Exp. 2: Write a program to demonstrate use of different datatypes and statements in python.

Problem Statements

1. Write a Python program to find those numbers which are divisible by 7 and multiple of 5, between 1500 and 2700 (both included).

2. Write a Python program to convert temperatures to and from celsius, fahrenheit.

[Formula : c/5 = f-32/9 [where c = temperature in celsius and f = temperature in fahrenheit]
Expected Output :
60°C is 140 in Fahrenheit
45°F is 7 in Celsius

3. Write a Python program to construct the following pattern, using a nested for loop.

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

4. Write a Python program that accepts a word from the user and reverse it

5. Write a Python program to count the number of even and odd numbers from a series of numbers. *Sample numbers* : numbers = (1, 2, 3, 4, 5, 6, 7, 8, 9)
Expected Output :
Number of even numbers : 5
Number of odd numbers : 4

6. Write a Python program that prints all the numbers from 0 to 7 except 3 and 6.
Note : Use 'continue' statement.
Expected Output : 0 1 2 4 5 7

7. Write a Python program which iterates the integers from 1 to 50. For multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print "FizzBuzz".

Sample Output :
fizzbuzz
1
2
fizz
4
buzz

8. Write a Python program which takes two digits m (row) and n (column) as input and generates a two-dimensional array. The element value in the i-th row and j-th column of the array should be i*j.
Note :
i = 0,1.., m-1
j = 0,1, n-1.

Test Data : Rows = 3, Columns = 4
Expected Result : [[0, 0, 0, 0], [0, 1, 2, 3], [0, 2, 4, 6]]

9. Write a Python program that accepts a sequence of lines (blank line to terminate) as input and prints the lines as output (all characters in lower case).

10. Write a Python program which accepts a sequence of comma separated 4 digit binary numbers as its input and print the numbers that are divisible by 5 in a comma separated sequence.
Sample Data : 0100,0011,1010,1001,1100,1001
Expected Output : 1010

11. Write a Python program that accepts a string and calculate the number of digits and letters.

Sample Data : Python 3.2
Expected Output :
Letters 6
Digits 2

12. Write a Python program to check the validity of password input by users
Validation :

- At least 1 letter between [a-z] and 1 letter between [A-Z].
- At least 1 number between [0-9].
- At least 1 character from [$#@].
- Minimum length 6 characters.
- Maximum length 16 characters.

13. Write a Python program to find numbers between 100 and 400 (both included) where each digit of a number is an even number. The numbers obtained should be printed in a comma-separated sequence.

14. Write a Python program to print alphabet pattern 'L'.

Expected Output:

```
*
*
*
*
*
*
*****
```

15. Write a Python program to print alphabet pattern 'T'.
Expected Output:

```
*****
  *
  *
  *
  *
  *
  *
```

16. Write a Python program to print alphabet pattern 'U'.

Expected Output:

```
 *   *
 *   *
 *   *
 *   *
 *   *
 *   *
  ***
```

17. Write a Python program to print alphabet pattern 'Z'.

Expected Output:

```
*******
     *
    *
   *
  *
 *
*******
```

18. Write a Python program to calculate a dog's age in dog's years. Note: For the first two years, a dog year is equal to 10.5 human years. After that, each dog year equals 4 human years.

Expected Output:

```
Input a dog's age in human years: 15
The dog's age in dog's years is 73
```

19. Write a Python program to check whether an alphabet is a vowel or consonant.

Expected Output:

```
Input a letter of the alphabet: k
k is a consonant.
```

20. Write a Python program to convert month name to a number of days.

Expected Output:

```
List of months: January, February, March, April, May, June, July, August
, September, October, November, December
Input the name of Month: February
No. of days: 28/29 days
```

21. Write a Python program to sum of two given integers. However, if the sum is between 15 to 20 it will return 20.

22. Write a Python program to check a string represent an integer or not.

Expected Output:

```
Input a string: Python
The string is not an integer.
```

23. Write a Python program to check a triangle is equilateral, isosceles or scalene.

Note :
An equilateral triangle is a triangle in which all three sides are equal.
A scalene triangle is a triangle that has three unequal sides.
An isosceles triangle is a triangle with (at least) two equal sides.
Expected Output:

```
Input lengths of the triangle sides:
x: 6
y: 8
z: 12
Scalene triangle
```

24. Write a Python program that reads two integers representing a month and day and prints the season for that month and day.

Expected Output:

```
Input the month (e.g. January, February etc.): july
Input the day: 31
Season is autumn
```

25. Write a Python program to display astrological sign for given date of birth.

Expected Output:

```
Input birthday: 15
```

```
Input month of birth (e.g. march, julyetc): may
Your Astrological sign is : Taurus
```

26. Write a Python program to display the sign of the Chinese Zodiac for given year in which you were born.

Expected Output:

```
Input your birth year: 1973
Your Zodiac sign : Ox
```

27. Write a Python program to find the median of three values.

Expected Output:

```
Input first number: 15
Input second number: 26
Input third number: 29
The median is 26.0
```

28. Write a Python program to get next day of a given date.
Expected Output:

```
Input a year: 2016
Input a month [1-12]: 08
Input a day [1-31]: 23
The next date is [yyyy-mm-dd] 2016-8-24
```

29. Write a Python program to calculate the sum and average of n integer numbers (input from the user). Input 0 to finish.

30. Write a Python program to create the multiplication table (from 1 to 10) of a number.

Expected Output:

```
Input a number: 6
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
```

```
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60
```

31. Write a Python program to construct the following pattern, using a nested loop number.

Expected Output:

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```

32. Write a program to display following pattern

```
*
*   *
*   *   *
*   *   *   *
```

33. Write a program to display following pattern

```
0
0   1
0   1   2
0   1   2   3
```

34. Write a program to display following pattern

```
*   *   *   *   *   *
*   *   *   *   *
*   *   *   *
*   *   *
```

* *

*

35. Write a program to display following pattern

0 1 2 3 4 5

0 1 2 3 4

0 1 2 3

0 1 2

0 1

0

Note:

[**Every student of each Batch should solve minimum 7 problem statements]**

Problem Statements (SET,TUPLE)

1. Write a Python program to create and iterate over sets.

2. Write a program to perform following operations on sets.

1. Union 2.intersection 3. Set difference. 4. Symmetric difference. 5. Disjoint 6.issubset 7.propersubset 8. Superset 9.propersuperset

3. Write a program to implement various methods for modifying set.

1. add () 2. remove() 3.discard() 4.pop() 5.clear()

4. Write a program to demonstrate use of nested sets.

5. Write a program to access tuple using positive and negative indices.

6. Write a program to implement functions on tuple.

Problem Statements (Dictionary)

1. Write a Python script to sort (ascending and descending) a dictionary by value.

2. Write a Python script to add a key to a dictionary

Sample Dictionary: {0: 10, 1: 20}
Expected Result : {0: 10, 1: 20, 2: 30}

3. Write a Python script to concatenate following dictionaries to create a new one.

Sample Dictionary :
dic1={1:10, 2:20}
dic2={3:30, 4:40}
dic3={5:50,6:60}
Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}

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

5. Write a Python program to iterate over dictionaries using for loops.

6. Write a Python script to generate and print a dictionary that contains a number (between 1 and n) in the form (x, x*x).

Sample Dictionary (n = 5) :
Expected Output : {1: 1, 2: 4, 3: 9, 4: 16, 5: 25}

7. Write a Python script to print a dictionary where the keys are numbers between 1 and 15 (both included) and the values are square of keys.
Sample Dictionary
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100, 11: 121, 12: 144, 13: 169, 14: 196, 15: 225}

8. Write a Python script to merge two Python dictionaries.

9. Write a Python program to iterate over dictionaries using for loops.

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

11. Write a Python program to multiply all the items in a dictionary.

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

13. Write a Python program to map two lists into a dictionary.

14. Write a Python program to sort a given dictionary by key.

15. Write a Python program to get the maximum and minimum value in a dictionary.

16. Write a Python program to get a dictionary from an object's fields.

17. Write a Python program to remove duplicates from Dictionary.

18. Write a Python program to check a dictionary is empty or not.

19. Write a Python program to combine two dictionary adding values for common keys.

d1 = {'a': 100, 'b': 200, 'c':300}
d2 = {'a': 300, 'b': 200, 'd':400}
Sample output: Counter({'a': 400, 'b': 400, 'd': 400, 'c': 300})

20. Write a Python program to print all unique values in a dictionary.

Sample Data : [{"V":"S001"}, {"V": "S002"}, {"VI": "S001"}, {"VI": "S005"}, {"VII":"S005"}, {"V":"S009"},{"VIII":"S007"}]
Expected Output : Unique Values: {'S005', 'S002', 'S007', 'S001', 'S009'}

21. Write a Python program to create and display all combinations of letters, selecting each letter from a different key in a dictionary.
Sample data : {'1':['a','b'], '2':['c','d']}
Expected Output:
ac
ad
bc
bd

22. Write a Python program to find the highest 3 values of corresponding keys in a dictionary.
23. Write a Python program to combine values in python list of dictionaries.

Sample data: [{'item': 'item1', 'amount': 400}, {'item': 'item2', 'amount': 300}, {'item': 'item1', 'amount': 750}]
Expected Output: Counter({'item1': 1150, 'item2': 300})

24. Write a Python program to create a dictionary from a string.

Note: Track the count of the letters from the string.
Sample string : 'w3resource'
Expected output: {'w': 1, '3': 1, 'r': 2, 'e': 2, 's': 1, 'o': 1, 'u': 1, 'c': 1}

25. Write a Python program to print a dictionary in table format.

26. Write a Python program to count the values associated with key in a dictionary.
Expected Output:
6
2

27. Write a Python program to convert a list into a nested dictionary of keys.

28. Write a Python program to sort a list alphabetically in a dictionary.

29. Write a Python program to remove spaces from dictionary keys.

30. Write a Python program to get the top three items in a shop.
Sample data: {'item1': 45.50, 'item2':35, 'item3': 41.30, 'item4':55, 'item5': 24}
Expected Output:
item4 55
item1 45.5
item3 41.3

31. Write a Python program to get the key, value and item in a dictionary.

32. Write a Python program to print a dictionary line by line.

33. Write a Python program to check multiple keys exists in a dictionary.

34. Write a Python program to count number of items in a dictionary value that is a list.

35. Write a Python program to sort Counter by value.

Sample data : {'Math':81, 'Physics':83, 'Chemistry':87}
Expected data: [('Chemistry', 87), ('Physics', 83), ('Math', 81)]

36. Write a Python program to create a dictionary from two lists without losing duplicate values.
Sample lists: ['Class-V', 'Class-VI', 'Class-VII', 'Class-VIII'], [1, 2, 2, 3]
Expected Output: defaultdict(<class 'set'>, {'Class-V': {1}, 'Class-VI': {2}, 'Class-VII': {2}, 'Class-VIII': {3}})

37. Write a Python program to replace dictionary values with their average.

38. Write a Python program to match key values in two dictionaries.

Sample dictionary: {'key1': 1, 'key2': 3, 'key3': 2}, {'key1': 1, 'key2': 2}
Expected output: key1: 1 is present in both x and y

39. Write a Python program to find the key of the maximum value in a dictionary.

Sample Output:
Original dictionary elements:
{'Theodore': 19, 'Roxanne': 22, 'Mathew': 21, 'Betty': 20}
Finds the key of the maximum and minimum value of the said dictionary:
('Roxanne', 'Theodore')

40. Write a Python program to create a dictionary of keys x, y, and z where each key has as value a list from 11-20, 21-30, and 31-40 respectively. Access the fifth value of each key from the dictionary.

```
 {'x': [11, 12, 13, 14, 15, 16, 17, 18, 19],
'y': [21, 22, 23, 24, 25, 26, 27, 28, 29],
'z': [31, 32, 33, 34, 35, 36, 37, 38, 39]}
15
25
35
x has value [11, 12, 13, 14, 15, 16, 17, 18, 19]
y has value [21, 22, 23, 24, 25, 26, 27, 28, 29]
z has value [31, 32, 33, 34, 35, 36, 37, 38, 39]
```

41. Write a Python program to drop empty Items from a given Dictionary.

```
 Original Dictionary:
{'c1': 'Red', 'c2': 'Green', 'c3': None}
New Dictionary after dropping empty items:
{'c1': 'Red', 'c2': 'Green'}
```

42. Write a Python program to filter a dictionary based on values.

```
 Original Dictionary:
{'Cierra Vega': 175, 'Alden Cantrell': 180, 'Kierra Gentry': 165, 'Pierre Cox': 190}
Marks greater than 170:
{'Cierra Vega': 175, 'Alden Cantrell': 180, 'Pierre Cox': 190}
```

43. Write a Python program to convert more than one list to nested dictionary.

```
 Original strings:
['S001', 'S002', 'S003', 'S004']
['Adina Park', 'Leyton Marsh', 'Duncan Boyle', 'Saim Richards']
[85, 98, 89, 92]
Nested dictionary:
[{'S001': {'Adina Park': 85}}, {'S002': {'Leyton Marsh': 98}}, {'S003': {'Duncan Boyle': 89}}, {'S004':
{'Saim Richards': 92}}]
```

44. Write a Python program to filter the height and width of students, which are stored in a dictionary.

```
 Original Dictionary:
{'Cierra Vega': (6.2, 70), 'Alden Cantrell': (5.9, 65), 'Kierra Gentry': (6.0, 68), 'Pierre Cox': (5.8, 66)}
Height > 6ft and Weight> 70kg:
```

{'Cierra Vega': (6.2, 70)}

45. Write a Python program to check all values are same in a dictionary.

Original Dictionary:
{'Cierra Vega': 12, 'Alden Cantrell': 12, 'Kierra Gentry': 12, 'Pierre Cox': 12}
Check all are 12 in the dictionary.
True
Check all are 10 in the dictionary.
False

46. Write a Python program to create a dictionary grouping a sequence of key-value pairs into a dictionary of lists.

Original list:
[('yellow', 1), ('blue', 2), ('yellow', 3), ('blue', 4), ('red', 1)]
Grouping a sequence of key-value pairs into a dictionary of lists:
{'yellow': [1, 3], 'blue': [2, 4], 'red': [1]}

47. Write a Python program to split a given dictionary of lists into list of dictionaries.

Original dictionary of lists:
{'Science': [88, 89, 62, 95], 'Language': [77, 78, 84, 80]}
Split said dictionary of lists into list of dictionaries:
[{'Science': 88, 'Language': 77}, {'Science': 89, 'Language': 78}, {'Science': 62, 'Language': 84}, {'Science': 95, 'Language': 80}]

48. Write a Python program to remove a specified dictionary from a given list.

Original list of dictionary:
[{'id': '#FF0000', 'color': 'Red'}, {'id': '#800000', 'color': 'Maroon'}, {'id': '#FFFF00', 'color': 'Yellow'}, {'id': '#808000', 'color': 'Olive'}]
Remove id #FF0000 from the said list of dictionary:
[{'id': '#800000', 'color': 'Maroon'}, {'id': '#FFFF00', 'color': 'Yellow'}, {'id': '#808000', 'color': 'Olive'}]

49. Write a Python program to convert string values of a given dictionary, into integer/float datatypes.

Original list:
[{'x': '10', 'y': '20', 'z': '30'}, {'p': '40', 'q': '50', 'r': '60'}]
String values of a given dictionary, into integer types:

[{'x': 10, 'y': 20, 'z': 30}, {'p': 40, 'q': 50, 'r': 60}]
Original list:
[{'x': '10.12', 'y': '20.23', 'z': '30'}, {'p': '40.00', 'q': '50.19', 'r': '60.99'}]
String values of a given dictionary, into float types:
[{'x': 10.12, 'y': 20.23, 'z': 30.0}, {'p': 40.0, 'q': 50.19, 'r': 60.99}]

50. A Python Dictionary contains List as value. Write a Python program to clear the list values in the said dictionary.

Original Dictionary:
{'C1': [10, 20, 30], 'C2': [20, 30, 40], 'C3': [12, 34]}
Clear the list values in the said dictionary:
{'C1': [], 'C2': [], 'C3': []}

51. A Python Dictionary contains List as value. Write a Python program to update the list values in the said dictionary.
Original Dictionary:
{'Math': [88, 89, 90], 'Physics': [92, 94, 89], 'Chemistry': [90, 87, 93]}
Update the list values of the said dictionary:
{'Math': [89, 90, 91], 'Physics': [90, 92, 87], 'Chemistry': [90, 87, 93]}

52. Write a Python program to extract a list of values from a given list of dictionaries.
Original Dictionary:
[{'Math': 90, 'Science': 92}, {'Math': 89, 'Science': 94}, {'Math': 92, 'Science': 88}]
Extract a list of values from said list of dictionaries where subject = Science
[92, 94, 88]
Original Dictionary:
[{'Math': 90, 'Science': 92}, {'Math': 89, 'Science': 94}, {'Math': 92, 'Science': 88}]
Extract a list of values from said list of dictionaries where subject = Math
[90, 89, 92]

53. Write a Python program to find the length of a given dictionary values.

Original Dictionary:
{1: 'red', 2: 'green', 3: 'black', 4: 'white', 5: 'black'}
Length of dictionary values:
{'red': 3, 'green': 5, 'black': 5, 'white': 5}
Original Dictionary:
{'1': 'Austin Little', '2': 'Natasha Howard', '3': 'Alfred Mullins', '4': 'Jamie Rowe'}
Length of dictionary values:

{'Austin Little': 13, 'Natasha Howard': 14, 'Alfred Mullins': 14, 'Jamie Rowe': 10}

54. Write a Python program to get the depth of a dictionary.

Expected Output:
4

55. Write a Python program to access dictionary key's element by index.

Expected Output:
physics
math
chemistry

56. Write a Python program to convert a given dictionary into a list of lists.

Original Dictionary:
{1: 'red', 2: 'green', 3: 'black', 4: 'white', 5: 'black'}
Convert the said dictionary into a list of lists:
[[1, 'red'], [2, 'green'], [3, 'black'], [4, 'white'], [5, 'black']]
Original Dictionary:
{'1': 'Austin Little', '2': 'Natasha Howard', '3': 'Alfred Mullins', '4': 'Jamie Rowe'}
Convert the said dictionary into a list of lists:
[['1', 'Austin Little'], ['2', 'Natasha Howard'], ['3', 'Alfred Mullins'], ['4', 'Jamie Rowe']]

57. Write a Python program to filter even numbers from a given dictionary values.
Original Dictionary:
{'V': [1, 4, 6, 10], 'VI': [1, 4, 12], 'VII': [1, 3, 8]}
Filter even numbers from said dictionary values:
{'V': [4, 6, 10], 'VI': [4, 12], 'VII': [8]}
Original Dictionary:
{'V': [1, 3, 5], 'VI': [1, 5], 'VII': [2, 7, 9]}
Filter even numbers from said dictionary values:
{'V': [], 'VI': [], 'VII': [2]}

58. Write a Python program to get all combinations of key-value pairs in a given dictionary.

Original Dictionary:
{'V': [1, 4, 6, 10], 'VI': [1, 4, 12], 'VII': [1, 3, 8]}
Combinations of key-value pairs of the said dictionary:
[{'V': [1, 4, 6, 10], 'VI': [1, 4, 12]}, {'V': [1, 4, 6, 10], 'VII': [1, 3, 8]}, {'VI': [1, 4, 12], 'VII': [1, 3, 8]}]

Original Dictionary:
{'V': [1, 3, 5], 'VI': [1, 5]}
Combinations of key-value pairs of the said dictionary:
[{'V': [1, 3, 5], 'VI': [1, 5]}]

59. Write a Python program to find the specified number of maximum values in a given dictionary.
Original Dictionary:
{'a': 5, 'b': 14, 'c': 32, 'd': 35, 'e': 24, 'f': 100, 'g': 57, 'h': 8, 'i': 100}
1 maximum value(s) in the said dictionary:
['f']
2 maximum value(s) in the said dictionary:
['f', 'i']
5 maximum value(s) in the said dictionary:
['f', 'i', 'g', 'd', 'c']

60. Write a Python program to find shortest list of values with the keys in a given dictionary.

Original Dictionary: {'V': [10, 12], 'VI': [10], 'VII': [10, 20, 30, 40], 'VIII': [20], 'IX': [10, 30, 50, 70], 'X': [80]}

Shortest list of values with the keys of the said dictionary: ['VI', 'VIII', 'X']