import os, sys, re, tim

import Tkinter as tk

import MySQLdb

frame, lw, num\_commodities, response\_label, commodities\_list, commodities\_dict = None, None, None, None, [], {}

def compute\_price():

global frame, num\_commodities, response\_label

// LABEL

response\_label = tk.Label(frame)

selnum = lw.curselection()

if not selnum or len(selnum) == 0:

response\_label = tk.Label(frame)

response\_label["text"] = "Select a commodity"

response\_label.pack()

return None

cmdt\_name = commodities\_list[int(selnum[0])]

cmdt\_attrib\_list = commodities\_dict[cmdt\_name]

price = cmdt\_attrib\_list[1]

count\_cmdt = num\_commodities.get()

total\_price = int(count\_cmdt) \* float(price)

if response\_label:

response\_label.destroy()

response\_label = tk.Label(frame)

response\_label["text"] = str(total\_price)

response\_label.pack()

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

win = tk.Tk()

# First, create a frame:

frame = tk.Frame(win)

scrollbar = None

# Next, create a scrollbar .

#scrollbar=tk.Scrollbar(frame, bg='black')

#scrollbar.pack(side='right', expand='yes', fill='y')

# Manipulate the scrollbar here... I am leaving this as an exercise for you.

dbconn = MySQLdb.connect("localhost", "<username>", "<password>", "guitest")

sql = "SELECT Name, Id, pricePerUnit FROM commodities"

dbcursor = dbconn.cursor()

dbcursor.execute(sql)

commodities\_dict = {}

allrecs = dbcursor.fetchall()

for rec in allrecs:

Name = rec[0]

ID = rec[1]

Price = rec[2]

commodities\_dict[commodity\_name] = [ ID, Price ]

lw = tk.Listbox(frame, selectmode=tk.SINGLE,

yscrollcommand=scrollbar, height=5)

indx = 0

commodities\_list = commodities\_dict.keys()

for Name in commodities\_list:

lw.insert(indx, Name)

indx += 1

lw.pack()

//LABEL

prompt\_label = tk.Label(frame, text="Enter Quantity")

prompt\_label.pack()

// ENTRY

num\_commodities = tk.Entry(frame)

num\_commodities.pack()

// BUTTON TO COMPUTE

compute\_btn = tk.Button(frame, text="Compute Price",

command=compute\_price)

compute\_btn.pack()

frame.pack()

win.mainloop()