1. **Python | Ways to remove a key from dictionary**

# declare a dictionary

dict1 = {1: "one", 2: "two", 3: "three", 4: "four"}

print("Original Dictionary: ", dict1)

# using del and giving the key as index

del dict1[1]

print("Dictionary after del: ", dict1)

# using del and giving the key as index

del dict1[3]

print("Dictionary after del: ", dict1)

1. **Ways to sort list of dictionaries by values in Python – Using itemgetter**

from operator import itemgetter

lis = [{ "name" : "Nandini", "age" : 20},

{ "name" : "Manjeet", "age" : 20 },

{ "name" : "Nikhil" , "age" : 19 }]

print "The list printed sorting by age: "

print sorted(lis, key=itemgetter('age'))

print ("\r")

print "The list printed sorting by age and name: "

print sorted(lis, key=itemgetter('age', 'name'))

print ("\r")

# using sorted and itemgetter to print list sorted by age in descending order

print "The list printed sorting by age in descending order: "

print sorted(lis, key=itemgetter('age'),reverse = True)

1. **Ways to sort list of dictionaries by values in Python – Using lambda function**

employees = [{'id':'101', 'name':'Adam', 'department':'IT'}, {'id':'102', 'name':'Ricky', 'department':'Electronics'}, {'id':'103', 'name': 'Shane', 'department':'Auto'}]

print("List of employees :")

print(employees)

sorted\_employees = sorted(employees, key = lambda x: x['department'])

print("\nSorted List of employees :")

print(sorted\_employees)

1. **Python | Merging two Dictionaries**

# Python code to merge dict using update() method

def Merge(dict1, dict2):

return(dict2.update(dict1))

# Driver code

dict1 = {'a': 10, 'b': 8}

dict2 = {'d': 6, 'c': 4}

# This return None

print(Merge(dict1, dict2))

# changes made in dict2

print(dict2)

1. **Python – Convert key-values list to flat dictionary**

from itertools import product

my\_dict = {'month\_num' : [1, 2, 3, 4, 5, 6], 'name\_of\_month' : ['Jan', 'Feb', 'March', 'Apr', 'May', 'June']}

print("The dictionary is : ")

print(my\_dict)

my\_result = dict(zip(my\_dict['month\_num'], my\_dict['name\_of\_month']))

print("The flattened dictionary is: ")

print(my\_result)

1. **Python – Insertion at the beginning in OrderedDict**

from collections import OrderedDict

dic1 = OrderedDict([('A', '100'), ('B', '200'), ('C', '300')])

insrt = OrderedDict([("D", '400')])

final = OrderedDict(list(insrt.items()) + list(dic1.items()))

# print result

print ("Resultant Dictionary :")

print(final)

1. **Python | Check order of character in string using OrderedDict( )**

from collections import OrderedDict

def checkOrder(input, pattern):

# create empty OrderedDict

# output will be like {'a': None,'b': None, 'c': None}

dict = OrderedDict.fromkeys(input)

# traverse generated OrderedDict parallel with

# pattern string to check if order of characters

# are same or not

ptrlen = 0

for key,value in dict.items():

if (key == pattern[ptrlen]):

ptrlen = ptrlen + 1

# check if we have traverse complete

# pattern string

if (ptrlen == (len(pattern))):

return 'true'

# if we come out from for loop that means

# order was mismatched

return 'false'

# Driver program

if \_\_name\_\_ == "\_\_main\_\_":

input = 'engineers rock'

pattern = 'egr'

print (checkOrder(input,pattern))

1. **Dictionary and counter in Python to find winner of election**

from collections import Counter

def winner(input):

votes = Counter(input)

dict = {}

for value in votes.values():

# initialize empty list to each key to

# insert candidate names having same

# number of votes

dict[value] = []

for (key,value) in votes.items():

dict[value].append(key)

# sort keys in descending order to get maximum

# value of votes

maxVote = sorted(dict.keys(),reverse=True)[0]

# check if more than 1 candidates have same

# number of votes. If yes, then sort the list

# first and print first element

if len(dict[maxVote])>1:

print (sorted(dict[maxVote])[0])

else:

print (dict[maxVote][0])

# Driver program

if \_\_name\_\_ == "\_\_main\_\_":

input =['john','johnny','jackie','johnny',

'john','jackie','jamie','jamie',

'john','johnny','jamie','johnny',

'john']

winner(input)

1. **Python – Append Dictionary Keys and Values ( In order ) in dictionary**

test\_dict = {"Gfg" : 1, "is" : 3, "Best" : 2}

# printing original dictionary

print("The original dictionary is : " + str(test\_dict))

# + operator is used to perform adding keys and values

res = list(test\_dict.keys()) + list(test\_dict.values())

# printing result

print("The ordered keys and values : " + str(res))

1. **Python | Sort Python Dictionaries by Key or Value**

# Function calling

def dictionairy():

# Declare hash function

key\_value ={}

# Initializing value

key\_value[2] = 56

key\_value[1] = 2

key\_value[5] = 12

key\_value[4] = 24

key\_value[6] = 18

key\_value[3] = 323

print ("Task 1:-\n")

print ("Keys are")

# iterkeys() returns an iterator over the

# dictionary’s keys.

for i in sorted (key\_value.keys()) :

print(i, end = " ")

def main():

# function calling

dictionairy()

# Main function calling

if \_\_name\_\_=="\_\_main\_\_":

main()