List



* When we want to store no of values in one variable, we can take List.
* A list in Python is used to store the sequence of various types of data.
* List items are ordered, changeable, and allow duplicate values.
* List items are indexed, the first item has index [0], the second item has index [1] etc.

mylist = ["apple", "banana", "cherry"]

nos=[11,33,4,55,66,77,88]

mixdata=[11,”ram”,33,”raj”,”Simran”,55]

Exercise

1. Write a Python program to sum all the items in a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 1021

Hint: Use the built-in sum() function or a loop to iterate through the list and add all values.

2. Write a Python program to print only even values from a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 44 22 200 66 2

Hint: Use a loop and the modulus operator % to check if a number is even.

3. Write a Python program to print whether the values in a list are odd or even.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result:

11 is odd

44 is even

500 is even

22 is even

99 is odd

77 is odd

200 is even

66 is even

Hint: Loop through the list and use the modulus operator % to determine odd or even status.

4. Write a Python program to get the largest number from a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 500

Hint: Use the max() function to find the largest number.

5. Write a Python program to get the smallest number from a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 2

Hint: Use the min() function to find the smallest number.

6. Write a Python program to add the first and last value of the list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 13

Hint: Access the first value using list[0] and the last value using list[-1], then add them.

7. Write a Python program to print a list in sorted order.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 2 11 22 44 66 77 99 200 500

Hint: Use the sort() function to sort the list and then print the result.

8. Write a Python program to print a list in reverse order.

Sample List: [11, 44, 500, 22]

Expected Result: 22, 500, 44, 11

Hint: Use the reverse() method to reverse the list.

9. Write a Python program to print a list in ascending and descending order.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result:

Ascending: 2 11 22 44 66 77 99 200 500

Descending: 500 200 99 77 66 44 22 11 2

Hint: Use the sort() and reverse() functions.

10. Write a Python program to remove duplicates from a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2, 11, 22]

Expected Result: [500, 99, 77, 200, 66, 2, 11, 22]

11. Write a Python program to check if a list is empty and display the total number of elements.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2, 11, 22]

Expected Result: There are 11 elements

Hint: Use the len() function to find the number of elements, and check if len(list) == 0 to determine if it's empty.

12. Write a Python program to clone or copy a list.

Sample List: [11, 44, 500]

Expected Result: [11, 44, 500]

13. Write a Python program to find elements larger than a given value in a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2, 11, 22]

Input: Enter greater than value: 50

Expected Result: 500 99 77 200 66

Hint: Use a loop or list comprehension to filter numbers greater than the given value.

14. Write a Python program to change the sign of elements in a list.

Sample List: [11, -44, 500, -22, -99, -77, 200, -66, 2]

Expected Result: [-11, 44, -500, 22, 99, 77, -200, 66, -2]

15. Write a Python program to print a list after removing the 0th, 4th, and 5th elements.

Sample List: ['Red', 'Green', 'White', 'Black', 'Pink', 'Yellow']

Expected Output: ['Green', 'White', 'Black']

Hint: Use the del keyword or slicing to remove specific elements by index.

16. Write a Python program to remove even numbers from a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2, 11, 22]

Expected Result: [11,99, 77, 11]

Hint: Use a loop or list comprehension to filter out even numbers by checking if number % 2 == 0.

17. Write a Python program to take a value from the user and add it at a specified index.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2, 11, 22]

Input: Enter position: 2, Enter value: 999

Expected Result: [11, 999, 44, 500, 22, 99, 77, 200, 66, 2, 11, 22]

Hint: Use the insert() method to add the value at the specified index.

18. Write a Python program to append one list to another list.

Sample List 1: [11, 44, 500, 22, 99]

Sample List 2: [77, 200, 66, 2, 11, 22]

Expected Result: [11, 44, 500, 22, 99, 77, 200, 66, 2, 11, 22]

Hint: Use the extend() method or + operator to combine the two lists.

19. Write a Python program to find common items from two lists.

Sample List 1: [11, 44, 500]

Sample List 2: [77, 44, 11]

Expected Result: [11, 44]

Hint: Use loop to find common elements.

20. Write a Python program to check if the first and last numbers in a list are the same.

Sample List 1: [100, 200, 320, 40, 100]

Sample List 2: [751, 6, 3, 5, 9]

Expected Result: True for the first list, False for the second list

Hint: Compare list[0] with list[-1] to check if the first and last numbers are the same.

21. Write a Python program to square each element in a list.

Sample List: [11, 2, 4, 3, 6, 7]

Expected Result: [121, 4, 16, 9, 36, 49]

Hint: Use a loop or list comprehension to square each number using number \*\* 2.

22. Write a Python program to remove empty strings from a list.

Sample List: ["Raj", "", "Rahul", "Mansi", "", "Manav", "Disha"]

Expected Result: ["Raj", "Rahul", "Mansi", "Manav", "Disha"]

Hint: Use a loop .

23. Write a Python program to calculate the product of all items in a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 453752160000

24. Write a Python program to find the second largest number in a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 200

Hint: Sort the list and get the second-to-last element or use a loop to track the two largest numbers.

25. Write a Python program to count the number of even and odd numbers in a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result:

Even numbers: 6

Odd numbers: 3

Hint: Use a loop and the modulus operator % to check if each number is even or odd.

26. Write a Python program to swap the first and last element of a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: [2, 44, 500, 22, 99, 77, 200, 66, 11]

Hint: Swap list[0] and list[-1] directly.

27. Write a Python program to find the index of the maximum value in a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: Index of max value (500) is 2

Hint: Use max() to find the maximum value, and list.index() to find its index.

28. Write a Python program to find all unique elements from a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2, 11, 22]

Expected Result: [44, 500, 99, 77, 200, 66, 2]

29. Using the following list of programming languages per paradigm:

procedural = ["c", "fortran", "pascal"]

object\_oriented = ["java", "c++", "python"]

functional = ["haskell", "scala", "lisp"]

Write a program that asks the user to enter a programming language and tells them which paradigm it belongs to.

30. Using following list of cities per country,

india = ["mumbai", "banglore", "chennai", "delhi"]

pakistan = ["lahore","karachi","islamabad"]

bangladesh = ["dhaka", "khulna", "rangpur"]

Do the following task:

1. Write a program that asks user to enter a city name and it should tell which country the city belongs to
2. Write a program that asks user to enter two cities and it tells you if they both are in same country or not. For example if I enter mumbai and chennai, it will print "Both cities are in India" but if I enter mumbai and dhaka it should print "They don't belong to same country"

31. You have a list of your favourite marvel super heroes.

heros=['spider man','thor','hulk','iron man','captain america']

Using this find out,

1. Length of the list

2. Add 'black panther' at the end of this list

3. You realize that you need to add 'black panther' after 'hulk',

so remove it from the list first and then add it after 'hulk'

4. Now you don't like thor and hulk because they get angry easily :)

So you want to remove thor and hulk from list and replace them with doctor strange (because he is cool).

32. You have a list of your favourite desserts.

desserts = ['cheesecake', 'brownie', 'ice cream', 'tiramisu', 'pavlova']

Using this list, find out:

1. How many desserts do you have in your list?
2. Add 'macaron' at the starting of the list and show it.
3. You want 'ice cream' to be left after 'pavlova'. Remove it from the end and add it at the correct position.

33. After throwing the dice several times, you got this result,

dice\_result = [5,6,4,2,5,4,4,5,3,3,2,6,1,2,1,1,6,5]

Using a for loop find out the followings:

How many times have you got 6s

34. Write a Python program to count the number of strings in a list that have a length greater than 3.

Sample List: ["cat", "dog", "elephant", "rat", "hippopotamus", "fox"]

Expected Result: 2 (elephant and hippopotamus have length > 3)

Hint: Use a loop or list with len() to check the length of each string.

35. Write a Python program to replace all negative numbers in a list with zero.

Sample List: [11, -44, 500, -22, -99, 77, 200, -66, 2]

Expected Result: [11, 0, 500, 0, 0, 77, 200, 0, 2]

Hint: Use a loop or list check if each number is less than 0 and replace it with 0.

36. Write a Python program to create a new list containing only the first half of elements from the original list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: [11, 44, 500, 22] (first 4 elements since 9/2 ≈ 4)

Hint: Use slicing with len(list)//2 to get the first half.

37. Write a Python program to find the sum of all numbers divisible by 3 in a list.

Sample List: [11, 12, 15, 22, 99, 77, 200, 66, 2]

Expected Result: 192 (12 + 15 + 99 + 66 = 192)

Hint: Use a loop and the modulus operator % to check divisibility by 3, then add those numbers.

38. Write a Python program to check if all elements in a list are positive.

Sample List 1: [11, 44, 500, 22]

Sample List 2: [11, -44, 500, 22]

Expected Result: True for first list, False for second list

Hint: Use a len() and count=count+1

39. Write a Python program to split a list into two lists: one with even indices and one with odd indices.

Sample List: [11, 44, 500, 22, 99, 77]

Expected Result:

Even indices: [11, 500, 99] (indices 0, 2, 4)

Odd indices: [44, 22, 77] (indices 1, 3, 5)

Hint: Use a loop and i %2

40. Write a Python program to find the average of all numbers in a list.

Sample List: [11, 44, 500, 22, 99, 77, 200, 66, 2]

Expected Result: 113.44 (sum = 1021, count = 9, 1021/9 ≈ 113.44)

Hint: Use sum() and len() to calculate the average (sum/length).

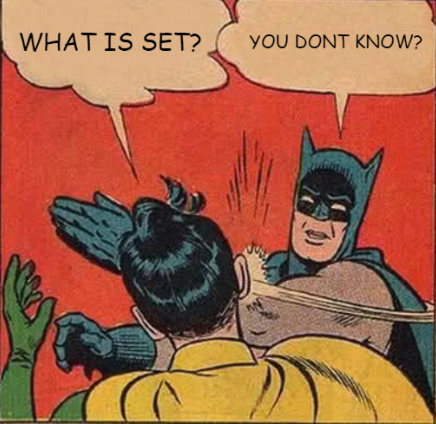
41. Write a Python program to count how many times a specific value appears in a list.

Sample List: [11, 44, 500, 22, 99, 11, 22, 66, 2]

Input: Enter value to count: 11

Expected Result: 2 (11 appears twice)

Hint: Use the count() method or a loop to count occurrences of the input value.



* Sets are used to store multiple items in a single variable.
* Usually sets are used for mathematical operations like union,intersection,etc.
* In set duplicates value are not allowed.

thisset = {"apple", "banana", "cherry"}

s1={11,22,33,44,55}

Exercise

1. Write a Python program to create a set.

Sample data: {11, 22, 33, 44, 55}

Expected Output: {11, 22, 33, 44, 55}

2. Write a Python program to print all the items in a set on new lines.

Hint: Use a for loop to iterate over the set and print each item.

Sample data: {11, 22, 33, 44, 55}

Expected Output:

11

22

33

44

55

3. Write a Python program to add member(s) to a set.

Hint: Use the add() method to add a new element to the set.

Sample data: {11, 22, 33, 44, 55}

Enter value: 66

Expected Output: {11, 22, 33, 44, 55, 66}

4. Write a Python program to remove an item from a set if it is present.

Hint: Use the discard() or remove() method to remove an element.

Sample data: {11, 22, 33, 44, 55}

Enter value: 33

Expected Output: {11, 22, 44, 55}

5. Write a Python program to create an intersection of sets.

Hint: Use the intersection() method or & operator to find common elements between two sets.

Sample data:

s1 = {11, 22, 33, 44, 55}

s2 = {11, 22, 99, 66, 55}

Expected Output: {11, 22, 55}

6. Write a Python program to create a union of sets.

Hint: Use the union() method or | operator to combine two sets.

Sample data:

s1 = {11, 22, 33, 44, 55}

s2 = {11, 22, 99, 66, 55}

Expected Output: {11, 22, 33, 44, 55, 99, 66}

7. Write a Python program to create a set difference.

Hint: Use the difference() method or - operator to find elements in one set that are not in another.

Sample data:

s1 = {11, 22, 33, 44, 55}

s2 = {11, 22, 99, 66, 55}

Expected Output: {33, 44}

8. Write a Python program to check if a set is a subset of another set.

Hint: Use the issubset() method to check if all elements of one set are present in another.

Sample data:

s1 = {11, 22, 33, 44, 55}

s2 = {11, 22, 55}

Expected Output: True

9.Write a Python program to clear a set.

Hint: Use the clear() method to remove all elements from the set.

Sample data:

s1 = {11, 22, 33, 44, 55}

Logic: s1.clear()

Expected Output: set()

10. Write a Python program to find the maximum and minimum values in a set.

Hint: Use Python's max() and min() functions to find the largest and smallest values.

Sample data:

s1 = {11, 22, 33, 44, 55}

Expected Output:

Max value is 55

Min value is 11

11. Write a Python program to find the length of a set.

Hint: Use the len() function to find the number of elements in a set.

Sample data: {11, 22, 33, 44, 55}

Expected Output: 5

12.Write a Python program to check if an element is present in a set.

Hint: Use the in keyword to check if an element exists in the set.

Sample data: {11, 22, 33, 44, 55}

Enter value: 22

Expected Output: True

13. Write a Python program to find the symmetric difference between two sets.

Sample data:

s1 = {11, 22, 33, 44, 55}

s2 = {11, 22, 99, 66, 55}

Expected Output: {33, 44, 99, 66}

14. Write a Python program to check if two sets have any common elements.

Hint: Use the isdisjoint() method, which returns True if sets have no elements in common.

Sample data:

s1 = {11, 22, 33}

s2 = {44, 55, 66}

Expected Output: True

15. Write a Python program to find the difference between two sets.

Hint: Use the difference() method or - operator to get elements that are in one set but not the other.

Sample data:

s1 = {11, 22, 33, 44, 55}

s2 = {33, 44}

Expected Output: {11, 22, 55}

16. Write a Python program to find the union of multiple sets.

Hint: Use the union() method or | operator to combine all elements from multiple sets.

Sample data:

s1 = {11, 22}

s2 = {33, 44}

s3 = {55, 66}

Expected Output: {11, 22, 33, 44, 55, 66}

17. Write a Python program to remove all elements from a set.

Hint: Use the clear() method to empty the set.

Sample data:

s1 = {11, 22, 33}

Expected Output: set()

18. Write a Python program to check if one set is a superset of another.

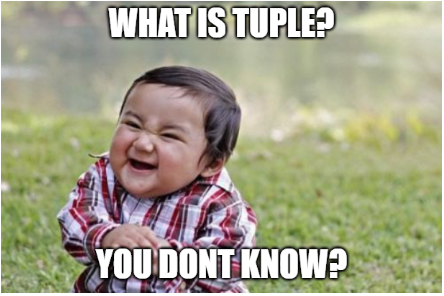
Hint: Use the issuperset() method to check if all elements of another set are contained in the current set.

Sample data:

s1 = {11, 22, 33, 44, 55}

s2 = {11, 22}

Expected Output: True



A tuple is created by placing all the items (elements) inside parentheses (), separated by commas. The parentheses are optional; however, it is a good practice to use them.

Exercise

1. Write a Python program to create a tuple.

Hint: Tuples are created by placing values inside parentheses ().

Sample data:

t1 = (11, 22, 33, 44, 55)

Expected Output: (11, 22, 33, 44, 55)

2. Write a Python program to create a tuple with different data types.

Hint: Tuples can hold values of different data types, such as integers, strings, and floats.

Sample data:

t1 = (11, 22, "ram", 44, 55.66, "rahul")

Expected Output: (11, 22, 'ram', 44, 55.66, 'rahul')

3. Write a Python program to create a tuple with numbers and print one item.

Hint: You can access tuple elements using their index, starting from 0.

Sample data:

t1 = (11, 22, 33, 44, 55)

index=int(input("Enter index =>"))

Expected Output: 33 (for printing the 3rd item)

4. Write a Python program to add an item in a tuple.

Hint: Tuples are immutable, so to add an item, convert the tuple to a list, add the item, then convert it back to a tuple.

Sample data:

t1 = (11, 22, 33, 44, 55)

Enter data: 100

Expected Output: (11, 22, 33, 44, 55, 100)

5. Write a Python program to get the 4th element from last of a tuple.

Hint: You can access tuple elements using positive and negative indices.

Sample data:

t1 = (11, 22, 33, 44, 55)

Expected Output: 44 (for both 4th element and 4th from last)

6. Write a Python program to check whether an element exists within a tuple.

Hint: Use the in keyword to check if an element is present in a tuple.

Sample data:

t1 = (11, 22, 33, 44, 55)

Enter value for search: 44

Expected Output: Yes, 44 is there

7. Write a Python program to convert a list to a tuple.

Hint: Use the tuple() function to convert a list to a tuple.

Sample data:

list1 = [11, 22, 33, 44, 55]

Expected Output: (11, 22, 33, 44, 55)

8. Write a Python program to remove an item from a tuple.

Hint: Since tuples are immutable, convert the tuple to a list, remove the item, then convert it back to a tuple.

Sample data:

t1 = (11, 22, 33, 44, 55)

Enter value for delete: 22

Expected Output: (11, 33, 44, 55)

9. Write a Python program to find the index of an item in a tuple.

Hint: Use the index() method to find the position of an element in a tuple.

Sample data:

t1 = (11, 22, 33, 44, 55)

Enter value to find its index: 22

Expected Output: 1

10. Write a Python program to find the length of a tuple.

Hint: Use the len() function to find the number of elements in a tuple.

Sample data:

t1 = (11, 22, 33, 44, 55)

Expected Output: 5

11. Write a Python program to reverse a tuple.

Sample data:

t1 = (11, 22, 33, 44, 55)

Expected Output: (55, 44, 33, 22, 11)

12. Write a Python program to slice a tuple.

Hint: Use slicing syntax tuple[start:end] to extract a portion of the tuple.

Sample data:

t1 = (11, 22, 33, 44, 55, 66, 77)

Slice from index 2 to 5

Expected Output: (33, 44, 55)

13. Write a Python program to unpack a tuple into several variables.

Hint: Use variable unpacking to assign tuple elements to variables.

Sample data:

t1 = (11, 22, 33)

Expected Output:

a = 11, b = 22, c = 33

14. Write a Python program to count the occurrences of an item in a tuple.

Hint: Use the count() method to find how many times an element appears.

Sample data:

t1 = (11, 22, 33, 44, 33, 22, 33)

Enter value to count: 33

Expected Output: 3

15. Write a Python program to multiply all elements in a tuple by 2.

Hint: Use a list comprehension or a loop to iterate through the tuple and multiply each element, then convert it back to a tuple.

Sample data:

t1 = (1, 2, 3, 4)

Expected Output: (2, 4, 6, 8)

15. Write a Python program to find the largest and smallest items in a tuple.

Hint: Use the max() and min() functions.

Sample data:

t1 = (11, 22, 33, 44, 55)

Expected Output:

Max: 55

Min: 11

16. Write a Python program to sort a tuple of numbers in ascending order.

Hint: Convert the tuple to a list, sort the list, and convert it back to a tuple.

Sample data:

t1 = (55, 22, 33, 11, 44)

Expected Output: (11, 22, 33, 44, 55)

17. Write a Python program to merge two tuples.

Hint: Use the + operator to concatenate two tuples.

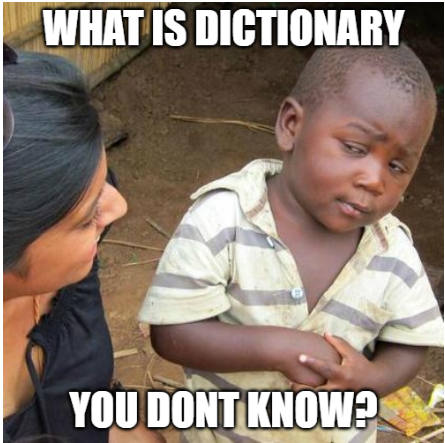
Sample data:

t1 = (11, 22, 33)

t2 = (44, 55, 66)

Expected Output: (11, 22, 33, 44, 55, 66)

18. Create a list of your friends' names and now create a list of tuples. The tuple should contain the friend’s name and the length of the name. For Example: if someone’s name is Aditya, the tuple would be: (‘Aditya’, 6)



Creating a dictionary is as simple as placing items inside curly braces {} separated by commas.

An item has a key and a corresponding value that is expressed as a pair (key: value).

Dictionaries are used to store data values in key:value pairs.

shoes = {

"adidas": 1200,

"nike": 1800,

"sketchers": 3000

}

Exercise

1. Write a Python script to store value in a dictionary and print.

Sample data:

marks = {

"ram": 33,

"rahul": 45,

"devesh": 30,

"jayul": 34,

"meena": 29,

"nisha": 37,

"karan": 40,

"anita": 18,

"siddhi": 25

}

Expected Output:

{"ram": 33, "rahul": 45, "devesh": 30, "jayul": 34, "meena": 29, "nisha": 37, "karan": 40, "anita": 18, "siddhi": 25}

Hint: Use curly braces {} to define a dictionary and print() to display it.

2. Write a Python script to print data in vertical form from a dictionary.

Sample data:

marks = {

"ram": 33,

"rahul": 45,

"devesh": 30,

"jayul": 34,

"meena": 29,

"nisha": 37,

"karan": 40,

"anita": 18,

"siddhi": 25

}

Expected Output:

ram 33

rahul 45

devesh 30

jayul 34

meena 29

nisha 37

karan 40

anita 18

siddhi 25

Hint: Use a loop to iterate through the dictionary items and print each key-value pair.

3. Write a Python script to check whether a given key already exists in a dictionary.

Sample data:

marks = {

"ram": 33,

"rahul": 45,

"manav": 30,

"jayul": 34,

"meena": 29,

"siddhi": 48

}

Enter key for search: rahul

Output:

Yes, it exists.

Hint: Use the in keyword to check if the key is present in the dictionary.

4. Write a Python script to print data in vertical form and display whether that student is pass or fail from the dictionary. (18 marks to pass)

Sample data:

marks = {

"ram": 33,

"rahul": 15,

"devesh": 30,

"jayul": 34,

"jiya": 16,

"sadhana": 11,

"meena": 19,

"karan": 20

}

Expected Output:

name mark result

ram 33 pass

rahul 15 fail

devesh 30 pass

jayul 34 pass

jiya 16 fail

sadhana 11 fail

meena 19 pass

karan 20 pass

Hint: Use a loop and a conditional statement to determine if each student passed or failed.

5. Write a Python script to print data in vertical form and display only pass students from the dictionary. (18 marks to pass)

Sample data:

marks = {

"ram": 33,

"rahul": 15,

"devesh": 30,

"jayul": 34,

"jiya": 16,

"sadhana": 11,

"meena": 19,

"karan": 20

}

Expected Output:

name mark result

ram 33 pass

devesh 30 pass

jayul 34 pass

meena 19 pass

karan 20 pass

Hint: Filter the results based on the passing criteria while printing.

6. Write a Python program to sum all the items in a dictionary.

Sample data:

items = {

"maggie": 20,

"parleg": 10,

"crackjack": 20,

"noodles": 32,

"chips": 15,

"cookies": 18

}

Expected Output:

125 Rs

Hint: Use the sum() function combined with values() to calculate the total.

7. Write a Python program to remove a key from a dictionary.

Sample data:

marks = {

"ram": 33,

"rahul": 15,

"devesh": 30,

"jayul": 34,

"jiya": 16,

"sadhana": 11,

"meena": 19

}

Enter key to delete: rahul

Expected Output:

marks = {"ram": 33, "devesh": 30, "jayul": 34, "jiya": 16, "sadhana": 11, "meena": 19}

Hint: Use the del statement or the pop() method to remove the key from the dictionary.

8. Write a Python program to find the maximum and minimum marks from a dictionary.

Sample data:

marks = {

"ram": 33,

"rahul": 45,

"devesh": 30,

"jayul": 34,

"meena": 29,

"karan": 40,

"anita": 18,

"siddhi": 25

}

Expected Output:

Max: 45, Min: 18

Hint: Use max() and min() functions on values() to find the highest and lowest marks.

9. Write a Python program to count the number of students who passed (marks >= 18) in the dictionary.

Sample data:

marks = {

"ram": 33,

"rahul": 15,

"devesh": 30,

"jayul": 34,

"jiya": 16,

"sadhana": 11,

"meena": 19,

"karan": 20,

"anita": 25

}

Expected Output:

Number of passed students: 5

Hint: Use a loop and a counter variable to keep track of the number of students who meet the passing criteria.

10. Write a Python program to update a dictionary with new student marks.

Sample data:

marks = {

"ram": 33,

"rahul": 45

}

Enter new data:

devesh 30

Expected Output:

{"ram": 33, "rahul": 45, "devesh": 30}

Hint: Use the syntax dict[key] = value to add new key-value pairs.

11. Write a Python program to get a list of all keys in a dictionary.

Sample data:

marks = {

"ram": 33,

"rahul": 45,

"devesh": 30,

"jayul": 34

}

Expected Output:

["ram", "rahul", "devesh", "jayul"]

Hint: Use the keys() method to obtain the keys and convert them to a list if needed.

12. Write a Python program to get a list of all values in a dictionary.

Sample data:

marks = {

"ram": 33,

"rahul": 45,

"devesh": 30,

"jayul": 34

}

Expected Output:

[33, 45, 30, 34]

Hint: Use the values() method to get the values and convert them to a list if needed.

13. Write a Python program to print only the students who failed from the dictionary (marks < 18).

Sample data:

marks = {

"ram": 33,

"rahul": 15,

"devesh": 30,

"jayul": 34,

"jiya": 16,

"sadhana": 11,

"meena": 19,

"karan": 20

}

Expected Output:

name mark result

rahul 15 fail

jiya 16 fail

sadhana 11 fail

Hint: Use a loop and conditional statements to filter and display students who failed.

14. Write a Python program to get the length of a dictionary.

Sample data:

marks = {

"ram": 33,

"rahul": 45,

"devesh": 30,

"jayul": 34,

"jiya": 16

}

Expected Output:

Length of dictionary: 5

Hint: Use the len() function to count the number of key-value pairs in the dictionary.

15. We have following information on countries and their population (population is in crores),

Country Population

China 143

India 136

USA 32

UK 21

Using above create a dictionary of countries and its population

Write a program that asks user for 4 type of inputs,

option 1 print: if user enter print then it should print all countries with their population in this format,

china==>143

india==>136

usa==>32

uk==>21

option 2 add: if user input add then it should further ask for a country name to add.

If country already exist in our dataset then it should print that it exist and do nothing. If it doesn't then it asks for population and add that new country/population in our dictionary and print it

option 3 remove: when user inputs remove it should ask for a country to remove. If country exist in our dictionary then remove it and print new dictionary using format shown above in (a). Else print that country doesn't exist!

option 4 query: on this again ask user for which country he or she wants to query. When user inputs that country it will print population of that country.

16. You are given following list of stocks and their prices in last 3 days,

Stock Prices

info [600,630,620]

ril [1430,1490,1567]

mtl [234,180,160]

Write a program that asks user for operation. Value of operations could be,

option 1, print: When user enters print it should print following,

info ==> [600, 630, 620] ==> avg: 616.67

ril ==> [1430, 1490, 1567] ==> avg: 1495.67

mtl ==> [234, 180, 160] ==> avg: 191.33

option 2, add: When user enters 'add', it asks for stock ticker and price. If stock already exist in your list (like info, ril etc) then it will append the price to the list. Otherwise it will create new entry in your dictionary. For example entering 'tata' and 560 will add tata ==> [560] to the dictionary of stocks.

17. Let's say your expenses for every month are listed below:

January - 2200

February - 2350

March - 2600

April - 2130

May - 2190

June - 1980

July - 2400

August - 2250

September - 2100

October - 2400

November - 2150

December - 2500

Create a Dictionary to store these monthly expenses and using that find out:

1. In February, how many dollars did you spend extra compared to January?
2. Calculate your total expenses for the first quarter (January to March) of the year.
3. Check if you spent exactly 2400 dollars in any month.
4. Modify the expense for June (2080 dollars) to your monthly expenses.
5. You returned an item that you bought in April and received a refund of 200 dollars.
6. Determine which month had the highest expense and print the month and the amount.
7. Calculate the average monthly expense for the first half of the year (January to June).
8. Find the month with the lowest expense and print the month and the amount.

18. You and your wife argued about expenses last night. You both want to know who is spending more in a month. Now you both go to the Little Yoda he is a good python programmer. He suggested that both of you add an entry in a dictionary next time you spend money. So that you can have a clear picture of your expenses and plan to reduce them. Both dictionaries are as below-

Your expenses -

Clothes - 1100

Shoes - 1000

Watch - 900

Mobile Recharge - 699

Petrol - 1980

Your Wife’s expenses -

Mobile Recharge - 799

DTH recharge - 999

Clothes - 2310

Makeup - 3670

Shoes - 999

Find out the total expenses for each of you.

Find out who spending more

Find out which thing you and your wife spending more