Challenging Practice Questions - Python Basics, Conditions & Loops (ML & AI Context)

1. Sum of Digits (Loop Logic) Write a Python program that takes an integer input and calculates the sum of its digits (e.g., 123 → 6). ■ Hint: Think of this as feature extraction from numeric data.

2. Palindrome Checker Write a program that checks whether a string (e.g., 'madam') is a palindrome. ■ Palindromes are like symmetry detection in data preprocessing.

3. Number Guessing Game (AI Simulation) The program should generate a random number between 1 and 50. The user must guess it: - Print 'Too high' if the guess is greater. - Print 'Too low' if the guess is smaller. - Continue until correct. ■ This simulates an iterative learning process.

4. List Statistics Analyzer Given a list of numbers, calculate: - Maximum - Minimum - Mean - Standard Deviation (without using statistics module). ■ Think of this as dataset summary stats before training a model.

5. Prime Numbers in a Range Write a program that prints all prime numbers between 1 and 100 using a loop. ■ This is similar to filtering features/data that meet a condition.

6. Fibonacci Series Generator (While Loop) Write a program to generate the first N Fibonacci numbers (where N is input by the user). ■ Fibonacci shows recursion/sequence patterns, useful in time-series AI models.

7. Dictionary of Word Frequencies Ask the user to enter a sentence. Count how many times each word occurs and store it in a dictionary. ■ This is a basic NLP preprocessing step (like Bag of Words).

8. Student Records with Functions Define a function that takes a list of student dictionaries (with 'name', 'marks') and returns the student with the highest marks. ■ Simulates finding top-performing model/data point.

9. Password Strength Checker Write a program that checks if a password entered by the user is strong: - At least 8 characters - Contains both letters and numbers - Contains at least one special character (!@#$%) ■ This mimics input validation in AI systems.

10. File I/O – Dataset Cleaner - Write a file 'data.txt' containing numbers separated by spaces. - Read the file, remove duplicates, sort the numbers, and write them back to 'cleaned_data.txt'. ■ This mimics data cleaning before ML training.