# 1. Write a program that repeatedly asks the user to enter product names and prices. Store all of these in a dictionary whose keys are the product names and whose values are the prices. When the user is done entering products and prices, allow them to repeatedly enter a product name and print the corresponding price or a message if the product is not in the dictionary.

Ans:

dict = {} *# this is a empty dictionary*

**while** True:

k = raw\_input("Enter your product Name: ")

value = input("Enter price: ")

dict[k] = value *# storing keys and values in the dictionary*

cha = raw\_input("Want to insert again in the dictionary(y/n): ")

**if** cha == "n": *# if choice is no*

**print**("**\n**All Elements are inserted in the dictionary successfully**\n**")

**break** *# get out of the while loop*

**else**:

cha == "y" *# continue inserting keys and values in the dictionary*

**while** True:

**print** "**\n**Tto terminate the program, type 'end'"

k = raw\_input("Enter product name: ") *# input the product name to print the corresponding price*

**if** k **in** dict:

**print** "The price of "'"',k,'"'" is",dict[k] *# print the corresponding price of the entered product name*

**elif** k == "end":

**print** "**\n**The program ended"

**break**

**else**:

**print** '"',k,'"' " is not in the Dictionary"

Enter your product Name: Pen

Enter price: 20

Want to insert again in the dictionary(y/n): y

Enter your product Name: Book

Enter price: 400

Want to insert again in the dictionary(y/n): y

Enter your product Name: Laptop

Enter price: 40000

Want to insert again in the dictionary(y/n): n

All Elements are inserted in the dictionary successfully

Tto terminate the program, type 'end'

Enter product name: Laptop

The price of " Laptop " is 40000

Tto terminate the program, type 'end'

Enter product name: Book

The price of " Book " is 400

Tto terminate the program, type 'end'

Enter product name: Pen

The price of " Pen " is 20

Tto terminate the program, type 'end'

Enter product name: end

The program ended

2. .Using the dictionary created in the previous problem, allow the user to enter a dollar amount and print out all the products whose price is less than that amount.

Ans:

dict = {'Apple':15000,'Orange':1200,'Banana':500,'Grape':2000}

amount = input("Enter amount: $") # input the dollar amount

print "\nProducts whose price is less than "'"$',amount,'"'":"

for key in dict:

if dict[key] < amount: # if the price is less than the entered amount is

print "Product:",key

output: Enter amount: $15000 Products whose price is less than "$ 15000 ": Product: Orange Product: Grape Product: Banana

3.Write a program that uses a dictionary that contains ten user names and passwords. The program should ask the user to enter their username and password. If the username is not in the dictionary, the program should indicate that the person is not a valid user of the system. If the username is in the dictionary, but the user does not enter the right password, the program should say that the password is invalid. If the password is correct, then the program should tell the user that they are now logged in to the system.

Ans:

data={'name1': '1234', 'name2': '6267', 'name3': '4946'}

name = raw\_input('Enter username: ')

pas = raw\_input('Enter password: ')

if pas in data[name]:

print 'Loged in successful', name

else:

print 'Invalid code'

Output:

Enter username: name1 Enter password: 1234 Loged in successful name1

Or,

# 3. Write a program that asks the user to enter a password. If the user enters the right password, the program should tell them they are logged in to the system. Otherwise, the program should ask them to reenter the password. The user should only get five tries to enter the password, after which point the program should tell them that they are kicked off of the system.

Ans:

i = 0

a = 770000

**while** i < 5:

pa = input("Enter the password: ")

**if** pa == a:

**print**("Login Succesful")

**break**

**else**:

**print**("Wrong Password")

i += 1

**if** i == 5:

**print**("**\n**You are kicked off the system")

Output:

Enter the password: 12342

Wrong Password

Enter the password: 12

Wrong Password

Enter the password: 234

Wrong Password

Enter the password: 770000

Login Succesful

# 6. Dictionaries provide a convenient way to store structured data. Here is an example dictionary:

# d=[{ ' name ' : ' Todd ' , ' phone ' : ' 555-1414 ' , ' email ' : ' todd@mail.net ' }, { ' name ' : ' Helga ' , ' phone ' : ' 555-1618 ' , ' email ' : ' helga@mail.net ' }, { ' name ' : ' Princess ' , ' phone ' : ' 555-3141 ' , ' email ' : '' }, { ' name ' : ' LJ ' , ' phone ' : ' 555-2718 ' , ' email ' : ' lj@mail.net ' }]¶

# Write a program that reads through any dictionary like this and prints the following:

# (a) All the users whose phone number ends in an 8

In [1]:

*# Dictionaries of username, phone number and email*

d = [{'name' : 'Todd' , 'phone' : '555-1414' , 'email' : 'todd@mail.net'},

{'name' : 'Helga' , 'phone' : '555-1618' , 'email' : 'helga@mail.net'},

{'name' : 'Princess' , 'phone' : '555-3141' , 'email' : ''},

{'name' : 'LJ' , 'phone' : '555-2718' , 'email' : 'lj@mail.net'}]

**print** "The Name of the Users whose phone number ends in an 8**\n**"

**for** key **in** d: *# if key is in dictionary*

**for** value **in** key: *# if the value is in key*

str = key.get('phone') *# read and get all of the keys 'phone' from the dictionaries*

**if** str[len(str)-1] == '8': *# if all of the phone number that ends in an 8*

**print** "User:",key.get('name') *# print all of the users whose phone number ends in an 8*

**break** *# break statement*

The Name of the Users whose phone number ends in an 8

User: Helga

User: LJ

# (b) All the users that don’t have an email address listed

In [2]:

*# Dictionaries of username, phone number and email*

d = [{ 'name' : 'Todd' , 'phone' : '555-1414' , 'email' : 'todd@mail.net' },

{ 'name' : 'Helga' , 'phone' : '555-1618' , 'email' : 'helga@mail.net' },

{ 'name' : 'Princess' , 'phone' : '555-3141' , 'email' : ' ' },

{ 'name' : 'LJ' , 'phone' : '555-2718' , 'email' : 'lj@mail.net' }]

**print** "The Name of the Users that don’t have an email address listed**\n**"

**for** key **in** d: *# if key is in dictionary*

**for** value **in** key: *# if the value is in key*

str = key.get('email') *# read and get all of the keys 'email' from the dictionaries*

**if** str[len(str)-1] == ' ': *## if all of the user that don’t have an email address*

**print** "User:",key.get('name') *# print all of the users that don’t have an email address listed*

**break** *# break statement*

The Name of the Users that don’t have an email address listed

User: Princess

In [ ]:

# 7. The keys in a dictionary are guaranteed to be unique, but the values are not. Write a function called count\_values that takes a single dictionary as an argument and returns the number of dis-tinct values it contains. Given the input {’red’:1, ’green’:1, ’blue’:2} , for example, it would return 2.

In [1]:

**def** count\_values(dict): *# Here count\_values is the function name with single argument dict*

values\_count = {} *# empty dictionary*

**for** key **in** dict: *# if key is in the dictionary*

values\_count[dict[key]] = dict[key]

**return** len(values\_count) *# return the length of values\_count*

dict = {'red':1, 'green':1, 'blue':2} *# given input*

**print** "Unique Values in Dictionary:",count\_values(dict)

Unique Values in Dictionary: 2

# lect-5:

# 1. Write a program that uses a while loop (not a for loop) to read through a string and print the characters of the string one-by-one on separate lines.

# Ans:

i = 0

a = 'I am Radia'

**while** i < len(a):

**print** a[i]

i += 1

# Output:

I

a

m

R

a

d

i

a

# 2.Write the same program to print the second characters of the every strings.

In [2]:

i = 0

a = ['Raida', 'Islam','Oyshee',]

**for** i **in** a:

**if** len(i) > 1:

**print** i[1]

a

s

y

# 4. Recall that, given a string s , s.index( ' x ' ) returns the index of the first x in s and an error if there is no x.

# (a) Write a program that asks the user for a string and a letter. Using a while loop, the program should print the index of the first occurrence of that letter and a message if the string does not contain the letter.

In [ ]:

i = 0

**while** i < 5: *# when condition is true*

s = raw\_input("Enter a String: ") *# input a string*

letter = raw\_input("Enter a Letter: ") *# input a letter*

**if** letter **in** s: *# if letter is in the string*

find = s.index(letter) *# find out the first index of the letter*

**print** "**\n**First occurrence of "'"',letter,'"'" is found at index:",find,"**\n**" *# print statement*

**else**: *# if letter is not in the string 's'*

**print** "**\n**The String does not contain the Letter**\n**"

i = i + 1

Enter a String: My name is Radia

Enter a Letter: R

First occurrence of " R " is found at index: 11

Enter a String: I am a student

Enter a Letter: d

First occurrence of " d " is found at index: 10

Enter a String: Hellow

Enter a Letter: r

The String does not contain the Letter

Lect-5:

# 7. The keys in a dictionary are guaranteed to be unique, but the values are not. Write a function called count\_values that takes a single dictionary as an argument and returns the number of dis-tinct values it contains. Given the input {’red’:1, ’green’:1, ’blue’:2} , for example, it would return 2.

In [1]:

**def** count\_values(dict): *# Here count\_values is the function name with single argument dict*

values\_count = {} *# empty dictionary*

**for** key **in** dict: *# if key is in the dictionary*

values\_count[dict[key]] = dict[key]

**return** len(values\_count) *# return the length of values\_count*

dict = {'red':1, 'green':1, 'blue':2} *# given input*

**print** "Unique Values in Dictionary:",count\_values(dict)

**Output:**

Unique Values in Dictionary: 2