Data Science Internship – February 2026

Internship Task Documentation

Task Instructions

1. Log in to your **LMS** and navigate to:
   Assessment & Task → Task 2: Feb Internship Logic Building Task – 2.
2. Open the **Google Form** provided in the task section to access your assigned Python problem.
3. Solve the problem using either **Jupyter Notebook** or **Google Colab**.
   Save your solution file in **.ipynb** format.
4. Upload (push) the `.ipynb` file to your **GitHub repository**.
   Ensure the repository link is in **HTTPS format** (e.g., `https://github.com/username/repository-name`).
5. Complete the **Google Form** by entering your required details and pasting your **GitHub repository HTTPS link**, then submit the form.

Submission Guidelines

- Your code must be **clean, well-structured, and properly organized**.
- Include **clear comments** explaining your logic wherever necessary.
- Only submissions with a valid **GitHub HTTPS link submitted through the Google Form** will be considered for evaluation.

Task 2: Feb Internship Logic Building Task

Python Programming Practice Problems

Problem 1: Unique Words in a Sentence

Problem Statement

Given a sentence, identify all unique words and display:

1. The count of unique words
2. The unique words themselves

Hint

- Split the sentence into individual words.
- Convert the list of words into a set to remove duplicates.
- Print both the set and its length.

Input

Enter sentence: python is easy and python is powerful

Expected Output

Unique words count: 5
Unique words: {'python', 'is', 'easy', 'and', 'powerful'}

Real-Time Applications

- Text analytics
- Natural Language Processing (NLP) preprocessing
- Search engine indexing
- Chat analysis systems

Problem 2: Highest Salary from Employee Data

Problem Statement

Store employee names and their salaries, and determine which employee has the highest salary.

Hint

- Use a dictionary where the key is the employee name and the value is the salary.
- Compare salary values to find the highest.

Input Format

A dictionary where:

- Keys represent employee names
- Values represent employee salaries

Example Input

```
employees = {
    "Ravi": 75000,
    "Anita": 68000,
    "Kiran": 72000
}
```

Expected Output

Highest Salary: Ravi - 75000

Real-Time Applications

- HR management systems
- Payroll analysis

Problem 3: Find Maximum and Minimum Values

Problem Statement

Find the maximum and minimum values from a list of numbers.

Hint

- Initialize both maximum and minimum with the first element of the list.
- Use a loop to compare each element.

Input Format

A list of integers where each element represents a numeric value.

Example Input

numbers = [45, 22, 89, 10, 66]

Expected Output

List: [45, 22, 89, 10, 66]
Max: 89
Min: 10

Real-Time Applications

- Sales data analysis
- Temperature monitoring systems
- Performance metric evaluation

Problem 4: Count Products Above a Price Threshold

Problem Statement

Count how many products have a price greater than 1000.

Hint

- Use a loop to compare each price with 1000.

Input Format

A list of integers where each element represents the price of a product.

Example Input

prices = [450, 1200, 899, 1500, 300]

Expected Output

Products above 1000: 2

Real-Time Applications

- E-commerce product filtering
- Budget analysis

Problem 5: Calculate Attendance Percentage

Problem Statement

Calculate the attendance percentage from a list containing "P" (Present) and "A" (Absent).

Hint

- Count the number of "P" values.
- Divide by the total number of days.

Example Input

attendance = ["P", "P", "A", "P", "P"]

Expected Output

Attendance Percentage: 80.0

Real-Time Applications

- Internship tracking systems
- HR attendance management systems

Problem 6: Remove Duplicate Phone Numbers

Problem Statement

You are given a list of phone numbers that may contain duplicates. Write a Python program to remove duplicate phone numbers and display only the unique phone numbers.

Hint

- Convert the list into a set.
- Sets automatically remove duplicate values.

Input Format

A list of integers where each integer represents a phone number.

Example Input

phone_numbers = [9876543210, 9123456789, 9876543210]

Expected Output

Unique phone numbers: {9876543210, 9123456789}

Real-Time Applications

- Contact management applications
- Customer Relationship Management (CRM) systems
- Marketing databases

Problem 7: Count Character Frequency

Problem Statement

Count how many times each character appears in a given string.

Hint

- Use a dictionary.
- Increment the count for each character occurrence.

Expected Output

{'p': 2, 'y': 1, 't': 1, 'h': 1, 'o': 1, 'n': 1}

Real-Time Applications

- Text analysis
- Password strength validation

Problem 8: Convert List to Tuple

Problem Statement

Convert a list into a tuple and display it.

Hint

- Use the `tuple()` function.

Expected Output

Tuple: (10, 20, 30)

Real-Time Applications

- Making data read-only
- Configuration settings storage

Problem 9: Check if a Key Exists in a Dictionary

Problem Statement

Check whether a specific key exists in a dictionary.

Hint

- Use the `in` keyword.

Expected Output

Employee exists

Real-Time Applications

- User validation systems
- Configuration verification

Problem 10: Calculate Average Marks

Problem Statement

Calculate the average marks of a student from a list of marks.

Hint

- Calculate the sum of all values.
- Divide by the total number of marks.

Expected Output

Average Marks: 72.5

Real-Time Applications

- Student performance evaluation
- Result processing systems