**Week 6- Programs on dictionary and Files**

**Name: Naren Chandrashekhar**

**SRN: PES2UG20CS216**

**Section: G**

|  |  |
| --- | --- |
| Program 1 | Write a program to print a dictionary where the keys are numbers between 1 and 15 and the values are cube of keys. |
|  | **Input:**  **Enter range till you need to generate cubes: 10**  **output:**  **The resultant dictionary with cube as value numbers between 1 and 10 is**  **{1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6: 216, 7: 343, 8: 512, 9: 729, 10: 1000}** |
|  | Algorithm:  Step1: Start  Step2: Read value from user to find cubes till the number, x  Step3: Create a new dictionary  Step4: for I in range from1 to x+1  Step5: d[i]=(i,i^3)  Step6: repeat step 5 till all elements have been added to d  Step7: print the dictionary  Step8: End |
|  | Program:  x = int(input("Enter a number till when you want to find its cube to "))  d = dict() #creating anew dictionary  for i in range(1,x+1):  d[i] = pow(i,3) #using pow function to find cube of the number  print(d) |
|  | Output: |
| Program 2 | Construct dictionary phone\_book with :  Key:number of entries, Values: (name,phone number,email,address) and perform the following operations:  i) Add a new number to phone\_book  ii) delete entry from phone book. |
|  | Algorithm:  Step1: Start  Step2: set value in values  Step3: Enumerate through values and add them into a dictionary, d  Step4: Display the dictionary  Step5: Change the length of the dictionary by adding anew element and reading its value into the newly added key  Step6: Print the new dictionary  Step7: Enter the user input to delete the specified key from the dictionary  Step8: Check is entered key is present in the dictionary  Step9: if present, delete the key. If not print “key not found”  Step10: Print the dictionary  Step11: End |
|  | Program:  values = [("Rashma",8105731555,"rashma@gmail.com","bangalore"),  ("Saritha",9582161900,"saritha@gmail.com","Mangalore"),  ("Bharathi",9276895311,"bharathi@yahoo.com","Coimbatore"),  ("deepthi",8976885553,"deepthi@gmail.com","bangalore"),  ("kakoli",8816121598,"kakili@gmail.com","dispur")]  d = dict(enumerate(values,1)) #raed values to dictionary one by one  print("The phone book is \n",d,"\n")  # i) Add new number to phone book  #adding the new value at the end of the dictionary  d[len(d)+1] = [('sreenath', 9872345670, 'sreenath@pes.edu', 'kolar')]  print("The phone book after adding a new number to the phone book is \n",d,"\n")  # ii) Delete an entry based on the key entered by user  key = int(input("Enter the phone book entry to be deleted "))  if key in d: #condition to check if input key is present in the dictionary  del d[key] #deleting key from dictionary  else:  print("Key not found")    print("After deleting key from the phone book, the phone book is \n",d) |
|  | Output: |
| Program 3 | Given list of students, marks for phy,chem,maths and biology form a dictionary where key is SRN and values is dictionary containing PCMB marks of respective student. |
|  | Algorithm:  Step1: Start  Step2: Set srns, p\_marks, c\_marks, m\_marks, b\_marks  Step3: Create two new dictionaries, student\_marks and marks\_det  Step4: iterate through srns  Step5: assign subject to marks scored in the particular subject  Step6: repeat step5 for all srns and keep adding values to student\_marks  Step7: Print student\_marks  Step8: End |
|  | Program:  srns = ["PECS001","PECS015","PECS065","PECS035","PECS038"] #set srns  #set marks scored by each student in each subject, physics, chemistry, maths and biology  p\_marks = [98,99,85,92,79]  c\_marks = [91,90,84,98,99]  m\_marks = [78,39,60,50,84]  b\_marks = [95,59,78,80,89]  #create a new dictionary for student marks and marks  student\_marks={}  marks\_det={}  #for loop used to add marks into marks\_det for each student for each subject  for i in range(0,len(srns)):  marks\_det['Physics'] = p\_marks[i]  marks\_det['Chemistry'] = c\_marks[i]  marks\_det['Maths'] = m\_marks[i]  marks\_det['Biology'] = b\_marks[i]  student\_marks[srns[i]] = marks\_det  print(student\_marks) |
|  | Output: |
| Program 4 | a)Read movie data from mov1.csv file. CSV file mov1.csv has three columns c1 has year,c2 has rating,c3 has movie name.  b) write the year of release and movie name from mov1.csv to a text file |
|  | Algorithm:  Step1: Start  Step2: open file ‘mov1.csv’ in read mode  Step3: read contents of the file and store in x  Step4: print x  Step5: close the file  Step6: open two files, one in read mode, ‘mov1.csv’ and one in write mode, ‘newmov1.txt’  Step7: read each line of file into variable line  Step8: extract year of release and movie name from line  Step9: write year of release and movie name to newmov1.txt  Step10: repeat step7,step8 and step9 till all lines are read from mov1.csv  Step11: close all files  Step12: End |
|  | Program:  """  Read movie data from mov1.csv file. CSV file mov1.csv has three columns c1 has  year,c2 has rating,c3 has movie name.  """  file1 = open('mov1.csv','r') #opening file mov1.csv in read mode  x = file1.read() #reading the entire contents of the file  print(x)  file1.close()  """  write the year of release and movie name from mov1.csv to a text file  """  file2 = open('mov1.csv','r')  file3 = open('newmov1.txt','w') #creating a new file to store the values year of release and movie name  #read one line at a time, extract the year of release and movie name and  #write these values to the file newmov1.txt  line = file2.readline()  while line:  line = line.strip() #remove extra characters and \n  y = line.split(',')  print(y[0],y[-1], file = file3) #write year of release and movie name to file newmov1.txt  line = file2.readline()  # close the files  file2.close()  file3.close() |
|  | Output:    Output of newmov1.txt: |
| Program 5 | From file mov1.csv make a dictionary with Key as year and values as name of movies released in that year. |
|  | Algorithm:  Step1: Start  Step2: open file, ‘mov1.csv’ in read mode  Step3: create a new dictionary  Step4: read each line of file into line  Step5: condition to check if movies are repeated  Step6: add year of release and movie name into the dictionary  Step7: read all lines of the file  Step8: Repeat step5 Step6, step7 till all lines are read  Step9: Print d  Step10: close all files |
|  | Program:  file1 = open('mov1.csv','r') #open file in read mode  d = dict()  #read one line at a time, extract the year of release and movie name and  #write these values to the dictionary  line = file1.readline()  while line:  line = line.strip()  list1 = line.split(',')  if list1[0] not in d: #condition to check repitition  d[list1[0]]=[]  d[list1[0]].insert(0,list1[-1]) #adding values to the dictionary  line = file1.readline()  print(d)  file1.close() |
|  | Output: |
| Program 6 | In the given dictionary find total marks and percentage  **{'PECS001': {'phy': 79, 'chem': 99, 'mat': 84, 'Bio': 84}, 'PECS015': {'phy': 79, 'chem': 99, 'mat': 84, 'Bio': 84}, 'PECS065': {'phy': 79, 'chem': 99, 'mat': 84, 'Bio': 84}, 'PECS035': {'phy': 79, 'chem': 99, 'mat': 84, 'Bio': 84}, 'PECS038': {'phy': 79, 'chem': 99, 'mat': 84, 'Bio': 84}}** |
|  | Algorithm:  Step1: Start  Step2: Initialize dictionary, Stu\_marks with srn as key and subject:marks as values  Step3: Print Stu\_marks  Step4: Create a new dictionary Score\_card  Step5: for each srn in Stu\_marks, read the marks for all subjects  Step6: compute the total and percentage from marks  Step7: Add the srn as key and percentage, marks as values to Score\_card  Step8: Repeat steps 5 to 7 till all items in Stu\_marks are read  Step9: print Score\_card  Step10: End |
|  | Program:  #set Stu\_marks  Stu\_marks={'PECS001': {'phy': 79, 'chem': 90, 'mat': 84, 'Bio': 87},  'PECS015': {'phy': 59, 'chem': 76, 'mat': 74, 'Bio': 66},  'PECS065': {'phy': 89, 'chem': 58, 'mat': 94, 'Bio': 81},  'PECS035': {'phy': 71, 'chem': 91, 'mat': 81, 'Bio': 86},  'PECS038': {'phy': 75, 'chem': 98, 'mat': 75, 'Bio': 84}}  print("detailed marks \n",Stu\_marks,"\n")  Score\_card = {} #create a dictionary  #compute total and percentage for each student and add to Score\_card  for srn,marks in Stu\_marks.items():  d = dict()  d['total'] = sum(marks.values())  d['percent'] = d['total']/4  Score\_card[srn] = d    print(Score\_card) |
|  | Output: |
|  |  |