Python Coding Challenge – Detailed Documentation

Overview

This document explains a **multi-part Python coding challenge** that demonstrates core Python concepts such as variables, strings, lists, tuples, dictionaries, sets, loops, conditionals, functions, object-oriented programming, and basic user interaction.

The program is divided into multiple logical sections, each solving a small real-world problem. Together, they showcase beginner-to-intermediate Python skills.

---

1. Student Profile Creator

Purpose

To create and display a simple student profile using variables and string manipulation.

Code Summary

- Stores student name, age, and favourite subject
- Prints profile details
- Updates favourite subjects dynamically

Concepts Used

- Variables
- Strings
- String concatenation
- Print statements

Output Example

Name : Sree Lekha

Age : 27

Favourite Subjects : SQL

Updated Favourite Subjects

---

2. Feedback Analyzer

Purpose

To analyze user feedback text and extract useful insights.

Operations Performed

- Word count
- Count occurrences of the word "data"
- Extract first and last 5 characters
- Reverse the feedback string

Concepts Used

- String methods (split, lower, count)
- Slicing
- User input

Output Example

--- Feedback Analysis Report ---

Total words: 10

Occurrences of 'data': 2

First 5 characters: Hello

Last 5 characters: world

Reversed feedback: dlrow olleH

---

3. Retail Inventory Manager

Purpose

To manage a simple inventory system using lists and tuples.

Features

- Fixed product categories using tuples

- Dynamic product list using lists
- Add a new product
- Remove a product by index

Concepts Used

- Lists and tuples
- append() and pop()
- enumerate()
- Input validation

Output Example

Current Products:

0: Laptop

1: Pen

2: T-Shirt

---

4. Contact Directory

Purpose

To manage contact information using nested lists.

Features

- Store contacts (name, phone, email)
- Add and remove contacts
- Sort contact list alphabetically

Concepts Used

- Nested lists
- append() and pop()
- sort()

---

5. Student Grades & Hobbies Tracker

Purpose

To track student grades and hobbies using dictionaries and sets.

Features

- Update student grades
- Store unique hobbies using a set
- Convert set to immutable frozenset

Concepts Used

- Dictionaries
- Sets and frozensets
- Key-value updates

---

6. BMI Calculator

Purpose

To calculate Body Mass Index and classify health status.

Formula

BMI = weight / (height²)

Categories

- Underweight
- Normal weight
- Overweight
- Obese

Concepts Used

- Arithmetic operations
- Conditional statements

- Rounding values

---

7. Shopping Cart System

Purpose

To simulate a shopping cart with discount logic.

Features

- Add multiple item prices
- Stop input using sentinel value (0)
- Apply discounts based on subtotal

Discount Rules

- ≥ 1000 → 20%
- ≥ 500 → 10%
- ≥ 200 → 5%

Concepts Used

- While loop
- Break & continue
- Conditional logic

---

8. Employee (Student) Scores Report

Purpose

To generate and analyze random scores.

Features

- Generate random scores
- Display scores with index
- Filter high scores
- Calculate average

Concepts Used

- List comprehension
- enumerate()
- random module

---

9. Simple Calculator App

Purpose

To perform arithmetic operations using functions.

Supported Operations

- Addition (+)
- Subtraction (-)
- Multiplication (*)
- Division (/)
- Power (pow)
- Absolute (abs)
- Round

Concepts Used

- Functions
- Conditional branching
- Error handling (division by zero)

---

10. Bank Account Management System (OOP)

Purpose

To demonstrate Object-Oriented Programming in Python.

Features

- Create bank accounts
- Deposit and withdraw money
- Track total transactions using a global variable

Concepts Used

- Classes and objects
- Constructors (__init__)
- Methods
- Global variables

Output Example

Deposited $200 to Alice's account.

Withdrew $100 from Alice's account.

Total transactions across all accounts: 3

Conclusion

This coding challenge successfully demonstrates:

- Python fundamentals
- Data structures
- Control flow
- Functions
- Object-Oriented Programming
- Real-world problem solving

It is an excellent **portfolio project** for beginners and shows readiness for **Data Analyst / Python Developer entry-level roles**.

Recommended Improvements

- Convert the script into modular functions
- Add exception handling (try-except)

- Use menus for better user interaction
- Separate each module into different files

---

11. Consolidated Results / Sample Outputs

This section summarizes the **results produced by each module** when the program is executed. Actual values may vary based on user input and random generation.

Student Profile Creator – Result

- Name displayed: Sree Lekha
- Age displayed: 27
- Favourite Subjects updated from SQL to SQL, Python

Feedback Analyzer – Result

- Total words: Calculated from user input
- Occurrences of data: Case-insensitive count
- First 5 characters: Extracted from input
- Last 5 characters: Extracted from input
- Reversed feedback: Full string reversed

Retail Inventory Manager – Result

- Initial products listed with index
- New product added successfully
- Product removed based on valid index
- Final inventory displayed with fixed categories and updated product list

Contact Directory – Result

- Initial contacts printed
- New contact inserted
- Last contact removed
- User-entered contact added

- Contacts sorted in ascending (alphabetical) order

Student Grades & Hobbies Tracker – Result

- Grades updated for Meena and Arjun
- Unique hobbies stored in a set
- Hobbies converted to immutable frozenset

BMI Calculator – Result

- BMI value calculated and rounded to 2 decimals
- Health category displayed:
    - Underweight / Normal weight / Overweight / Obese

Shopping Cart System – Result

- Total items counted
- Subtotal calculated
- Discount rate applied based on subtotal
- Discount amount calculated
- Final payable total displayed

Scores Report – Result

- Random scores list generated
- Scores displayed with index
- Scores greater than 75 filtered
- Average score calculated

Simple Calculator App – Result

- Operation performed based on user choice
- Correct result displayed
- Division by zero handled safely

Bank Account Management System – Result

- Deposits and withdrawals processed

- Account balances updated correctly
- Insufficient balance handled
- Total transactions tracked globally
