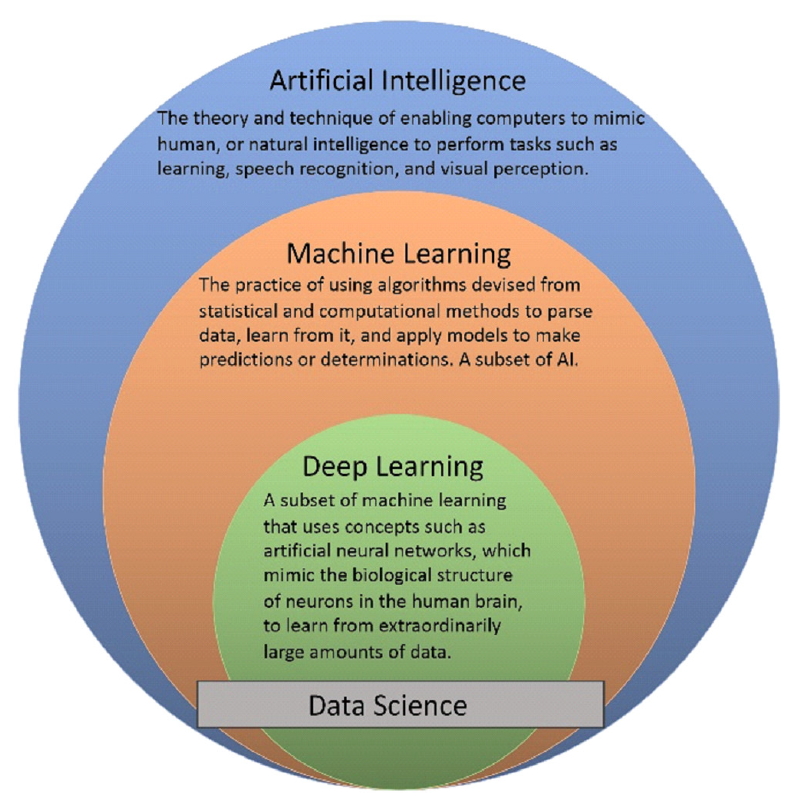
We live in an era inundated with data. Websites meticulously record every user's click. Your smartphone continuously logs your location and speed every second of the day. Smartwatches monitor heart rates, movement, diet, and sleep patterns around the clock. Smart cars track driving behaviours, smart homes observe living habits, and smart marketers analyse purchasing patterns. The Internet itself is a vast web of knowledge, encompassing a massive, cross-referenced encyclopaedia, along with specialized databases on topics like movies, music, sports results, pinball machines, and memes. Hidden within this data are answers to countless questions that no one has ever thought to ask.

**WHAT IS DATA SCIENCE**

Data Science is a field that combines skills from statistics, computer science, and specific areas of knowledge to make sense of data. It involves collecting data, cleaning it up, and analysing it to find patterns and insights. Using techniques like machine learning, data scientists create models that can predict outcomes and help solve problems. They present their findings using visual tools like charts and graphs, making it easier for others to understand and use the information to make better decisions

**WHAT IS MACHINE LEARNING**

Machine Learning is a branch of artificial intelligence that enables computers to learn from data and make decisions or predictions without being explicitly programmed for each specific task. By analysing patterns and relationships in data, machine learning algorithms can improve their performance over time as they are exposed to more information. This technology is widely used in applications such as recommending products, detecting spam, recognizing speech, and powering autonomous vehicles, allowing computers to perform tasks that usually require human intelligence and adapt as they gain more experience.



* **COMPETITIVE PROGRAMMING WITH PYTHON(CODEKATA)**

**WHY IT IS IMPORTANT TO SOLVE ATLEAST 10 PROBLEMS A DAY**

* **Better Problem-Solving**: Users get better at solving tricky problems and creating smart solutions, which boosts their overall thinking skills.
* **Improved Coding Skills**: Practicing with Python helps users write clearer, more effective code, making them better programmers.
* **Quicker Development**: Python’s easy-to-use features let users write and test their code faster, speeding up the development process.
* **Career Boost**: Being good at competitive programming with Python can open up more job opportunities in tech fields like software development and data science.

**INDUSTRY TRENDS**

**Responsible AI:**

* Imagine AI applications that are fair, unbiased, and don't discriminate. That's what Responsible AI is all about.
* As AI becomes more powerful, there's a growing concern that it could perpetuate social biases or lead to privacy breaches.
* This trend focuses on building AI systems with safeguards in place to ensure they're ethical and trustworthy.

**2. Generative AI:**

* Imagine a computer program that can create realistic images, write different kinds of creative content, or even generate new data sets. That's the potential of Generative AI.
* This is a rapidly developing field, but there's debate on how practical it is for real-world applications yet.
* Still, the ability to create new content or data for training other AI systems makes it a hot topic.

**3. Edge Intelligence:**

* Traditionally, data is sent to powerful computers in the cloud for analysis. With Edge Intelligence, processing happens closer to where the data is generated, on devices themselves.
* This is especially relevant for the Internet of Things (IoT) where numerous devices generate data constantly.
* Edge Intelligence allows for faster analysis and potentially reduces reliance on cloud infrastructure. Explore IOT(Internet of Strings)
* **BOOT CAMP FLOW**
* 3-4 Months
* 15-20 Weeks
* Module 1: Python Basics
* Covers fundamental Python concepts like data types, variables, loops, functions, etc.
* Module 2: Python Advanced
* dives into more advanced topics like comprehensions, file handling, regular expressions, object-oriented programming, etc.
* Module 3: Algorithmic Thinking with Python
* Introduces algorithmic thinking and problem-solving techniques using Python.
* Module 4: Data Handling in Python - 2 weeks-
* Teaches data manipulation using Pandas library and explores MongoDB database.
* Module 5: SQL
* Covers the basics of SQL queries, schemas, and normalization for relational databases.
* Module 6: SQL Advanced-
* Introduces advanced SQL concepts like joins, date/time functions, and subqueries.-1Week
* Module 7 & 8: Probability and Statistics with NumPy
* Explores probability and statistics concepts along with hands-on practice using NumPy.
* Module 9: Data Visualization in Python-
* Covers data visualization techniques with libraries like Matplotlib, Seaborn, and Plotly1 Week.
* Module 10: Data Engineering with Python
* Introduces data cleaning and exploratory data analysis (EDA) techniques.
* Module 11: Exploratory Data Analysis with Python
* Deep dives into Exploratory Data Analysis (EDA) for real-world data.-1 Weeks
* Module 12-15: Machine Learning with Sklearn
* Introduces machine learning concepts, supervised learning algorithms (regression & classification) with evaluation metrics using scikit-learn library.
* Module 16: Machine Learning with Sklearn - Continued
* Explores unsupervised learning algorithms and their applications.-4 weeks
* Module 17-19: Deep Learning
* Covers the fundamentals of deep learning models, including neural networks, with PyTorch implementation.
* Module 20: Natural Language Processing
* Introduces Natural Language Processing (NLP) techniques with applications like sentiment analysis and question answering.
* Module 21: Computer Vision
* Explores computer vision fundamentals and deep learning solutions for tasks like object detection.
* Module 22: Model Deployment in AWS Cloud Platform-
* Introduces deploying machine learning models on the AWS cloud platform.
* Module 23: Putting it together - Solving DS problems
* Provides hands-on experience with industry projects under expert guidance.
* Module 24: Mock Interviews
* Prepares you for job interviews with mock sessions conducted by industry professionals.

**ASSIGNMENT**

**Writing pseudocodes and flowcharts for simple and moderate business logic.**

**1.Subracting 2 Numbers**

**2. Multiplying 2 Numbers**

**3. Adding of 5 numbers**

**4. Average of 5 Numbers**

**5.Even or odd number**

**6. Flow chart Division**

**Flowchart**

**•Picture representation of an algorithm/program**

**•Some of the symbols used in flowchart are**

****