DTC – Data Science and Machine Learning in Python

Sifal, Kathmandu Phone No.: 01-446-7153

Data Science and Machine Learning in Python

OBJECTIVES

The DTC – Data Science and Machine Learning in Python course is targeted for beginners who want to:

- Learn how to think and write meaningful piece of code in Python.
- Learn how to read Python code that has been written by somebody else.
- Learn how to map literary description of a problem (requirement) to an application/library coded in Python. In summary, this course teaches how to program using Python programming language for data science and machine learning.

This is a core basic level course that is essential for anyone who have no prior programming experience but wish to be a professional Python engineer in future

TARGET GROUP

- Anyone who has some basic knowledge about programming and wants to learn to write applications in Python for any purpose e.g. curiosity, hobby, to complete an academic project, to work towards a career as Python programmer, to help in project management, etc.

Prerequisites:

- Basic knowledge about programming, bits/bytes, procedures, classes, computer architecture, etc. If you just have a theoretical knowledge that is perfectly okay but you should have strong convictions on what programming is, and what you hope to achieve from this class.
- Willing and eager to spend at least 10-20 hours (varying from student-to-student) per week outside of the training class to read/write codes in Python (self-study and practice).
- There is no prior educational level requirement for this course. Anyone from 10+2 student to someone who is doing her PHD in Genetic Engineering is welcome to take this course.
- If you are only interested in theory and have no interest/patience in spending at least 10 hours every week throughout the duration of the course, then this course is clearly not for you.
- If you have absolutely no idea about programming or do not see yourself doing programming in the next six -odd months, then this class may not be for you!

TRAINING METHOD

- The course is spread over 80 hours that consists of lecture and lab work.
- Lab exercises are mandatory, have a fixed deadline,
- and are graded. The course puts heavy emphasis on lab exercises because software programming can only be learnt well by explicitly putting into practice the principles that have been taught (i.e. in simpler terms – by doing lots and lots of coding). Late submission (past the deadline) of exercises incur some penalty from total points.
- Instructors may provide relevant lecture/lab notes to students as (and when) necessary in the form of printed handouts and or via emails.
- Instructors may provide supplementary code snippets to students via email or in lab class to support the theory and or lab material that is being taught.
- At the end of the course, students may have to give an exam (which will be optional), that will test their knowledge on the material covered during the course. This exam may be practical and/or theoretical and is mandatory for any student wishing to join a higher level.
- Students are graded on the basis of attendance, lab exercises and exam in the increasing order of importance.

In summary, the only effective way to learn programming is to write lots of code. So, in order to really make this training productive, students are encouraged to spend as much time as necessary to complete the lab exercises on time. As part of the course, students will spend at least 30 hours in the lab but especially if you are new to programming or are coming from a non-computer-science background, it is recommended that you spend at least 10-20 hours per week outside of the class on your own to practice coding in Python.

COURSE DURATION

- 80 hours
- Classes
  - ✓ Morning/Evening

COURSE STRUCTURE (2 Months)

Week-1 (10 hours) – Python Basics

- Introduction to Python Programming Language
- Core Data Structures of Python
- Number
- String
- List
- Tuples
- Dictionary
- Set
- Advance Operations on Core Data-structures
- Decision and Branching
- If, elseif, Else, Break, Continue
- Looping
- for in, while
- Functions
- Functions
- Lambda Functions
- Map, Reduce, Filter
- Function Recursion
- Decorators

Week-2 (10 hours) – Python

- List and Dictionary Comprehension
- Exceptions and Exception Handling
- File Handling
- Object Oriented Programming (OOP)
- Introduction to Classes
- Inheritance, Encapsulation, Polymorphism, Abstraction
- Method Overloading
- Building Custom Packages and Modules

Week 3 (10 hours) – Basics of Data Science

- Introduction to Data Science
- Introduction to Numpy and Matplotlib
- Matrix Operations with numpy
- Random Variable and Probability Distributions
- Probability
- Properties of Probability Distributions
- Mean, Median, Mode
- Variance, Skewness, Kurtosis
- Multivariate Normal Distribution
- Co-variance, Correlation
- Introduction to scikit-learn
- Data pre-processing techniques using scikit-learn
- Dimensionality Reduction as data pre-processing
- Principal Component Analysis (PCA)
- Linear Discriminant Analysis (LDA)

Week 4 (10 hours) – Machine Learning

- Introduction to Reinforcement Learning
- Q-Learning with Python
- Introduction to Clustering
- K-Means Clustering
- Agglomerative Clustering
- Introduction to Supervised Learning
- Naive Bayes Classification

Week 5 (10 hours) – Machine Learning

- Linear and Polynomial Regression
- K-Nearest Neighbors
- Decision Tree
- Balancing Bias vs Variance of ML Model
- Ensemble Learning
- Random Forest and Adaptive Boost
- Identifying Important Features of Data
- Support Vector Machines

Week 6 (10 hours) – Deep Learning

- Introduction to logistic regression
- Computation Graph and Gradient Decent
- Introduction to artificial neuron (Perceptron)
- Multi Layer Perceptron
- Introduction to Artificial Neural Networks
- Designing Artificial Neural Networks with Keras
- Gradient Decent Variants
- Classification and Regression using Neural Networks

Week 7 (10 hours) – Deep Learning

- Introduction to Convolutional Neural Network (CNN)
- Object Classification with CNN
- Standard CNN Architectures
- Introduction To Object Detection
- The YOLO Algorithm

Week 8 (10 hours) – Natural language Processing +Web Interface

- Introduction to NLTK
- Text pre-processing
- POS Tagging and Named-Entity Recognition
- Latent Semantic Analysis
- Introduction to Recurrent Neural Network
- Word2Vec Algorithm for Text Vectorization
- Natural language Processing with LSTM
- Giving Web Interface to ML Application using Flask/Django

LABS

Lab assignments will focus on the practice and mastery of contents covered in the lectures; and introduce critical and fundamental problem-solving techniques to the students.

DISCLAIMER

Please note that Deerwalk Training Center reserves the right to change the course syllabus of DTC – Data Science and Machine Learning course at any time without prior notification.