GUVI ZEN CLASSROOM TRAINING

Trainer: Chandramouli Das

Topic: Introduction to Data Science & Python Basics

Day: 1

**------------------------------------------------------------**

# **1. What is Data Science?**

Data Science is an interdisciplinary field that blends statistics, mathematics, and computer science with domain expertise. It is used to extract valuable insights from structured and unstructured data.  
  
At its core, Data Science helps organizations make decisions by:  
- Understanding user behavior  
- Improving product features  
- Predicting future trends  
- Automating tasks using AI/ML  
  
The output of data science can be dashboards, models, recommendations, or automated systems.

# **2. Real-World Applications**

**• Netflix: Movie recommendation using collaborative filtering.  
• Uber: Predicting estimated time of arrival and dynamic pricing.  
• Amazon: Product recommendations and inventory optimization.  
• Healthcare: Predicting disease likelihood based on patient data.  
• Finance: Detecting fraud in real-time transactions.**

# **3. Career Roles in Data Science**

**- Data Analyst: Visualizes and interprets data to answer business questions.  
- Data Scientist: Builds models to predict or classify based on historical data.  
- Machine Learning Engineer: Builds and deploys predictive systems at scale.  
- Data Engineer: Designs systems for data ingestion, storage, and transformation.**

# **4. Data Science Lifecycle**

The typical steps involved in solving a data science problem:  
1. Define the Problem  
2. Collect Data  
3. Clean and Prepare Data  
4. Explore the Data (EDA)  
5. Build Model(s)  
6. Evaluate Models  
7. Deploy the Solution  
8. Monitor and Improve

# **5. Python Setup with Anaconda**

*Python is the preferred language for Data Science. Anaconda simplifies Python package management and provides tools like Jupyter Notebook for interactive coding.  
  
Steps:  
• Download from https://www.anaconda.com  
• Install and launch Anaconda Navigator  
• Open Jupyter Notebook  
• Create and run Python notebooks interactively*

# **6. Python Fundamentals**

Let’s start with basic programming concepts using Python.

## **6.1 Variables and Data Types**

*A variable stores data that can be used later in the code. Python is dynamically typed.  
  
Example:  
name = "Alice"  
age = 28  
height = 5.9  
is\_employed = True  
  
Data Types:  
- int, float, str, bool, NoneType*

***Try this:***

city = 'Chennai'  
  
print("City:", city)  
print("Type of city:", type(city))

## **6.2 Operators**

**- Arithmetic: +, -, \*, /, %  
- Assignment: =, +=, -=, \*=  
- Comparison: ==, !=, >, <  
- Logical: and, or, not  
- Membership: in, not in  
- Identity: is, is not**

***Example:  
a = 5  
b = 2  
print(a + b)  
print(a > b)  
print(a % b)***

## **6.3 Challenge - Create Your Bio**

Create variables to store your name, age, skill, and location. Print each using f-strings.  
  
name = 'Chandramouli'  
age = 30  
skill = 'Python'  
location = 'Kolkata'  
  
print(f"Hi, I am {name}, a {skill} expert from {location}.")

# **7. Practice Problems**

1. Declare a variable for your favorite movie and print it.  
2. Take two numbers and show all arithmetic operations.  
3. Check if a string contains a certain letter.  
4. Use logical operators to evaluate True/False combinations.  
5. Ask the user to input name and greet them.

# **8. Summary**

**- Data Science solves real-world problems using data.  
- Python is easy and powerful for DS tasks.  
- You learned about variables, types, and operators.  
- You used f-strings, logic, and printed meaningful outputs.**

# **9. Day 1 Assignment**

**- Install Anaconda and open Jupyter  
- Create variables and test all operator types  
- Complete all practice questions  
- Submit a screenshot of your Python outputs**

## **Appendix 1: Concept Deep Dive**

*Extended notes and practice exercises on topic 1. Use this space to take notes, add more examples, or paste your Jupyter screenshots.  
  
Example Code 1:  
for i in range(5):  
 print('Loop iteration', i)*

## **Appendix 2: Concept Deep Dive**

*Extended notes and practice exercises on topic 2. Use this space to take notes, add more examples, or paste your Jupyter screenshots.  
  
Example Code 2:  
for i in range(5):  
 print('Loop iteration', i)*

## **Appendix 3: Concept Deep Dive**

*Extended notes and practice exercises on topic 3. Use this space to take notes, add more examples, or paste your Jupyter screenshots.  
  
Example Code 3:  
for i in range(5):  
 print('Loop iteration', i)*

## **Appendix 4: Concept Deep Dive**

*Extended notes and practice exercises on topic 4. Use this space to take notes, add more examples, or paste your Jupyter screenshots.  
  
Example Code 4:  
for i in range(5):  
 print('Loop iteration', i)*

## **Appendix 5: Concept Deep Dive**

*Extended notes and practice exercises on topic 5. Use this space to take notes, add more examples, or paste your Jupyter screenshots.  
  
Example Code 5:  
for i in range(5):  
 print('Loop iteration', i)*

## **Appendix 6: Concept Deep Dive**

*Extended notes and practice exercises on topic 6. Use this space to take notes, add more examples, or paste your Jupyter screenshots.  
  
Example Code 6:  
for i in range(5):  
 print('Loop iteration', i)*

## **Appendix 7: Concept Deep Dive**

*Extended notes and practice exercises on topic 7. Use this space to take notes, add more examples, or paste your Jupyter screenshots.  
  
Example Code 7:  
for i in range(5):  
 print('Loop iteration', i)*

## **Appendix 8: Concept Deep Dive**

*Extended notes and practice exercises on topic 8. Use this space to take notes, add more examples, or paste your Jupyter screenshots.  
  
Example Code 8:  
for i in range(5):  
 print('Loop iteration', i)*

# **10. Deep Dive: Real-World Case Study - Netflix Recommendations**

Netflix leverages Data Science to recommend personalized movies and shows to users. It collects data such as:  
- Viewing history  
- Device type  
- Time of watching  
- User preferences  
  
Based on this data, collaborative filtering and content-based filtering algorithms predict what you may like. These algorithms are continually trained on millions of datapoints and retrained as you interact more. The result is a highly personalized homepage that is unique to every user.

# **11. Deep Dive: Difference Between Data Scientist and Data Analyst**

Though they both work with data, their responsibilities differ:  
  
\*\*Data Analyst:\*\*  
- Works with dashboards  
- Cleans and visualizes data  
- Answers business-specific questions  
  
\*\*Data Scientist:\*\*  
- Builds predictive models  
- Uses ML/DL techniques  
- Works with unstructured data  
  
In smaller companies, these roles may overlap, but in larger organizations, they are distinct and specialized.

# **12. Interview Tips: Python Basics You Must Know**

When applying for entry-level Data Science or Analyst roles, you're often tested on the following:  
- Variables and Scope (local vs global)  
- Type Conversion (int to str, str to float)  
- Basic Loops and Conditions  
- Writing Functions  
- List vs Tuple vs Dictionary  
  
It’s advisable to practice small scripts and dry-run code to understand logical errors.

# **13. FAQs from Beginners**

Q1: Is Data Science only about coding?  
A: No. While coding is essential, you also need logical thinking, storytelling, and understanding of business context.  
  
Q2: How long does it take to become a Data Scientist?  
A: Varies based on background. Typically 6–12 months of consistent learning and hands-on practice.  
  
Q3: Can non-programmers learn Data Science?  
A: Absolutely! Python is beginner-friendly, and the field is welcoming to career switchers.

# **14. Practice Notebook Structure (Optional Homework)**

*Create a Jupyter Notebook with the following structure:  
1. Heading: Day 1 - Python Basics  
2. Section 1: Introduction (Markdown)  
3. Section 2: Code for Variables, Types, and Operators  
4. Section 3: Outputs with Explanations  
5. Section 4: Reflection - What did you learn today?  
6. Section 5: Mini Project - Write a self-introduction using variables*