BAHAGIAN PEPERIKSAAN DAN PENILAIAN
JABATAN PENDIDIKAN POLITEKNIK DAN KOLEJ KOMUNITI
KEMENTERIAN PENDIDIKAN TINGGI

JABATAN TEKNOLOGI MAKLUMAT DAN KOMUNIKASI

PEPERIKSAAN AKHIR

SESI II : 2022/2023

DFP40203: PYTHON PROGRAMMING

Jenis item	Politeknik Pengubal	Bil. Perlu Gubal	Bilangan Set
Subjektif : 3 Soalan	PSP	1 Set	Set 1

JANGAN BUKA KERTAS SOALAN INI SEHINGGA DIBERITAHU OLEH KETUA PENGAWAS

1. Your responsibility as a programmer employed at Lelemove, a shipping company, is to create an application capable of determining the cost of parcels based on their dimensions. During your investigation of the company's code repositories, you came across the previous programmer's main program (main.py). Your objective is to finalize the program, enhancing Lelemove's software for calculating parcel prices. The pricing table for parcels is outlined below:

Weight (kg)	Volume (cm^3)	Price ($)
0-1	0-5000	3
0-1	5001-10,000	5
0-1	>10,000	7
1-5	0-5000	5
1-5	5001-10,000	7
1-5	>10,000	9
>5	0-5000	7
>5	5001-10,000	9
>5	>10,000	11

Formula to calculate volume is

Volume = length x width x height

```
from parcel_calculator import calculate_price

# Get user input for parcel dimensions and weight
length = float(input("Enter parcel length in centimeters: "))
width = float(input("Enter parcel width in centimeters: "))
height = float(input("Enter parcel height in centimeters: "))
weight = float(input("Enter parcel weight in kilograms: "))

# Calculate parcel price using imported function
price = calculate_price(length, width, height, weight)

# Print parcel price to user
print("The price of your parcel is: $", price)
```

main.py

Read the question below to guide you to complete the program.

CLO1 P3 i. Build the *main.py* program. Do you encounter any error? (give screenshot) **(2 Marks)**

CLO1 P1 ii. Describe the meaning of this error? **(3 Marks)**

CLO1 P4 iii. Constructs another module to complete the *main.py* program and save length,width,height and weight of parcel on text file. **(10 Marks)**

2. During your exploration of the company's code repositories, you stumbled upon another program that remains unfinished. This time, your objective is to develop a Graphical User Interface (GUI) for the program, enabling it to calculate parcel prices. Apply your programming expertise and understanding to finalize the program.

```
Import  x  as tk
import parcel_calculator

class   x   :
   def __init__(self, master):
      self.master = master
      master.title("Parcel Calculator")

      # Create length input label and entry field
      self.length_label = tk.Label(master, text="Length (cm):")
      self.length_label.grid(row=0, column=0)
      self.length_entry = tk.Entry(master)
      self.length_entry.grid(row=0, column=1)

      # Create width input label and entry field
      self.width_label = tk.Label(master, text="Width (cm):")
      self.width_label.grid(row=1, column=0)
      self.width_entry = tk.Entry(master)
```

```
        self.width_entry.grid(row=1, column=1)

        # Create height input label and entry field
        self.height_label = tk.Label(master, text="    x   ")
        self.height_label.grid(row=2, column=0)
        self.height_entry = tk.Entry(master)
        self.height_entry.grid(row=2, column=1)

        # Create weight input label and entry field
        self.weight_label = tk.Label(master, text="Weight (kg):")
        self.weight_label.grid(row=3, column=0)
        self.weight_entry = tk.Entry(master)
        self.weight_entry.grid(row=3, column=1)

        # Create calculate button
        self.calculate_button = tk.Button(master, text="Calculate", command=self.   x  )
        self.calculate_button.grid(row=4, column=0, columnspan=2)

        # Create price label
        self.price_label = tk.Label(master, text="")
        self.price_label.grid(row=5, column=0, columnspan=2)

    def calculate_price(self):
        # Get parcel dimensions and weight from entry fields
        length = float(self.length_entry.get())
        width = float(self.width_entry.get())
        height = float(self.height_entry.get())
        weight = float(self.weight_entry.get())


        # Calculate parcel price using imported function
        price = parcel_calculator.calculate_price(     x    ,    x    ,    x    ,    x     )

        # Update price label with calculated price
        self.price_label.config(text="The price of your parcel is: $" + str(     x     ))

    x     = tk.Tk()
    parcel_calculator_gui = ParcelCalculatorGUI(a)
    a.mainloop()
```

gui_parcel.py

CLO1 P4

i. Manipulates x with your answer to complete the program above to get a GUI version of the program. **(10 Marks)**

CLO1 P4

ii. Upon completing the program, you noticed that the graphical user interface (GUI) lacked visual appeal, potentially hindering user experience. Your assignment is to construct at least 3 additional GUI features to create a more user-friendly interface. **(10 Marks)**
i.e additional message box, pull up button, color variations, layout and etc.

3. The development of the parcel shipping calculation program is nearing completion. The remaining task involves implementing a feature to store all shipment data in a database, utilizing any TinySQL/SQL system of your choice. To accomplish this, your responsibility is to create an additional program specifically designed to capture and record the relevant data. **(15 marks)**

CLO1 P4 i. Builds a database named Lelemove System.

CLO1 P4 ii. Make a table with given attributes as in table below with item_id as the primary key.

Attribute	Data type
Item_id	Varchar(10)
Item_height	Varchar(5)
Item_width	Varchar(5)
Item_length	Varchar(5)
Item_volume	Varchar(10)
Item_price	Varchar(5)

CLO1 P4 iii. Constructs a button to add all the data given by the user to the database.

CLO1 P4 iv. Perform the exception handling (message box) to notify the user that the data given has been stored.

i.e