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

Sample Output

Dict = { 'M1' : [67,79,90,73,36], 'M2' : [89,67,84], 'M3' : [82,57] }

Number of Items in a Dictionary : 10

2-**Write a Python program by function to print a dictionary line by line.**

Sample Output

```
Dict = {  "Sam" : {"M1" : 89, "M2" : 56, "M3" : 89},
           "Suresh" : {"M1" : 49, "M2" : 96, "M3" : 89} }

Sam
M1 : 89
M2 : 56
M3 : 89
Suresh
M1 : 49
M2 : 96
M3 : 89
```

3-Let's use functions to calculate your trip's costs:

- Define a function called `hotel_cost` with one argument `nights` as input. The hotel costs $140 per night. So, the function `hotel_cost` should return 140 * nights.
- Define a function called `plane_ride_cost` that takes a string, `city`, as input. The function should return a different price depending on the location, similar to the code example above. Below are the valid destinations and their corresponding round-trip prices.
- `"Charlotte": 183`
- `"Tampa": 220`
- `"Pittsburgh": 222`
- `"Los Angeles": 475`

-Below your existing code, define a function called `rental_car_cost` with an argument called days. Calculate the cost of renting the car: Every day you rent the car costs $40.(cost=40*days) if you rent the car for 7 or more days, you get $50 off your total(cost-=50). Alternatively (elif), if you rent the car for 3 or more days, you get $20 off your total. You cannot get both of the above discounts. Return that cost. -Then, define a function called `trip_cost` that takes two arguments, `city` and `days`. Like the example above, have your function return the sum of calling the `rental_car_cost(days)`, `hotel_cost(days)`, and `plane_ride_cost(city)` functions.

- Modify your `trip_cost` function definion. Add a third argument, `spending_money`. Modify what the `trip_cost` function does. Add the variable `spending_money to the sum that it returns.

4-Follow the stpes:

- First, def a function called `cube` that takes an argument called `number`.
- Make that function return the cube of that number (i.e. that number multiplied by itself and multiplied by itself once again).
- Define a second function called `by_three` that takes an argument called number. if that number is divisible by 3,`by_three`should call `cube(number)` and return its result. Otherwise, `by_three` should return False. -Check if it works

5-**Write a function to sort dictionary by values (Ascending/ Descending).**

Sample Output

Dictionary = { "m1" : 78 , "m2" : 89 , "m3" : 64 , "m4" : 35 , "m5" : 71 }

Ascending Order = [('m4', 35), ('m3', 64), ('m5', 71), ('m1', 78), ('m2', 89)]

Descending Order = [('m2', 89), ('m1', 78), ('m5', 71), ('m3', 64), ('m4', 35)]