PYTHON

```
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *

class MOME:
  keygen=[]
  inicdata="0"

  def __init__(self,filename,inicdata):
    self.keygen=["0","0"]
    self.MOME(filename,inicdata)

  def MOME(self,filename,Input):
    #if self.keygen[1]==Input:#prevenir redundancia
      #return self.keygen[0]#aplicar em aplicacoes reais.
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('-')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]
##
def save_data():
  data=textfield.get()
  if data==None:
    data="Enter"
  textfield.delete(0,END)
  gui_input.set(data)
  gui_output.set(object_1.MOME(filename,data))
  print "output " + gui_output.get()

if __name__=='__main__':

  filename=sys.argv[1]
```

```
  object_1=MOME(filename,"0")

  app = Tk()
  app.title("MOME")
  app.geometry('200x100+200+100')
  #Gui Variables
  gui_output=StringVar()
  gui_output.set(object_1.keygen[0])
  gui_input=StringVar()
  gui_input.set(object_1.keygen[1])

  Label(app,text="Entry: ").pack()

  textfield=Entry(app)
  textfield.pack()

  Button(app,text="Save",command=save_data).pack()

  l1=Label(app,textvariable=gui_output,height=10)
  l1.pack(side='left')

  print "start"

  app.mainloop()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
import pygame.mixer
from Tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  def __init__(self):
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()#class variable
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    self.textfield=Entry(self)#instance variable
    self.textfield.pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=LEFT,padx=2)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")
```

```
class LMome:
  keygen=["0","0"]
  problema="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    if LMome.keygen[1]==Input:
      return LMome.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as causa:
      LMome.problema=causa
      print('Problema %s' % causa)
      return "error"
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        LMome.keygen[0]=item[1]
        LMome.keygen[1]=Input
        f.close()
        break
    return LMome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('370x130+10+10')

    #sounds = pygame.mixer
    #sounds.init()
    #s=sounds.Sound("Weather Girls - Its Raining Men.mp3")
    #wait_finish(s.play())

    #Local variables
    Gui.gui_output=StringVar()
    Gui.problema=StringVar()
    Gui.gui_output.set('0')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      print("Exiting program")
```

```
      sleep(1)
      app.destroy()
      #exit()
    def _Gui__save_data():
      Frame_1.gui_input.set(panel.textfield.get())
      Input=LMome.keygen[0]+":"+Frame_1.gui_input.get()#keyMOME
      print("entrada %s" % Input)
      if Frame_1.gui_input==None:
        Frame_1.gui_input.set("Enter")
      panel.textfield.delete(0,END)
      devolver="saida -> " + LMome().mome(filename,Input)#instance overflow ?
      Gui.gui_output.set(devolver)
      Gui.problema.set(LMome.problema)
      print("output %s" % Gui.gui_output.get())

    panel=Frame_1()
    panel.pack(side=TOP,pady=10)

    l1=Label(app,textvariable=Gui.gui_output,height=1,font=12)
    l1.pack(pady=10)
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=1)
    l2.pack(side='left',pady=2)
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack(side='right',padx=5,pady=5)
    #side='bottom',padx=2,pady=2
    app.protocol("WM_DELETE_WINDOW",shutdown)
    app.mainloop()

if __name__=='__main__':

  filename=sys.argv[1]

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
import pygame.mixer
from Tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  def __init__(self):
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()#class variable
```

```
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    self.textfield=Entry(self)#instance variable
    self.textfield.pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=LEFT,padx=2)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")


class LMome:
  keygen=["0","0"]
  problema="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    if LMome.keygen[1]==Input:
      return LMome.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as causa:
      LMome.problema=causa
      #print('Problema %s' % causa)
      return "error"
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        LMome.keygen[0]=item[1]
        LMome.keygen[1]=Input
        f.close()
        break
    return LMome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('370x130+10+10')

    #sounds = pygame.mixer
    #sounds.init()
    #s=sounds.Sound("Weather Girls - Its Raining Men.mp3")
    #wait_finish(s.play())

    #Local variables
    Gui.gui_output=StringVar()
```

```
    Gui.problema=StringVar()
    Gui.problema.set("status ok")
    Gui.gui_output.set('0')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      Gui.problema.set("Exiting program")
      #print("Exiting program")
      sleep(3)
      app.destroy()
      #exit()
    def _Gui__save_data():
      Frame_1.gui_input.set(panel.textfield.get())
      Input=LMome.keygen[0]+":"+Frame_1.gui_input.get()#keyMOME
      #print("entrada %s" % Input)
      if Frame_1.gui_input==None:
        Frame_1.gui_input.set("Enter")
      panel.textfield.delete(0,END)
      devolver="saida -> " + LMome().mome(filename,Input)#instance overflow ?
      Gui.gui_output.set(devolver)
      Gui.problema.set(LMome.problema)
      #print("output %s" % Gui.gui_output.get())

    panel=Frame_1()
    panel.pack(side=TOP,pady=10)

    l1=Label(app,textvariable=Gui.gui_output,height=1,font=12)
    l1.pack(pady=10)
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=1)
    l2.pack(side='left',pady=2)
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack(side='right',padx=5,pady=5)
    #side='bottom',padx=2,pady=2
    app.protocol("WM_DELETE_WINDOW",shutdown)
    app.mainloop()

if __name__=='__main__':

  filename=sys.argv[1]

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
```

```
#import wx
#import cl
import pygame.mixer
from Tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  def __init__(self):
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()#class variable
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    self.textfield=Entry(self)#instance variable
    self.textfield.pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=LEFT,padx=2)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")


class LMome:
  keygen=["0","0"]
  problema="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    if LMome.keygen[1]==Input:
      return LMome.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as causa:
      LMome.problema=causa
      print 'Problema %s' % causa
      return "error"
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        LMome.keygen[0]=item[1]
        LMome.keygen[1]=Input
        f.close()
        break
    return LMome.keygen[0]

class Gui():
  def __init__(self):
```

```
app = Tk()
app.title("MOME")
app.geometry('370x130+10+10')

#sounds = pygame.mixer
#sounds.init()
#s=sounds.Sound("Weather Girls - Its Raining Men.mp3")
#wait_finish(s.play())

#Local variables
Gui.gui_output=StringVar()
Gui.problema=StringVar()
Gui.gui_output.set('0')
#Funtions
def wait_finish(channel):
  while channel.get_busy():
    pass
def shutdown():
  #if askokcancel(title="are you sure",message="do you realy want to quit"):
  print "Exiting program"
  sleep(1)
  app.destroy()
  #exit()
def _Gui__save_data():
  Frame_1.gui_input.set(panel.textfield.get())
  Input=LMome.keygen[0]+":"+Frame_1.gui_input.get()#keyMOME
  print ("entrada %s" % Input)
  if Frame_1.gui_input==None:
    Frame_1.gui_input.set("Enter")
  panel.textfield.delete(0,END)
  devolver="saida -> " + LMome().mome(filename,Input)#instance overflow ?
  Gui.gui_output.set(devolver)
  Gui.problema.set(LMome.problema)
  print "output " + Gui.gui_output.get()

panel=Frame_1()
panel.pack(side=TOP,pady=10)

l1=Label(app,textvariable=Gui.gui_output,height=1,font=12)
l1.pack(pady=10)
#pady=2,side='left'
l2=Label(app,textvariable=Gui.problema,height=1)
l2.pack(side='left',pady=2)
#side='left',pady=2
Button(app,text="Enter",command=__save_data,width=5).pack(side='right',padx=5,pady=5)
#side='bottom',padx=2,pady=2
app.protocol("WM_DELETE_WINDOW",shutdown)
app.mainloop()
```

```
if __name__=='__main__':

  filename=sys.argv[1]

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
#import pygame.mixer
from tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  def __init__(self):
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()#class variable
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    self.textfield=Entry(self)#instance variable
    self.textfield.pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=LEFT,padx=2)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")


class LMome:
  keygen=["0","0"]
  problema="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    if LMome.keygen[1]==Input:
      return LMome.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as causa:
      LMome.problema=causa
      #print('Problema %s' % causa)
      return "error"
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Input) #Bool
      if keyfound:
```

```
        #MOME UPDATE
        LMome.keygen[0]=item[1]
        LMome.keygen[1]=Input
        f.close()
        break
    return LMome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('370x130+10+10')

    #sounds = pygame.mixer
    #sounds.init()
    #s=sounds.Sound("Weather Girls - Its Raining Men.mp3")
    #wait_finish(s.play())

    #Local variables
    Gui.gui_output=StringVar()
    Gui.problema=StringVar()
    Gui.problema.set("status ok")
    Gui.gui_output.set('0')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      Gui.problema.set("Exiting program")
      #print("Exiting program")
      sleep(3)
      app.destroy()
      #exit()
    def _Gui__save_data():
      Frame_1.gui_input.set(panel.textfield.get())
      Input=LMome.keygen[0]+":"+Frame_1.gui_input.get()#keyMOME
      #print("entrada %s" % Input)
      if Frame_1.gui_input==None:
        Frame_1.gui_input.set("Enter")
      panel.textfield.delete(0,END)
      devolver="saida -> " + LMome().mome(filename,Input)#instance overflow ?
      Gui.gui_output.set(devolver)
      Gui.problema.set(LMome.problema)
      #print("output %s" % Gui.gui_output.get())

    panel=Frame_1()
    panel.pack(side=TOP,pady=10)
```

```
    l1=Label(app,textvariable=Gui.gui_output,height=1,font=12)
    l1.pack(pady=10)
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=1)
    l2.pack(side='left',pady=2)
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack(side='right',padx=5,pady=5)
    #side='bottom',padx=2,pady=2
    app.protocol("WM_DELETE_WINDOW",shutdown)
    app.mainloop()

if __name__=='__main__':

  filename="file-3.txt"
  #sys.argv[1]

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
#import pygame.mixer
from tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  def __init__(self):
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()#class variable
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    self.textfield=Entry(self)#instance variable
    self.textfield.pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=LEFT,padx=2)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")


class LMome:
  keygen=["0","0"]
  problema="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    if LMome.keygen[1]==Input:
```

```
      return LMome.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as causa:
      LMome.problema=causa
      print ('Problema %s' % causa)
      return "error"
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        LMome.keygen[0]=item[1]
        LMome.keygen[1]=Input
        f.close()
        break
    return LMome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('370x130+10+10')

    #sounds = pygame.mixer
    #sounds.init()
    #s=sounds.Sound("Weather Girls - Its Raining Men.mp3")
    #wait_finish(s.play())

    #Local variables
    Gui.gui_output=StringVar()
    Gui.problema=StringVar()
    Gui.gui_output.set('0')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      print("Exiting program")
      sleep(1)
      app.destroy()
      #exit()
    def _Gui__save_data():
      Frame_1.gui_input.set(panel.textfield.get())
      Input=LMome.keygen[0]+":"+Frame_1.gui_input.get()#keyMOME
      print("entrada %s" % Input)
      if Frame_1.gui_input==None:
```

```
        Frame_1.gui_input.set("Enter")
      panel.textfield.delete(0,END)
      devolver="saida -> " + LMome().mome(filename,Input)#instance overflow ?
      Gui.gui_output.set(devolver)
      Gui.problema.set(LMome.problema)
      print("output %s" % Gui.gui_output.get())

    panel=Frame_1()
    panel.pack(side=TOP,pady=10)

    l1=Label(app,textvariable=Gui.gui_output,height=1,font=12)
    l1.pack(pady=10)
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=1)
    l2.pack(side='left',pady=2)
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack(side='right',padx=5,pady=5)
    #side='bottom',padx=2,pady=2
    app.protocol("WM_DELETE_WINDOW",shutdown)
    app.mainloop()

if __name__=='__main__':

  filename=input('Enter file prog : ')

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re

class MOME:
  keygen=[0,0]
  def MOME(self,mem,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    for item in mem:
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        break
    return self.keygen[0]


if __name__=='__main__':
  object_1=MOME()
  mem=[(0,0,0),(0,1,1),(1,0,1)]
```

```
  Input=0
  x=object_1.MOME(mem,Input)
  print x
  print object_1.keygen[0]#!/usr/bin/python2.6 -tt
import sys
import os
import re

class MOME:
  keygen=['0','0']
  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    f=open(filename,'rU')
    for line in f:
      line=line[:-1]
      item=line.split(':')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]


if __name__=='__main__':
  object_1=MOME()
  Hist='0'
  filename='file.txt'
  while True:
    Input=raw_input("enter: ")
    if Input=='sair':
      exit()

    if Input==Hist: #one shot
      continue
    Hist=Input
    ###
    x=object_1.MOME(filename,Input)
    print x
      #!/usr/bin/python2.6 -tt
import sys
import os
import re

class MOME:
  keygen=['0','0']
```

```
  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split(':')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]


if __name__=='__main__':
  object_1=MOME()
  Hist='0'
  filename='file.txt'
  while True:
    Input=raw_input("enter: ")
    if Input=='sair':
      exit()

    if Input==Hist: #one shot
      continue
    Hist=Input
    ###
    x=object_1.MOME(filename,Input)
    print x
      #/usr/bin/python2.6 -tt
import sys
import os
import re

class MOME:
  keygen=['0','0']
  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    try:
      f=open(filename,'r')
    except IOError:
```

```
      print ('IOError',filename)
      exit()
    for line in f:
      line=line[:-1]
      item=line.split(':')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]


if __name__=='__main__':
  object_1=MOME()
  Hist='empty'
  filename='file.txt'
  while True:
    Input=input("Input: ")
    if Input=='sair':
      exit()

    if Input==Hist: #one shot
      continue
    Hist=Input
    ###

    x=object_1.MOME(filename,Input)
    print (x)

#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *

class MOME:
  keygen=['0','0']
  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
```

```
    for line in f:
      line=line[:-1]
      item=line.split(':')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]

if __name__=='__main__':

  object_1=MOME()
  Hist='empty'
  ###
  app = Tk()
  app.title("MOME")
  app.geometry('200x50+50+50')
  ### Gui Variables
  gui_input=StringVar()
  gui_input.set(None)

  gui_output=StringVar()
  gui_output.set(None)
  ###
  filename='file.txt'
  ###

  Label(app,text="Output: ").pack()

  l1=Label(app,textvariable=gui_output)
  l1.pack()

  ###
  while True:
    Input=raw_input("enter: ")
    gui_input.set(Input)
    if Input=='sair':
      exit()
    if Input==Hist: #one shot
      continue
    Hist=Input
    ####
    gui_output.set(object_1.MOME(filename,Input))
    print "saida: %s" % gui_output.get()

  app.mainloop()
```

```
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *

class MOME:
  keygen=['0','0']
  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    try:
      f=open(filename,'r')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split(':')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]
##
def save_data():
  data=textfield.get()
  if data==None:
    data="Enter"
  textfield.delete(0,END)
  gui_input.set(data)
  gui_output.set(object_1.MOME(filename,data))
  print gui_output.get()

if __name__=='__main__':
  object_1=MOME()

  app = Tk()
  app.title("MOME")
  app.geometry('200x100+200+100')
  #Gui Variables
  gui_output=StringVar()
  gui_output.set(None)
  gui_input=StringVar()
  gui_input.set(None)
```

```
  filename='file.txt'

  Label(app,text="Entry: ").pack()

  textfield=Entry(app)
  textfield.pack()

  Button(app,text="Save",command=save_data).pack()

  l1=Label(app,textvariable=gui_output,height=10)
  l1.pack(side='left')

  print "start"

  app.mainloop()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *

class MOME:
  keygen=['0','0']
  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split(':')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]

if __name__=='__main__':

  object_1=MOME()
  Hist='empty'
```

```
  ###
  app = Tk()
  app.title("MOME")
  app.geometry('200x80+50+50')
  ### Gui Variables
  gui_input=StringVar()
  gui_input.set(None)

  gui_output=StringVar()
  gui_output.set(None)
  ###
  filename='file.txt'
  ###
  Label(app,text="Input: ").pack()

  l1=Label(app,textvariable=gui_input)
  l1.pack()

  Label(app,text="Output: ").pack()

  l2=Label(app,textvariable=gui_output)
  l2.pack()

  ###
  while True:
    Input=raw_input("enter: ")
    gui_input.set(Input)
    if Input=='sair':
      exit()
    if Input==Hist: #one shot
      continue
    Hist=Input
    ###
    gui_output.set(object_1.MOME(filename,Input))
    print "saida: %s" % gui_output.get()

  app.mainloop()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *

class MOME:
  keygen=['0','0']
  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
```

```
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split(':')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]

def leave_func():
  exit()

if __name__=='__main__':

  object_1=MOME()
  Hist='empty'
  ###
  app = Tk()
  app.title("MOME")
  app.geometry('200x100+50+50')
  ### Gui Variables
  gui_input=StringVar()
  gui_input.set(None)

  gui_output=StringVar()
  gui_output.set(None)
  ###
  filename='file.txt'
  ###
  Label(app,text="Input: ").pack()

  l1=Label(app,textvariable=gui_input)
  l1.pack()

  Label(app,text="Output: ").pack()

  l2=Label(app,textvariable=gui_output)
  l2.pack()

  Button(app,text="Sair",command=leave_func).pack()
```

```
  ###
  while True:
    Input=raw_input("enter: ")
    gui_input.set(Input)
    if Input==Hist: #one shot
      continue
    Hist=Input
    ###
    gui_output.set(object_1.MOME(filename,Input))
    print "saida: %s" % gui_output.get()

  app.mainloop()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *

class MOME:
  keygen=[]
  inicdata="0"

  def __init__(self,filename,inicdata):
    self.keygen=["0","0"]
    self.MOME(filename,inicdata)

  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('-')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]
##
def save_data():
  data=textfield.get()
```

```
  if data==None:
    data="Enter"
  textfield.delete(0,END)
  gui_input.set(data)
  gui_output.set(object_1.MOME(filename,data))
  print "output " + gui_output.get()

if __name__=='__main__':

  filename=sys.argv[1]

  object_1=MOME(filename,"0")

  app = Tk()
  app.title("MOME")
  app.geometry('200x100+200+100')
  #Gui Variables
  gui_output=StringVar()
  gui_output.set(object_1.keygen[0])
  gui_input=StringVar()
  gui_input.set(object_1.keygen[1])

  Label(app,text="Entry: ").pack()

  textfield=Entry(app)
  textfield.pack()

  Button(app,text="Save",command=save_data).pack()

  l1=Label(app,textvariable=gui_output,height=10)
  l1.pack(side='left')

  print "start"

  app.mainloop()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *

class MOME:
  keygen=[]
  inicdata="0"
  def __init__(self,filename,inicdata):
    self.keygen=["0","0"]
    self.MOME(filename,inicdata)
  def MOME(self,filename,Input):
```

```
    #if self.keygen[1]==Input:
      #return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('-')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]

class GUI:
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('200x100+200+100')
    self.gui_output=StringVar()
    self.gui_input=StringVar()
    self.gui_output.set("empty")
    self.gui_input.set("empty")
    Label(app,text="Entry: ").pack()
    self.textfield=Entry(app)
    self.textfield.pack()
    Button(app,text="Enter",command=self.save_data).pack()
    l1=Label(app,textvariable=self.gui_output,height=10)
    l1.pack(side='left')
    app.mainloop()

  def save_data(self):
    data=self.textfield.get()
    if data==None:
      data="Enter"
    self.textfield.delete(0,END)
    self.gui_input.set(data)
    self.gui_output.set(x.MOME(filename,data))
    #object_1.MOME(filename,data)
    print "output " + self.gui_output.get()

if __name__=='__main__':

  filename=sys.argv[1]
```

```
  x=MOME(filename,"0")

  app_1=GUI()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
#import pygame.mixer
from Tkinter import *

class Mome:
  keygen=["0","0"]
  def mome(self,filename,Input):
          #previne redundancia real
    #if self.keygen[1]==Input:
      #return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('-')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('200x100+200+100')
    Gui.gui_output=StringVar()
    #self.gui_input=StringVar()
    Gui.gui_output.set("empty")
    #self.gui_input.set("empty")
    Label(app,text="Entry: ").pack()
    Gui.textfield=Entry(app)
    Gui.textfield.pack()
    Button(app,text="Enter",command=__save_data).pack()
```

```
    l1=Label(app,textvariable=Gui.gui_output,height=10)
    l1.pack(side='left')
    app.mainloop()

def _Gui__save_data():
  data=Gui.textfield.get()
  print "entrada %s" % data
  if data==None:
    data="Enter"
  Gui.textfield.delete(0,END)
  Gui.gui_output.set(Mome().mome(filename,data))#instance overflow ?
  print "output " + Gui.gui_output.get()

if __name__=='__main__':

  filename=sys.argv[1]

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
#import pygame.mixer
from Tkinter import *

class Mome:
  keygen=["0","0"]
  def mome(self,filename,Input):
          #previne redundancia real
    #if self.keygen[1]==Input:
      #return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]
```

```
class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('200x100+200+100')
    Gui.gui_output=StringVar()
    #self.gui_input=StringVar()
    Gui.gui_output.set("empty")
    #self.gui_input.set("empty")
    Label(app,text="Entry: ").pack()
    Gui.textfield=Entry(app)
    Gui.textfield.pack()
    Button(app,text="Enter",command=__save_data).pack()
    l1=Label(app,textvariable=Gui.gui_output,height=10)
    l1.pack(side='left')
    app.mainloop()

def _Gui__save_data():
  data=Gui.textfield.get()
  #print "entrada %s" % data
  if data==None:
    data="Enter"
  Gui.textfield.delete(0,END)
  Gui.gui_output.set(Mome().mome(filename,data))#instance overflow ?
  print "output " + Gui.gui_output.get()

if __name__=='__main__':

  filename=sys.argv[1]

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
import pygame.mixer
from Tkinter import *
from time import sleep
#import Tkinter.messagebox

#Global Methods
def _Gui__save_data():
  Gui.gui_input.set(Gui.textfield.get())
  #print "entrada %s" % data
  if Gui.gui_input==None:
    Gui.gui_input.set("Enter")
```

```
  Gui.textfield.delete(0,END)
  devolver="saida -> " + Mome().mome(filename,Gui.gui_input.get()) #instance overflow ?
  Gui.gui_output.set(devolver)
  Gui.problema.set(Mome.causa)
  print "output " + Gui.gui_output.get()

class Mome:
  keygen=["0","0"]
  causa="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    #if Mome.keygen[1]==Input:
      #return self.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as Mome.causa:
      print 'Problema %s' % Mome.causa
      return "1"
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Mome.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        Mome.keygen[0]=item[2]
        Mome.keygen[1]=Input
        f.close()
        break
    return Mome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('370x150+10+10')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      print "Exiting program"
      sleep(1)
      app.destroy()
      exit()
    #Local variables
    Gui.gui_output=StringVar()
    Gui.gui_input=StringVar()
    Gui.problema=StringVar()
```

```
    Gui.gui_output.set(None)
    Gui.gui_input.set(None)
    Gui.problema.set(None)
    #
    Label(app,text="Entry: ").pack()
    #side=LEFT
    Gui.textfield=Entry(app)
    Gui.textfield.pack()
    #side=LEFT
    l3=Label(app,textvariable=Gui.gui_input,height=0)
    l3.pack()
    #side=RIGHT
    Gui.textfield.insert(0,"empty")
    l1=Label(app,textvariable=Gui.gui_output,height=0,font=12)
    l1.pack()
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=0)
    l2.pack()
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack()
    #side='bottom',padx=10,pady=10
    app.protocol("WM_DELETE_WINDOW",shutdown)
    app.mainloop()

if __name__=='__main__':

  #sounds = pygame.mixer
  #sounds.init()
  #s=sounds.Sound("Stevie Wonder - I Just Called To Say I Love You.mp3")
  #wait_finish(s.play())

  filename=sys.argv[1]

  app_1=Gui()

#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
import pygame.mixer
from Tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  print "inside frame"
  def __init__(self):
```

```
    print "inside frame_1"
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()
    Frame_1.gui_input.set(None)
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=10)
    #side=LEFT
    Frame_1.textfield=Entry(self)
    Frame_1.textfield.pack(side=LEFT,padx=10,pady=20)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=RIGHT,padx=10)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")

#Global Methods outer scope, use only class atributes,
#for it does not have scope to instances atributes.
def _Gui__save_data():
  Frame_1.gui_input.set(Frame_1.textfield.get())
  #print "entrada %s" % data
  if Frame_1.gui_input==None:
    Frame_1.gui_input.set("Enter")
  Frame_1.textfield.delete(0,END)
  devolver="saida -> " + Mome().mome(filename,Frame_1.gui_input.get()) #instance overflow ?
  Gui.gui_output.set(devolver)
  Gui.problema.set(Mome.causa)
  print "output " + Gui.gui_output.get()

class Mome:
  keygen=["0","0"]
  causa="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    #if Mome.keygen[1]==Input:
      #return self.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as Mome.causa:
      print 'Problema %s' % Mome.causa
      return "error"
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Mome.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        Mome.keygen[0]=item[2]
        Mome.keygen[1]=Input
        f.close()
        break
```

```
    return Mome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('370x150+10+10')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      print "Exiting program"
      sleep(1)
      app.destroy()
      exit()
    #Local variables
    Gui.gui_output=StringVar()
    Gui.problema=StringVar()
    Gui.gui_output.set(None)
    Gui.problema.set(None)
    #
    panel=Frame_1()
    panel.pack()

    l1=Label(app,textvariable=Gui.gui_output,height=0,font=12)
    l1.pack()
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=0)
    l2.pack()
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack()
    #side='bottom',padx=10,pady=10
    app.protocol("WM_DELETE_WINDOW",shutdown)
    app.mainloop()

if __name__=='__main__':

  #sounds = pygame.mixer
  #sounds.init()
  #s=sounds.Sound("Stevie Wonder - I Just Called To Say I Love You.mp3")
  #wait_finish(s.play())

  filename=sys.argv[1]

  app_1=Gui()

#!/usr/bin/python2.6 -tt
```

```
import sys
import os
import re
#import wx
#import cl
import pygame.mixer
from Tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  print "inside frame"
  def __init__(self):
    print "inside frame_1"
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    Frame_1.textfield=Entry(self)
    Frame_1.textfield.pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=LEFT,padx=2)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")

#Global Methods outer scope, use only class atributes,
#for it does not have scope to instances atributes.
def _Gui__save_data():
  Frame_1.gui_input.set(Frame_1.textfield.get())
  #print "entrada %s" % data
  if Frame_1.gui_input==None:
    Frame_1.gui_input.set("Enter")
  Frame_1.textfield.delete(0,END)
  devolver="saida -> " + Mome().mome(filename,Frame_1.gui_input.get()) #instance overflow ?
  Gui.gui_output.set(devolver)
  Gui.problema.set(Mome.causa)
  print "output " + Gui.gui_output.get()

class Mome:
  keygen=["0","0"]
  causa="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    #if Mome.keygen[1]==Input:
      #return self.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as Mome.causa:
```

```
      print 'Problema %s' % Mome.causa
      return "error"
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Mome.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        Mome.keygen[0]=item[2]
        Mome.keygen[1]=Input
        f.close()
        break
    return Mome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('350x100+10+10')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      print "Exiting program"
      sleep(1)
      app.destroy()
      exit()
    #Local variables
    Gui.gui_output=StringVar()
    Gui.problema=StringVar()
    Gui.gui_output.set('0')
    #
    panel=Frame_1()
    panel.pack(side=TOP)

    l1=Label(app,textvariable=Gui.gui_output,height=1,font=12)
    l1.pack(pady=2)
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=1)
    l2.pack(side='left',pady=2)
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack(side='right',padx=2,pady=2)
    #side='bottom',padx=2,pady=2
    app.protocol("WM_DELETE_WINDOW",shutdown)
    app.mainloop()

if __name__=='__main__':
```

```
  #sounds = pygame.mixer
  #sounds.init()
  #s=sounds.Sound("Stevie Wonder - I Just Called To Say I Love You.mp3")
  #wait_finish(s.play())

  filename=sys.argv[1]

  app_1=Gui()

#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
import pygame.mixer
from Tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  def __init__(self):
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()#class variable
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    self.textfield=Entry(self)#instance variable
    self.textfield.pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=LEFT,padx=2)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")


class Mome:
  keygen=["0","0"]
  problema="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    if Mome.keygen[1]==Input:
      return Mome.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as causa:
      Mome.problema=causa
      print 'Problema %s' % causa
      return "error"
```

```
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Mome.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        Mome.keygen[0]=item[2]
        Mome.keygen[1]=Input
        f.close()
        break
    return Mome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('350x100+10+10')

    #sounds = pygame.mixer
    #sounds.init()
    #s=sounds.Sound("Weather Girls - Its Raining Men.mp3")
    #wait_finish(s.play())

    #Local variables
    Gui.gui_output=StringVar()
    Gui.problema=StringVar()
    Gui.gui_output.set('0')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      print "Exiting program"
      sleep(1)
      app.destroy()
      #exit()
    def _Gui__save_data():
      Frame_1.gui_input.set(panel.textfield.get())
      #print "entrada %s" % data
      if Frame_1.gui_input==None:
        Frame_1.gui_input.set("Enter")
      panel.textfield.delete(0,END)
      devolver="saida -> " + Mome().mome(filename,Frame_1.gui_input.get()) #instance overflow ?
      Gui.gui_output.set(devolver)
      Gui.problema.set(Mome.problema)
      print "output " + Gui.gui_output.get()

    panel=Frame_1()
```

```
    panel.pack(side=TOP)

    l1=Label(app,textvariable=Gui.gui_output,height=1,font=12)
    l1.pack(pady=2)
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=1)
    l2.pack(side='left',pady=2)
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack(side='right',padx=2,pady=2)
    #side='bottom',padx=2,pady=2
    app.protocol("WM_DELETE_WINDOW",shutdown)
    app.mainloop()

if __name__=='__main__':

  filename=sys.argv[1]

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
#import pygame.mixer
from tkinter import *

class Mome:
  keygen=["0","0"]
  def mome(self,filename,Input):
          #previne redundancia real
    #if self.keygen[1]==Input:
      #return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print ('IOError %s' % filename)
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]
```

```
class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('200x100+200+100')
    Gui.gui_output=StringVar()
    #self.gui_input=StringVar()
    Gui.gui_output.set("empty")
    #self.gui_input.set("empty")
    Label(app,text="Entry: ").pack()
    Gui.textfield=Entry(app)
    Gui.textfield.pack()
    Button(app,text="Enter",command=__save_data).pack()
    l1=Label(app,textvariable=Gui.gui_output,height=10)
    l1.pack(side='left')
    app.mainloop()

def _Gui__save_data():
  data=Gui.textfield.get()
  #print ("entrada %s" % data)
  if data==None:
    data="Enter"
  Gui.textfield.delete(0,END)
  Gui.gui_output.set(Mome().mome(filename,data))#instance overflow ?
  print ("output %s" % Gui.gui_output.get())

if __name__=='__main__':

  filename="file-2.txt"

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import wx
#import cl
#import pygame.mixer
from tkinter import *

class Mome:
  keygen=["0","0"]
  def mome(self,filename,Input):
          #previne redundancia real
    #if self.keygen[1]==Input:
      #return self.keygen[0]
    try:
      f=open(filename,'rU')
```

```
    except IOError:
      print ('IOError %s' % filename)
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('200x100+200+100')
    Gui.gui_output=StringVar()
    #self.gui_input=StringVar()
    Gui.gui_output.set("empty")
    #self.gui_input.set("empty")
    Label(app,text="Entry: ",font=12).pack()
    Gui.textfield=Entry(app)
    Gui.textfield.pack()
    Button(app,text="Enter",command=__save_data,font=10).pack()
    l1=Label(app,textvariable=Gui.gui_output,height=12,font=12)
    l1.pack(side='left')
    app.mainloop()

def _Gui__save_data():
  data=Gui.textfield.get()
  #print ("entrada %s" % data)
  if data==None:
    data="Enter"
  Gui.textfield.delete(0,END)
  Gui.gui_output.set(Mome().mome(filename,data))#instance overflow ?
  print ("output %s" % Gui.gui_output.get())

if __name__=='__main__':

  filename="file-2.txt"

  app_1=Gui()
#!/usr/bin/python2.6 -tt
import sys
import os
```

```
import re
#import wx
#import cl
#import pygame.mixer
from tkinter import *
from time import sleep
#import Tkinter.messagebox

class Frame_1(Frame):
  def __init__(self):
    Frame.__init__(self)
    Frame_1.gui_input=StringVar()
    Label(self,text="Entry: ",font=12).pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    Frame_1.textfield=Entry(self)
    Frame_1.textfield.pack(side=LEFT,padx=2,pady=2)
    #side=LEFT
    l1=Label(self,textvariable=Frame_1.gui_input,height=0,font=12)
    l1.pack(side=LEFT,padx=2)
    #side=RIGHT
    #Frame_1.textfield.insert(0,"empty")

#Global Methods outer scope, use only class atributes,
#for it does not have scope to instances atributes.
def _Gui__save_data():
  Frame_1.gui_input.set(Frame_1.textfield.get())
  #print "entrada %s" % data
  if Frame_1.gui_input==None:
    Frame_1.gui_input.set("Enter")
  Frame_1.textfield.delete(0,END)
  devolver="saida -> " + Mome().mome(filename,Frame_1.gui_input.get()) #instance overflow ?
  Gui.gui_output.set(devolver)
  Gui.problema.set(Mome.problema)
  print ("output %s" % Gui.gui_output.get())

class Mome:
  keygen=["0","0"]
  problema="status ok"
  def mome(self,filename,Input):
          #previne redundancia real
    if Mome.keygen[1]==Input:
      return Mome.keygen[0]
    try:
      f=open(filename,'rU')
    except Exception as causa:
      Mome.problema=causa
      print ('Problema %s' % causa)
      return "error"
    for line in f:
```

```
      line=line[:-1]
      item=line.split('    ')
      keyfound=(item[0]==Mome.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        Mome.keygen[0]=item[2]
        Mome.keygen[1]=Input
        f.close()
        break
    return Mome.keygen[0]

class Gui():
  def __init__(self):
    app = Tk()
    app.title("MOME")
    app.geometry('350x100+10+10')
    #Funtions
    def wait_finish(channel):
      while channel.get_busy():
        pass
    def shutdown():
      #if askokcancel(title="are you sure",message="do you realy want to quit"):
      print ("Exiting program")
      sleep(1)
      app.destroy()
      #exit()

    #sounds = pygame.mixer
    #sounds.init()
    #s=sounds.Sound("Weather Girls - Its Raining Men.mp3")
    #wait_finish(s.play())

    #Local variables
    Gui.gui_output=StringVar()
    Gui.problema=StringVar()
    Gui.gui_output.set('0')
    #
    panel=Frame_1()
    panel.pack(side=TOP)

    l1=Label(app,textvariable=Gui.gui_output,height=1,font=12)
    l1.pack(pady=2)
    #pady=2,side='left'
    l2=Label(app,textvariable=Gui.problema,height=1)
    l2.pack(side='left',pady=2)
    #side='left',pady=2
    Button(app,text="Enter",command=__save_data,width=5).pack(side='right',padx=2,pady=2)
    #side='bottom',padx=2,pady=2
    app.protocol("WM_DELETE_WINDOW",shutdown)
```

```
    app.mainloop()

if __name__=='__main__':

  filename=input("Enter File name ?\n")
  #sys.argv[1]

  app_1=Gui()

#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *
import pygame.mixer
import sqlite3
#import pySerial

class MOME():
  keygen=[]
  inicdata="0"

  def __init__(self,filename,inicdata):
    self.keygen=["0","0"]
    self.MOME(filename,inicdata)

  def MOME(self,filename,Input):
    if self.keygen[1]==Input:
      return self.keygen[0]
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('-')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]

def leave_func():
  exit()
```

```
if __name__=='__main__':

  filename=sys.argv[1]
  ###
  object_1=MOME(filename,"5")#create instance
  ###
  Hist='empty'
  sounds=pygame.mixer
  sounds.init()
  sound_1=sounds.Sound("wrong.wav")


  ### Window APP
  app = Tk()
  app.title("MOME")
  app.geometry('200x100+50+50')
  ### Gui Variables
  gui_input=StringVar()
  gui_input.set(None)
  #sound_1.play()

  gui_output=StringVar()
  gui_output.set(None)
  ###
  Label(app,text="Input: ").pack()

  l1=Label(app,textvariable=gui_input)
  l1.pack()
  gui_input.set(object_1.keygen[1])

  Label(app,text="Output: ").pack()

  l2=Label(app,textvariable=gui_output)
  l2.pack()
  gui_output.set(object_1.keygen[0])

  Button(app,text="Sair",command=leave_func).pack()

  ###
  while True:
    Input=raw_input("enter: ")
    gui_input.set(Input)
    if Input==Hist: #one shot
      continue
    Hist=Input
    gui_output.set(object_1.MOME(filename,Input))
    print "saida: %s" % gui_output.get()
```

```
  app.mainloop()
#!/usr/bin/python2.6 -tt
import sys
import os
import re
#import pygame.mixer
from Tkinter import *

class MOME:
  keygen=[]
  inicdata="0"

  def __init__(self,filename,inicdata):
    self.keygen=["0","0"]
    self.MOME(filename,inicdata)

  def MOME(self,filename,Input):
    #if self.keygen[1]==Input:#prevenir redundancia
      #return self.keygen[0]#aplicar em aplicacoes reais.
    try:
      f=open(filename,'rU')
    except IOError:
      print 'IOError',filename
      exit()
    for line in f:
      line=line[:-1]
      item=line.split('-')
      keyfound=(item[0]==self.keygen[0] and item[1]==Input) #Bool
      if keyfound:
        #MOME UPDATE
        self.keygen[0]=item[2]
        self.keygen[1]=Input
        f.close()
        break
    return self.keygen[0]
##
def save_data():
  data=textfield.get()
  if data==None:
    data="Enter"
  textfield.delete(0,END)
  gui_input.set(data)
  gui_output.set(object_1.MOME(filename,data))
  print "output " + gui_output.get()

if __name__=='__main__':

  filename=sys.argv[1]
```

```
  object_1=MOME(filename,"0")

  app = Tk()
  app.title("MOME")
  app.geometry('200x100+200+100')
  #Gui Variables
  gui_output=StringVar()
  gui_output.set(object_1.keygen[0])
  gui_input=StringVar()
  gui_input.set(object_1.keygen[1])

  Label(app,text="Entry: ").pack()

  textfield=Entry(app)
  textfield.pack()

  Button(app,text="Enter",command=save_data).pack()

  l1=Label(app,textvariable=gui_output,height=10)
  l1.pack(side='left')

  print "start"

  app.mainloop()
0         0         0
0         1         1
1         0         2
2         1         0
1         1         0
0         2         0
0         ola       ola sergio
ola sergio        sair      0
0:0       0
0:1       1
1:0       2
2:1       0
1:1       0
0:2       0
0:ola     ola sergio
ola sergio:sair 0
0:ligar motor 1           motor 1 on
motor 1 on:desligar motor 1   motor 1 off
motor 1 off:sair          0

0:0:0
0:1:1
1:0:1
1:1:0
0:ola:ola chefe
```

ola chefe:exit:0
0:botoneira:motor_1
motor_1:0:motor_1
motor_1:botoneira:0

0-0-0
0-1-1
1-0-2
2-1-0
1-1-0
0-2-0
0 0 0
0 1 1
1 0 2
2 1 0
1 1 0
0 2 0
0 ola ola sergio
ola sergio sair 0
0:0 0
0:1 1
1:0 2
2:1 0
1:1 0
0:2 0
0:ola ola sergio
ola sergio:sair 0
0:ligar motor 1 motor 1 on
motor 1 on:desligar motor 1 motor 1 off
motor 1 off:sair 0

0:0:0
0:1:1
1:0:1
1:1:0
0:ola:ola chefe
ola chefe:exit:0
0:botoneira:motor_1
motor_1:0:motor_1
motor_1:botoneira:0