



**DIGYQUANT  
ANALYTICS**



# Python Programming

## 10 Days

## LEARNING PLAN

APPLY NOW ➤

Swipe >>>

**Training Program in  
Machine Learning  
Starts From 12-Aug-2023**

[www.digyquant.com](http://www.digyquant.com)



**Follow Us For  
regular updates**



**+91- 6381057440**



1

## Python Basics

- **Introduction to Python and its features**
- **Installing Python and a code editor**
- **Python variables, data types, and basic operations**

## Practice Questions

Swipe >>>

1. Write a Python program to calculate the area of a rectangle. Take input for length and width from the user.
2. Given a list of numbers, write a function to find the sum of all the even numbers in the list.
3. Write a program that takes a user's age as input and prints whether they are a child, teenager, adult, or senior citizen.
4. Create a Python function that converts a temperature from Celsius to Fahrenheit.
5. Write a Python program to check if a given number is prime or not.

# 2

## Control Flow and Loops

- **Conditional statements (if, else, elif)**
- **Using comparison and logical operators**
- **Loops: for and while**

### Practice Questions

Swipe >>>

1. Write a Python program that prints the first 10 numbers of the Fibonacci sequence.
2. Given a list of integers, write a function to find the largest and smallest numbers in the list.
3. Write a program that takes a string input from the user and counts the number of vowels and consonants in the string.
4. Create a Python program that generates a random number between 1 and 100 and lets the user guess the number.
5. Write a Python function to find the factorial of a given number using recursion.

# 3

## Lists & Tuples

- **Lists: Creating, indexing, slicing, adding, and removing elements**
- **List methods (append, extend, insert, remove, pop)**
- **Tuples: Creating, accessing, and basic tuple operations**

### Practice Questions

Swipe >>>

1. Given two lists, write a Python function to merge them and sort the resulting list in ascending order.
2. Create a Python function to find the factorial of a given number using recursion.
3. Write a program that takes a list of strings and removes all duplicate elements.
4. Implement a Python program to check if a given string is an anagram of another string.
5. Given a list of numbers, write a function to find the average and sum of the numbers.



4

## Dictionaries & Sets

- **Dictionaries: Creating, accessing, and modifying key-value pairs**
- **Dictionary methods (keys, values, items)**
- **Sets: Creating, adding, and set operations (union, intersection, difference)**

### Practice Questions

Swipe >>>

1. Write a Python program to create a dictionary of students and their respective grades. Allow the user to add new students and update their grades. Print the updated dictionary.
2. Write a Python program to find and print the top three students with the highest grades from the dictionary created in the previous question.
3. Write a Python program that takes two sets of integers as input and performs set operations to find and print their union, intersection, and difference.
4. Write a Python program that takes a sentence as input and counts the frequency of each letter (ignoring case). Display the result as a dictionary where keys are letters, and values are their respective frequencies.

# Training Program



- MACHINE LEARNING
- ARTIFICIAL INTELLIGENCE

12 AUGUST

ENROLL NOW

[WWW.DIGYQUANT.COM](http://WWW.DIGYQUANT.COM)





# 5

## Functions & Modules

- **Creating and calling functions with parameters and return values**
- **Built-in functions vs. user-defined functions**
- **Organizing code with modules and importing functions**

Swipe >>>

### Practice Questions

1. Write a function that takes a list of numbers as input and returns a new list containing only the even numbers.
2. Create a Python module with a function to calculate the area of a circle given its radius.
3. Write a program that reads a CSV file containing student names and their scores and calculates the average score.
4. Implement a Python function that checks if a given string is a palindrome.
5. Write a Python program to print the multiplication table of a given number.

# 6

## File Handling in Python

- **Reading from and writing to text files (open, read, write, close)**
- **Working with CSV and JSON files for data storage**
- **Using the "with" statement for automatic file closing**

### Practice Questions

Swipe >>>

1. Write a Python program that reads data from a CSV file containing student names and their scores. Calculate the average score and write the results to a new CSV file along with the students' names and scores.
2. Write a Python program that serializes a Python dictionary containing student details (name, age, grade, etc.) into a JSON file. Later, read the JSON file, deserialize the data, and display it.
3. Write a Python program that uses the "with" statement to open and read the contents of a text file. Display the content on the console, and the file will be automatically closed after reading.
4. Write a Python program that reads data from a CSV file containing sales records. Filter the data to find sales transactions above a certain threshold and write the filtered data to a new CSV file.

## Object-Oriented Programming (OOP)

- Understanding classes and objects
- Constructors and instance variables (`self`)
- Creating methods (instance, class, static)
- Encapsulation, inheritance, and polymorphism

### Practice Questions

Swipe >>>

1. Write a Python program that defines a class called "Car." Create objects of this class representing different car models and access their attributes and methods.
2. Write a Python program that defines a class called "Student." Implement a constructor to initialize the student's name and age as instance variables. Create objects of this class and display the student details.
3. Write a Python program that defines a class called "MathUtils." Implement instance methods to perform basic math operations (addition, subtraction, multiplication) and a class method to calculate the factorial of a number. Also, define a static method to check if a given number is prime.



## Python Libraries

- **Data Science:** Introduction to NumPy for numerical computing and Pandas for data manipulation
- **Web Development:** Exploring Flask or Django for building web applications
- **GUI Development:** Basics of Tkinter for creating graphical user interfaces

## Practice Questions

Swipe >>>

- Write a Python program that uses NumPy to create a matrix and performs various numerical operations like matrix multiplication and element-wise calculations. Then, use Pandas to read data from a CSV file, manipulate the data, and perform data analysis tasks like calculating mean, median, and standard deviation.
- Write a Python program that uses Tkinter to create a graphical user interface (GUI). Build a simple calculator application with buttons for arithmetic operations and an input field to display the result.

## Python Exception Handling and Debugging

- Handling exceptions using `try`, `except`, `else`, and `finally`
- Debugging techniques and common errors
- Reading and interpreting Python tracebacks

### Practice Questions

Swipe >>>

- Write a Python program that takes user input for two numbers and performs division. Use `try`, `except` blocks to handle the "`ZeroDivisionError`" if the user enters a zero as the divisor.
- Write a Python program that prompts the user to enter their age. Use `try`, `except` blocks to handle the "`ValueError`" if the user enters a non-integer value. Raise a custom exception if the age is negative or greater than 120.
- Common Error Handling: Write a Python program that reads data from a CSV file and performs calculations. Handle common errors like "`FileNotFoundException`" if the file doesn't exist and "`ValueError`" if the data cannot be converted to numeric values.



## Start Practicing Daily

- **Apply what you've learned in a mini-project (e.g., a simple calculator, a data analysis script)**
- **Practice on daily basis**

## Practice Questions

Swipe >>>

- **Write a Python program that takes a file as input and compresses it using the gzip or zlib library. Also, write a program to decompress the compressed file back to its original form.**
- **Write a Python program that uses the requests and BeautifulSoup libraries to scrape data from a website. Extract relevant information such as article titles, URLs, or prices, and display the data in a structured format.**
- **Write a Python program that uses the scikit-learn library to perform simple linear regression on a given dataset. Fit a linear model to the data and make predictions based on the regression model.**

# Training Program



- MACHINE LEARNING
- ARTIFICIAL INTELLIGENCE

12 AUGUST

ENROLL NOW

[WWW.DIGYQUANT.COM](http://WWW.DIGYQUANT.COM)

