CHAPTER 1 – PRACTICE SET

1. Write a program to print Twinkle twinkle little star poem in python.
2. Use REPL and print the table of 5 using it.
3. Install an external module and use it to perform an operation of your interest.
4. Write a python program to print the contents of a directory using the os module. Search online for the function which does that.
5. Label the program written in problem 4 with comments.

CHAPTER 2 – PRACTICE SET

1. Write a python program to add two numbers.
2. Write a python program to find remainder when a number is divided by z.
3. Check the type of variable assigned using input () function.
4. Use comparison operator to find out whether ‘a’ given variable a is greater than ‘b’ or not. Take a = 34 and b = 80
5. Write a python program to find an average of two numbers entered by the user.
6. Write a python program to calculate the square of a number entered by the user.

CHAPTER 3 – PRACTICE SET

1. Write a python program to display a user entered name followed by Good Afternoon using input () function.
2. Write a program to fill in a letter template given below with name and date.

```
letter = '''
       Dear <|Name|>,
       You are selected!
       <|Date|>
        '''
```

3. Write a program to detect double space in a string.
4. Replace the double space from problem 3 with single spaces.
5. Write a program to format the following letter using escape sequence characters.

```
letter = "Dear Harry, this python course is nice. Thanks!"
```

CHAPTER 4 - PRACTICE SET

1. Write a program to store seven fruits in a list entered by the user.
2. Write a program to accept marks of 6 students and display them in a sorted manner.
3. Check that a tuple type cannot be changed in python.
4. Write a program to sum a list with 4 numbers.
5. Write a program to count the number of zeros in the following tuple:

```
a = (7, 0, 8, 0, 0, 9)
```

CHAPTER 5 – PRACTICE SET

1. Write a program to create a dictionary of Hindi words with values as their English translation. Provide user with an option to look it up!
2. Write a program to input eight numbers from the user and display all the unique numbers (once).
3. Can we have a set with 18 (int) and '18' (str) as a value in it?
4. What will be the length of following set s:

```
s = set()
s.add(20)
s.add(20.0)
s.add('20') # length of s after these operations?
```

5. s = {}

   What is the type of 's'?
6. Create an empty dictionary. Allow 4 friends to enter their favorite language as value and use key as their names. Assume that the names are unique.
7. If the names of 2 friends are same; what will happen to the program in problem 6?
8. If languages of two friends are same; what will happen to the program in problem 6?
9. Can you change the values inside a list which is contained in set S?

```
s = {8, 7, 12, "Harry", [1,2]}
```

CHAPTER 6 – PRACTICE SET

1. Write a program to find the greatest of four numbers entered by the user.
2. Write a program to find out whether a student has passed or failed if it requires a total of 40% and at least 33% in each subject to pass. Assume 3 subjects and take marks as an input from the user.
3. A spam comment is defined as a text containing following keywords:
   "Make a lot of money", "buy now", "subscribe this", "click this". Write a program to detect these spams.
4. Write a program to find whether a given username contains less than 10 characters or not.
5. Write a program which finds out whether a given name is present in a list or not.
6. Write a program to calculate the grade of a student from his marks from the following scheme:
   90 – 100 => Ex
   80 – 90 => A
   70 – 80 => B
   60 – 70 =>C
   50 – 60 => D
   <50 => F
7. Write a program to find out whether a given post is talking about "Harry" or not.

CHAPTER 7 – PRACTICE SET

1. Write a program to print multiplication table of a given number using for loop.
2. Write a program to greet all the person names stored in a list ‘l’ and which starts with S.
   l = ["Harry", "Soham", "Sachin", "Rahul"]
3. Attempt problem 1 using while loop.
4. Write a program to find whether a given number is prime or not.
5. Write a program to find the sum of first n natural numbers using while loop.
6. Write a program to calculate the factorial of a given number using for loop.
7. Write a program to print the following star pattern.
   ```
     *
    ***
   *****  for n = 3
   ```
8. Write a program to print the following star pattern:
   ```
   *
   **
   ***    for n = 3
   ```
9. Write a program to print the following star pattern.
   ```
   * * *
   *   *   for n = 3
   * * *
   ```
10. Write a program to print multiplication table of n using for loops in reversed order.

CHAPTER 8 – PRACTICE SET

1. Write a program using functions to find greatest of three numbers.
2. Write a python program using function to convert Celsius to Fahrenheit.
3. How do you prevent a python print() function to print a new line at the end.
4. Write a recursive function to calculate the sum of first n natural numbers.
5. Write a python function to print first n lines of the following pattern:

```
***
**          - for n = 3
*
```

6. Write a python function which converts inches to cms.
7. Write a python function to remove a given word from a list ad strip it at the same time.
8. Write a python function to print multiplication table of a given number.

PROJECT 1: SNAKE, WATER, GUN GAME

We all have played snake, water gun game in our childhood. If you haven’t, google the rules of this game and write a python program capable of playing this game with the user.

CHAPTER 9 – PRACTICE SET

1. Write a program to read the text from a given file ‘poems.txt’ and find out whether it contains the word ‘twinkle’.
2. The game() function in a program lets a user play a game and returns the score as an integer. You need to read a file ‘Hi-score.txt’ which is either blank or contains the previous Hi-score. You need to write a program to update the Hi-score whenever the game() function breaks the Hi-score.
3. Write a program to generate multiplication tables from 2 to 20 and write it to the different files. Place these files in a folder for a 13 – year old.
4. A file contains a word “Donkey” multiple times. You need to write a program which replace this word with ##### by updating the same file.
5. Repeat program 4 for a list of such words to be censored.
6. Write a program to mine a log file and find out whether it contains ‘python’.
7. Write a program to find out the line number where python is present from ques 6.
8. Write a program to make a copy of a text file “this. txt”
9. Write a program to find out whether a file is identical & matches the content of another file.
10. Write a program to wipe out the content of a file using python.
11. Write a python program to rename a file to “renamed_by_ python.txt.

CHAPTER 10 – PRACTICE SET

1. Create a class *“Programmer”* for storing information of few programmers working at Microsoft.
2. Write a class *“Calculator”* capable of finding square, cube and square root of a number.
3. Create a class with a class attribute a; create an object from it and set ‘a’ directly using ‘object.a = 0’. Does this change the class attribute?
4. Add a static method in problem 2, to greet the user with hello.
5. Write a Class ‘Train’ which has methods to book a ticket, get status (no of seats) and get fare information of train running under Indian Railways.
6. Can you change the self-parameter inside a class to something else (say “harry”). Try changing self to “slf” or “harry” and see the effects.

CHAPTER 11- PRACTICE SET

1. Create a class (2-D vector) and use it to create another class representing a 3-D vector.

2. Create a class 'Pets' from a class 'Animals' and further create a class 'Dog' from 'Pets'. Add a method 'bark' to class 'Dog'.

3. Create a class 'Employee' and add salary and increment properties to it.

Write a method 'salaryAfterIncrement' method with a @property decorator with a setter which changes the value of increment based on the salary.

4. Write a class 'Complex' to represent complex numbers, along with overloaded operators '+' and '*' which adds and multiplies them.

5. Write a class vector representing a vector of n dimensions. Overload the + and * operator which calculates the sum and the dot(.) product of them.

6. Write __str__() method to print the vector as follows:

```
            7i + 8j +10k
```

Assume vector of dimension 3 for this problem.

7. Override the __len__() method on vector of problem 5 to display the dimension of the vector.

PROJECT 2 – THE PERFECT GUESS

We are going to write a program that generates a random number and asks the user to guess it.

If the player’s guess is higher than the actual number, the program displays “Lower number please”. Similarly, if the user’s guess is too low, the program prints “higher number please” When the user guesses the correct number, the program displays the number of guesses the player used to arrive at the number.

Hint: *Use the random module*.

CHAPTER 12 – PRACTICE SET

1. Write a program to open three files 1.txt, 2.txt and 3.txt if any these files are not present, a message without exiting the program must be printed prompting the same.

2. Write a program to print third, fifth and seventh element from a list using enumerate function.

3. Write a list comprehension to print a list which contains the multiplication table of a user entered number.

4. Write a program to display a/b where a and b are integers. If b=0, display infinite by handling the ‘ZeroDivisionError’.

5. Store the multiplication tables generated in problem 3 in a file named Tables.txt.

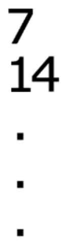
CHAPTER 13- PRACTICE SET

1. Create two virtual environments, install few packages in the first one. How do you create a similar environment in the second one?

2. Write a program to input name, marks and phone number of a student and format it using the format function like below:

“The name of the student is Harry, his marks are 72 and phone number is 99999888”

3. A list contains the multiplication table of 7. write a program to convert it to vertical string of same numbers.



```
7
14
.
.
.
```

4. Write a program to filter a list of numbers which are divisible by 5.

5. Write a program to find the maximum of the numbers in a list using the reduce function.

6. Run pip freeze for the system interpreter. Take the contents and create a similar virtualenv.

7. Explore the ‘Flask’ module and create a web server using Flask & Python.

MEGA PROJECT 1: JARVIS - VOICE-ACTIVATED VIRTUAL ASSISTANT

Jarvis is a voice-activated virtual assistant designed to perform tasks such as web browsing, playing music, fetching news, and responding to user queries using OpenAI's GPT-3.5-turbo model.

FEATURES

- Voice Recognition
- Utilizes the speech_recognition library to listen for and recognize voice commands.
- Activates upon detecting the wake word "Jarvis."
- Text-to-Speech
- Converts text to speech using pyttsx3 for local conversion.
- Uses gTTS (Google Text-to-Speech) and pygame for playback.
- Web Browsing.
- Opens websites like Google, Facebook, YouTube, and LinkedIn based on voice commands.
- Music Playback
- Interfaces with a musicLibrary module to play songs via web links.
- News Fetching
- Fetches and reads the latest news headlines using NewsAPI.
- OpenAI Integration
- Handles complex queries and generates responses using OpenAI's GPT-3.5-turbo.
- Acts as a general virtual assistant similar to Alexa or Google Assistant.
- Activates upon detecting the wake word "Jarvis."
- Text-to-Speech

WORKFLOW

1. Initialization
2. Greets the user with "Initializing Jarvis...."
3. Wake Word Detection
4. Listens for the wake word "Jarvis."
5. Acknowledges activation by saying "Ya."
6. Command Processing.
7. Processes commands to determine actions such as opening a website, playing music, fetching news, or generating a response via OpenAI.
8. Speech Output.
9. Provides responses using speak function with either pyttsx3 or gTTS.
10. Greets the user with "Initializing Jarvis...."
11. Wake Word Detection
12. Acknowledges activation by saying "Ya."

MEGA PROJECT 2: AUTO-REPLY AI CHATBOT

DESCRIPTION

This project automates the process of interacting with a chat application, specifically designed to analyze chat history and generate humorous responses using OpenAI's GPT-3.5-turbo model. The virtual assistant, named Naruto, is a character that roasts people in a funny way, based on the chat history.

FEATURES

14. Automated Chat Interaction
15. Uses pyautogui to perform mouse and keyboard operations, interacting with the chat application without manual intervention.
16. Chat History Analysis
17. Copies chat history from the chat application and analyzes it to determine if the last message was sent by a specific user (e.g., "Rohan Das").
18. Humorous Response Generation
19. Integrates with OpenAI's GPT-3.5-turbo model to generate funny, roast-style responses based on the analyzed chat history.
20. Clipboard Operations
21. Utilizes pyperclip to copy and paste text, facilitating the retrieval and insertion of chat messages.
22. Uses pyautogui to perform mouse and keyboard operations, interacting with the chat application without manual intervention.
23. Copies chat history from the chat application and analyzes it to determine if the last message was sent by a specific user (e.g., "Rohan Das").
24. Humorous Response Generation
25. Integrates with OpenAI's GPT-3.5-turbo model to generate funny, roast-style responses based on the analyzed chat history.

WORKFLOW

- Initialization and Setup
- Click on the Chrome icon to open the chat application.
- Wait for a brief period to ensure the application is open and ready for interaction.
- Chat History Retrieval
- Periodically select and copy chat history by dragging the mouse over the chat area and using the copy shortcut.
- Retrieve the copied text from the clipboard.
- Message Analysis