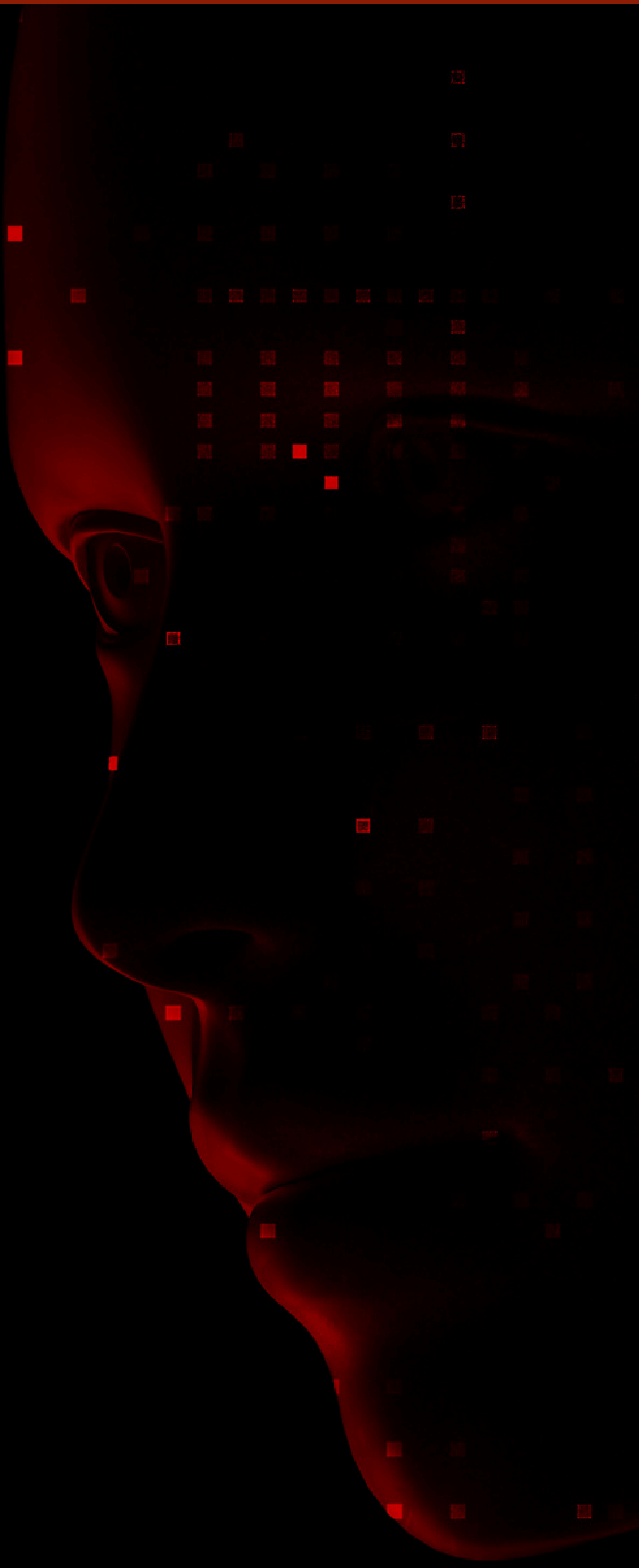
SIN EDUCATION

Professional AI Foundation Specialization (SOAI001)



- Python for AI and Data Science
- Foundations of AI
- Generative AI

SIN SCHOOL OF AI

COURSE CONTENT

PROFESSIONAL AI FOUNDATION SPECIALIZATION (SOAI001)

PYTHON FOR AI AND DATA-SCIENCE

- Introduction to Python Crash Course
    - Survey of 2019 by Stack Overflow.
    - Why to take this Crash Course ?
    - Program Surveys
    - Take a Minute to Setup yourself up for success.
    - Welcome to Course.
    - Course Prerequisites
    - How to pass the class ?
    - Finding out more information
    - Meet and Greet
- Beginning of your programming Journey
    - Python Relevant to IT Industry.
    - Code Comparison with Python.
    - Key Terms.
    - Resources.
    - Coding Blocks.
    - Getting information from the user.
    - Functions.
    - Keywords.
    - Arithmetic Operators.
    - Order of Operations.

    - Resources for more information.
    - Use the command line Windows and MacOS and Linux.
    - Using IDLE
    - Use Jupyter Lab and Notebooks.
    - Jupyter Resources.
    - Colab and its features.
    - Create a Colab Notebook.
    - VS Code and using VS Code with Python.
    - PyCharm and PyDev
    - Glossary Terms.
    - Quiz.
- Expressions and Variables.
    - Explore Python Syntax.
    - The Language of Python.
    - Basic Terminologies.
        - Variables.
        - Keywords.
        - Operators.
        - Expressions.
        - Functions.
        - Conditional Statement.
        - Naming Rules and Conventions.
        - The Zen of Python.
        - Resources of More information.
    - Annotating Variables By type.
    - How to annotate a variable.
    - Dynamic Typing.
    - Duck Typing.

    - Annotating variables with type comments.
    - Annotating variables directly.
    - How type annotations after runtime behavior.
    - Key takeaways.
  - Implicit Conversions.
  - Explicit Conversions.
  - Expressions and variables key terminologies.
  - Variables annoted by type.
  - Coding Skills.
    - Skill Group 1
    - Skill Group 2
    - Skill Group 3
    - Skill Group 4
  - Defining Functions
    - Built-in functions.
      - print()
      - type()
      - passing one functions into another.
      - str()
      - sorted()
      - max() and min()
    - Returning values.
  - None datatype in python
  - Principles of Code Reuse.
  - Code Style.
    - Self Documenting code.
    - Refactoring.
    - Terminologies.

- Terms.
- Knowledge.
- Coding Skills.
    - Skill Group 1
    - Skill Group 2

- Conditionals
    - Comparisons operators with equations.
        - Part 1 : Equality == and not equal to ≠ Operators.
            - Equality == operators versus the equals = operator
        - Part 2 : Greater than > and less then < operators.
        - Part 3 : Greater than or equal to ≥ and less than or equal to ≤ operators.
        - Part 4 : Practice
        - Resources.
    - Comparisons operators with string.
        - Part 1 : Equality == and not equal to ≠ Operators with Strings.
        - Part 2 : the Greater than > and less than < Operators.
        - Part 3 : The Greater than or equal to ≥ and less than or equal to ≤ Operators.
    - Logical Operators
        - Part 1 : The and Logical Operator
        - Part 2 : The or Logical Operator
        - Part 3 : The and Logical Operator
    - Branching if and else statements
    - Else Statement and examples.
    - Modulo Operator
    - Elif Statements
    - Complex Branching with Elif Statement.

  - Glossary.
- Loops
  - While Loops.
    - Anatomy of while loop.
    - Some more examples of While Loop.
    - Why Initializing Variables Matters ?
    - Infinite Loops and how to break them.
    - Break Statement in while loops.
    - Common errors in the while loops.
    - Terms used in the while loops.
    - Math Concepts
    - Coding skills for while loops.
      - Skill Group 1.
      - Skill Group 2.
  - For Loop.
    - When to use the for loop.
    - Range Function.
    - Start , Stop and Step.
    - Some of the best examples with the range functions are given below.
    - Key takeaways for range function.
    - Nested For Loops.
    - String and for Loops.
    - For Loops Recap.
- What are Strings ?
  - Using For Loops with String.
  - Key takeaways from above article.
  - Looping over a string.

    - How Looping over a String actually works.
    - For Loops.
    - While Loop with indexing.
    - While Loop with slicing.
    - List Comprehensions.
    - Additional advanced string loop techniques.
    - Slice and Join Strings.
    - What is Slicing and Joining String.
    - How to Slice Strings ?
    - How to Join Strings ?
    - How to combine slicing and joining strings ?
    - Common errors in for loops ?
    - Case 1
    - Case 2
- Recursion
    - Recursion in action in the it context .
    - Questions on recursion .
    - Glossary .
- Knowledge on Loops
    - Common functions.
    - Coding Skills.
    - Examples.
    - Remainder correct syntax is critical.
    - Common Syntax Errors.
- Strings in Python
    - Parts of String.
    - List index out of range.
    - Reverse indexing of String.

    - String indexing and slicing.
    - Creating new Strings.
    - Finding out the index of the string.
    - IN keyword in python.
    - Replace Domain Project.
    - Basic String Methods.
        - Strip() Methods
        - Count Method.
        - Endswith Method.
        - isnumeric() Method.
    - Typecasting in String.
    - Join Method in String.
    - Split() Method.
    - Split(Delimeter) Method.
    - Replace(OLD,NEW) Method.
    - String Formatting.
    - Limited Decimal Places.
- Lists.
    - Lists introduction.
    - Possible Index Errors.
    - List element access and formatting.
    - Modifying the contents of a lists.
- Tuples.
    - Tuples introduction.
    - Iterating over lists and tuples.
    - Using Enumerate() Function.
    - Other Examples.
- List Comprehensions.

  - List comprehensions examples.
  - For loops VS List comprehensions.
  - Lists comprehensions with conditional statement.
  - Common sequence operations.
  - List-specific operations and methods.
  - Tuple use cases.
  - The tuple() operator.
  - Tuples with mutable objects.
  - Returning multiple values from functions.
  - When to use for loops or list comprehensions.
  - Coding Skills.
    - Skill Group 1
    - Skill Group 2
    - Skill Group 3
    - Skill Group 4
    - Skill Group 5
    - Skill Group 6
    - Skill Group 7
  - Iterating over the contents of the dictionary.
  - Use of while loops and if else statement for dictionaries.
  - while loops.
  - if else statements.
- Methods on dictionary.
  - Introduction.
  - Dictionaries versus lists.
  - Both dictionaries and lists.
  - Dictionaries only.
  - Dictionary examples.

    - Lists only.
    - List Example.
    - Coding Skills.
        - Skill Group 1
        - Skill Group 2
        - Skill Group 3
- Object Oriented Programming
    - Introduction to OOPs.
    - Classes and Objects.
    - Object.
    - Classes.
    - New Keyword.
        - Object Based Language.
        - Object Oriented Language.
    - What is Method ?
    - Instance Methods.
    - Class Methods.
    - Static Methods.
    - Choosing a Method Type.
- Constructors and other special methods.
    - Creating an instance of class.
    - Modify variables.
    - Other special methods.
    - Special Methods.
- Methods as special operators
    - Different types of special operators.
    - performing special operations.
- Class and Methods.

    - Classes and instances.
    - special methods.
    - Document classes , methods and functions.
    - Exercises.
- Final Project.
    - Problem Statement.
    - Solving.
    - Research.
    - Planning.
    - Writing Script.
- Python in action
    - Problem and solutions.
    - the script.
    - your tools.
    - getting started.
- Find your path and perfect role.
    - Generalists vs specialist.
    - Choose your work environment.
    - Agency vs in-house teams.
    - Exploring technical careers.
- Creating and career development plan.
    - identifying career goals.
    - Creating a timeline.
- Transitioning to new career.
    - Think about your transferable skills.
    - Showcase your new skills.
    - Get to know more with networking.
    - Resources for more information.

- Getting Promoted.
- Ending Examinations and Projects are topic-wise and end exam will be taking place.

FOUNDATIONS OF AI

- Introduction to Artificial Intelligence.
    - Why to Study Artificial Intelligence ?
    - Correct Way to Study AI
    - Essential Tools for learning AI
    - Applications and use cases of AI
    - Data Science vs AI
    - Tools of Artificial Intelligence.
    - How AI can boost your productivity ?
- Overview of Machine Learning.
    - Welcome to Machine Learning
    - Applications of Machine Learning.
- Supervised vs Unsupervised Machine Learning.
    - What is Machine Learning .
    - Supervised Learning part 1
    - Supervised Learning part 2
    - Unsupervised Learning part 1
    - Unsupervised Learning part 2
    - Introduction to Jupyter Notebooks and Python.
        - Tools and Important Libraries.
    - Project 1 on Supervised Learning (Netflix recommendation system).
    - Project 2 on Unsupervised Learning (Book Recommendation system).
- Regression Model
    - Linear Regression model part 1

  - Linear Regression model part 2
  - Model Representation (LAB 1)
  - Cost function formula.
  - Cost function intuition.
  - Visualizing cost function.
  - Visualization examples.
  - Cost function (LAB 2)
  - Project 3 : Regression model (House Price Prediction)
- Training model with gradient descent
  - Gradient descent.
  - Implementing gradient descent.
  - Gradient Descent intuition.
  - Learning rate.
  - Gradient descent for linear regression.
  - Running gradient descent.
  - Gradient Descent (LAB 3)
  - Project 4 : Gradient descent technique.
- Multiple linear regression
  - Multiple features.
  - Vectorization part 1
  - Vectorization part 2
  - Python , NumPy and vectorization (LAB 4)
  - Gradient descent for multiple linear regression.
  - Multiple linear regression (LAB 5)
  - Project 5 : Multiple linear regression
- Gradient descent in practice
  - Feature scaling part 1
  - Feature scaling part 2

    - Checking gradient descent for convergence.
    - Choosing learning rate.
    - Feature scaling and learning rate (LAB 6)
    - Feature engineering.
    - Polynomial Regression.
    - Feature Engineering (LAB 7)
    - Linear regression and introduction to scikit learn (LAB 8)
- Classification with logistic regression.
    - Motivations to learn classification.
    - Classification (LAB 9)
    - Logistic regression.
    - Sigmoid function and logistic regression (LAB 10)
    - Decision Boundary
    - Decision Boundary (LAB 11)
    - Project 6 : Logistic Regression (Heart Disease Prediction).
- Cost function for Logistic regression.
    - Cost function for logistic regression.
    - Logistic loss (LAB 12)
    - Simplified Cost function for Logistic Regression.
    - Cost function for Logistic Regression (LAB 13)
- Gradient Descent for Logistic Regression.
    - Gradient Decent Implementation.
    - Gradient Descent for logistic regression (LAB 14)
    - Logistic regression with Scikit learn (LAB 15)
- The problem of overfitting.
    - The problem of overfitting.
    - Addressing overfitting.
    - Overfitting (LAB 16)

    - Cost function with regularization.
    - Regularized linear regression.
    - Regularized logistic regression.
    - Regularization (LAB 17)
- Ending notes for Machine Learning.
- Resource packs for Machine Learning.
- Final Project for Machine Learning.
- Final Exam for Foundations of AI

RISE OF GENERATIVE AI

- **Understanding Generative AI**:
    - Exploring how generative AI works.
    - LLMs as a thought partner.
    - AI is a general purpose technology.
    - Intake Survey.
    - Discussion on AI.
- **Generative AI Applications**:
    - Writing.
    - Reading.
    - Chatting.
    - What LLMs can and cannot do.
    - Tips for prompting.
    - Image Generation (Diffusion Model).
    - Resource Pack 1.
- **Generative AI Projects:**
    - Using generative AI in software applications.
    - Trying generative AI code yourself (Lab-1).
    - Lifecycle of a Generative AI project.

  - Cost intuition.
- **Advance technologies : Beyond Prompting**
  - Retrieval Augmented Generation (RAG).
  - Fine Tuning.
  - Pretraining on LLM.
  - Choosing a model.
  - How LLMs follow instructions : Instruction tuning and RLHF.
  - Tool use and agents.
- **Generative AI and business**
  - Day-to-Day usage of web UI LLMs.
  - Task analysis of jobs.
  - Additional job analysis examples.
  - New workflows and new opportunities.
  - Teams to build generative AI software.
  - Automation potential across sectors.
- **Generative AI and society.**
  - Concern about AI.
  - Artificial General Intelligence.
  - Responsible AI.
  - Building more intelligent world.
  - Resource pack 2.
- Final Exam on Generative AI.

ENDING OF SPECIALIZATION

- Special Session on AI Tools and its usage.
- Final Exam on Specialization (Content from all 3 courses will be coming).
- Felicitation Ceremony (Awarding of degree and awards).
- Thanks giving.

Dr Vinod Purohit
(Chairman SOAI)

Er. Harshvardhan
(Course Instructor)