

```

from tkinter import *
import random
import os
from tkinter import messagebox
import mysql.connector

# Replace these placeholders with your actual database credentials
db_host = "localhost"
db_user = "billing_user"
db_password = "password"
db_name = "bill_system"

try:
    db_connection = mysql.connector.connect(
        host=db_host,
        user=db_user,
        password=db_password,
        database=db_name
    )
    cursor = db_connection.cursor()
    print("Connected to MySQL database")
except mysql.connector.Error as err:
    print(f"Error: {err}")

# Create the 'bills' table if it doesn't exist
try:
    cursor.execute('''CREATE TABLE IF NOT EXISTS bills (
        bill_number INT AUTO_INCREMENT PRIMARY KEY,
        bill_data TEXT
    )''')
    print("Table 'bills' created or already exists.")
except mysql.connector.Error as err:
    print(f"Error creating table: {err}")

class Bill_App:
    def __init__(self, root):
        welcome_label = Label(root, text="Namasthe...! Welcome To Billing Software...@", font=('Bradley Hand ITC', 16, 'bold'), fg="dark blue")
        welcome_label.pack()
        self.root = root
        self.root.geometry("1400x700+0+0")
        self.root.title("Billing Software")
        bg_color = "rosybrown2"
        title = Label(self.root, text="Billing Software", bd=12, relief=GROOVE,
            bg=bg_color, fg="black", font=("times new roman", 25, "bold"),
            pady=2).pack(fill=X)
        watermark_label = Label(root, text="By RRGs, Sainath Lal R, Prajwal A J
            From Final Year(BCA)", font=('arial', 14, "bold"), fg="khaki",
            bg="black")

```

```

watermark_label.pack(side=BOTTOM, fill=X)
root.protocol("WM_DELETE_WINDOW", self.on_exit)
# Variables
self.cust_name = StringVar()
self.cust_phone = StringVar()
self.bill_no = StringVar()
#=====For Generating Random Bill Numbers=====
x = random.randint(1000, 9999)
self.bill_no.set(str(x))
self.search_bill = StringVar()
# =====Cosmetics Details=====
self.soap = IntVar()
self.face_cream = IntVar()
self.face_wash = IntVar()
self.hair_spray = IntVar()
self.hair_gel = IntVar()
self.body_lotion = IntVar()
# =====Grocery Details=====
self.rice = IntVar()
self.food_oil = IntVar()
self.daal = IntVar()
self.wheat = IntVar()
self.sugar = IntVar()
self.tea = IntVar()
# =====Cool Drinks Details=====
self.maza = IntVar()
self.coke = IntVar()
self.frooti = IntVar()
self.thumbs_up = IntVar()
self.limca = IntVar()
self.sprite = IntVar()
# =====Total=====
self.cosmetic_price = StringVar()
self.grocery_price = StringVar()
self.cold_drink_price = StringVar()
# =====Customer=====
self.c_name = StringVar()
self.c_phone = StringVar()
self.bill_no = StringVar()
x = random.randint(1000, 9999)
self.bill_no.set(str(x))
self.search_bill = StringVar()
# =====Tax=====
self.cosmetic_tax = StringVar()
self.grocery_tax = StringVar()
self.cold_drink_tax = StringVar()

# Customer Details Frame
F1 = LabelFrame(self.root, bd=10, relief=GROOVE, text="Customer Details",
font=("times new roman", 15, "bold"),

```

```

        fg="red3", bg=bg_color)
F1.place(x=0, y=80, relwidth=1)

    cname_lbl = Label(F1, text="Customer Name", bg=bg_color, fg="black",
font=("times new roman", 15, "bold")).grid(
        row=0, column=0, padx=20, pady=5)
    cname_txt = Entry(F1, width=15, textvariable=self.cust_name, font=("Bradley
Hand ITC", 15, "bold"), bd=7, relief=SUNKEN).grid(row=0,

        column=1,

        padx=10,

        pady=5)

    cphn_lbl = Label(F1, text="Phone Number", bg=bg_color, fg="black",
font=("times new roman", 15, "bold")).grid(
        row=0, column=2, padx=20, pady=5)
    cphn_txt = Entry(F1, width=15, textvariable=self.cust_phone, font="arial
15", bd=7, relief=SUNKEN).grid(row=0,

        column=3,

        padx=10,

        pady=5)

    c_bill_lbl = Label(F1, text="Bill Number", bg=bg_color, fg="black",
        font=("times new roman", 15, "bold")).grid(row=0,
column=4, padx=20, pady=5)
    c_bill_txt = Entry(F1, width=15, textvariable=self.search_bill, font="arial
15", bd=7, relief=SUNKEN).grid(row=0,

        column=5,

        padx=10,

        pady=5)

    bill_btn = Button(F1, text="Search", width=10, pady=5, font=("times new
roman", 12, "bold"), bd=7, command=self.find_bill).grid(
        row=0, column=6, padx=10, pady=10)

    # Cosmetic Details Frame
    F2 = LabelFrame(self.root, bd=10, relief=GROOVE, text="Cosmetic Details",
font=("times new roman", 15, "bold"),
        fg="red3", bg=bg_color)
    F2.place(x=5, y=185, width=325, height=380)

    bath_lbl = Label(F2, text="Bath Soap", font=("Bradley Hand ITC", 18,

```

```

"bold"), bg=bg_color, fg="black").grid(
    row=0, column=0, padx=10, pady=10, sticky="w")
    bath_txt = Entry(F2, width=10, textvariable=self.soap, font=("times new
roman", 18, "bold"), bd=5, relief=SUNKEN).grid(
    row=0, column=1, padx=10, pady=10)

    face_cream_lbl = Label(F2, text="Face Cream", font=("Bradley Hand ITC", 18,
"bold"), bg=bg_color,
                                fg="black").grid(row=1, column=0, padx=10, pady=10,
sticky="w")
    face_cream_txt = Entry(F2, width=10, textvariable=self.face_cream,
font=("times new roman", 16, "bold"), bd=5,
                                relief=SUNKEN).grid(row=1, column=1, padx=10,
pady=10)

    face_wash_lbl = Label(F2, text="Face Wash", font=("Bradley Hand ITC", 18,
"bold"), bg=bg_color,
                                fg="black").grid(row=2, column=0, padx=10, pady=10,
sticky="w")
    face_wash_txt = Entry(F2, width=10, textvariable=self.face_wash,
font=("times new roman", 16, "bold"), bd=5,
                                relief=SUNKEN).grid(row=2, column=1, padx=10,
pady=10)

    hair_spray_lbl = Label(F2, text="Hair Spray", font=("Bradley Hand ITC", 18,
"bold"), bg=bg_color,
                                fg="black").grid(row=3, column=0, padx=10, pady=10,
sticky="w")
    hair_spray_txt = Entry(F2, width=10, textvariable=self.hair_spray,
font=("times new roman", 16, "bold"), bd=5,
                                relief=SUNKEN).grid(row=3, column=1, padx=10,
pady=10)

    hair_gel_lbl = Label(F2, text="Hair Gel", font=("Bradley Hand ITC", 18,
"bold"), bg=bg_color,
                                fg="black").grid(row=4, column=0, padx=10, pady=10,
sticky="w")
    hair_gel_txt = Entry(F2, width=10, textvariable=self.hair_gel, font=("times
new roman", 16, "bold"), bd=5,
                                relief=SUNKEN).grid(row=4, column=1, padx=10, pady=10)

    body_lotion_lbl = Label(F2, text="Body Lotion", font=("Bradley Hand ITC",
18, "bold"), bg=bg_color,
                                fg="black").grid(row=5, column=0, padx=10, pady=10,
sticky="w")
    body_lotion_txt = Entry(F2, width=10, textvariable=self.body_lotion,
font=("times new roman", 16, "bold"), bd=5,
                                relief=SUNKEN).grid(row=5, column=1, padx=10,
pady=10)

```

```

# Grocery Details Frame
F3 = LabelFrame(self.root, bd=10, relief=GROOVE, text="Grocery Details",
font=("times new roman", 15, "bold"),
fg="red3", bg=bg_color)
F3.place(x=340, y=185, width=325, height=380)

g1_lbl = Label(F3, text="Rice", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
row=0, column=0, padx=10, pady=10, sticky="w")
g1_txt = Entry(F3, width=10, textvariable=self.rice, font=("times new
roman", 16, "bold"), bd=5, relief=SUNKEN).grid(
row=0, column=1, padx=10, pady=10)

g2_lbl = Label(F3, text="Food Oil", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color,
fg="black").grid(row=1, column=0, padx=10, pady=10,
sticky="w")
g2_txt = Entry(F3, width=10, textvariable=self.food_oil, font=("times new
roman", 16, "bold"), bd=5,
relief=SUNKEN).grid(row=1, column=1, padx=10, pady=10)

g3_lbl = Label(F3, text="Daal", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
row=2, column=0, padx=10, pady=10, sticky="w")
g3_txt = Entry(F3, width=10, textvariable=self.daal, font=("times new
roman", 16, "bold"), bd=5, relief=SUNKEN).grid(
row=2, column=1, padx=10, pady=10)

g4_lbl = Label(F3, text="Wheat", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
row=3, column=0, padx=10, pady=10, sticky="w")
g4_txt = Entry(F3, width=10, textvariable=self.wheat, font=("times new
roman", 16, "bold"), bd=5,
relief=SUNKEN).grid(row=3, column=1, padx=10, pady=10)

g5_lbl = Label(F3, text="Sugar", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
row=4, column=0, padx=10, pady=10, sticky="w")
g5_txt = Entry(F3, width=10, textvariable=self.sugar, font=("times new
roman", 16, "bold"), bd=5, relief=SUNKEN).grid(
row=4, column=1, padx=10, pady=10)

g6_lbl = Label(F3, text="Tea", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
row=5, column=0, padx=10, pady=10, sticky="w")
g6_txt = Entry(F3, width=10, textvariable=self.tea, font=("times new
roman", 16, "bold"), bd=5, relief=SUNKEN).grid(
row=5, column=1, padx=10, pady=10)

# Cold Drink Details Frame

```

```

F4 = LabelFrame(self.root, bd=10, relief=GROOVE, text="Cold Drink Details",
                 font=("times new roman", 15, "bold"), fg="red3",
bg=bg_color)
F4.place(x=670, y=185, width=325, height=380)

c1_lbl = Label(F4, text="Maza", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
    row=0, column=0, padx=10, pady=10, sticky="w")
c1_txt = Entry(F4, width=10, textvariable=self.maza, font=("times new
roman", 16, "bold"), bd=5, relief=SUNKEN).grid(
    row=0, column=1, padx=10, pady=10)

c2_lbl = Label(F4, text="Coke", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
    row=1, column=0, padx=10, pady=10, sticky="w")
c2_txt = Entry(F4, width=10, textvariable=self.coke, font=("times new
roman", 16, "bold"), bd=5, relief=SUNKEN).grid(
    row=1, column=1, padx=10, pady=10)

c3_lbl = Label(F4, text="Frooti", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color,
                 fg="black").grid(row=2, column=0, padx=10, pady=10,
sticky="w")
c3_txt = Entry(F4, width=10, textvariable=self.frooti, font=("times new
roman", 16, "bold"), bd=5,
                 relief=SUNKEN).grid(row=2, column=1, padx=10, pady=10)

c4_lbl = Label(F4, text="Thumbs Up", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color,
                 fg="black").grid(row=3, column=0, padx=10, pady=10,
sticky="w")
c4_txt = Entry(F4, width=10, textvariable=self.thumbs_up, font=("times new
roman", 16, "bold"), bd=5,
                 relief=SUNKEN).grid(row=3, column=1, padx=10, pady=10)

c5_lbl = Label(F4, text="Limca", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
    row=4, column=0, padx=10, pady=10, sticky="w")
c5_txt = Entry(F4, width=10, textvariable=self.limca, font=("times new
roman", 16, "bold"), bd=5, relief=SUNKEN).grid(
    row=4, column=1, padx=10, pady=10)

c6_lbl = Label(F4, text="Sprite", font=("Bradley Hand ITC", 18, "bold"),
bg=bg_color, fg="black").grid(
    row=5, column=0, padx=10, pady=10, sticky="w")
c6_txt = Entry(F4, width=10, textvariable=self.sprite, font=("times new
roman", 16, "bold"), bd=5, relief=SUNKEN).grid(
    row=5, column=1, padx=10, pady=10)

# Bill Area

```

```

F5 = Frame(self.root, bd=10, relief=GROOVE)
F5.place(x=1010, y=185, width=400, height=380)
bill_title = Label(F5, text="Bill Area", font="arial 15 bold", bd=7,
relief=GROOVE).pack(fill=X)
scrol_y = Scrollbar(F5, orient=VERTICAL)
self.txtarea = Text(F5, yscrollcommand=scrol_y.set)
scrol_y.pack(side=RIGHT, fill=Y)
scrol_y.config(command=self.txtarea.yview)
self.txtarea.pack(fill=BOTH, expand=1)

# Button Frame
F6 = LabelFrame(self.root, bd=10, relief=GROOVE, text="Bill Menu",
font=("times new roman", 15, "bold"),
fg="red2", bg=bg_color)
F6.place(x=0, y=565, relwidth=1, height=140)

m1_lbl = Label(F6, text="Total Cosmetic Price", bg=bg_color, fg="black",
font=("times new roman", 14, "bold")).grid(row=0, column=0,
padx=20, pady=1, sticky="w")
m1_txt = Entry(F6, width=18, textvariable=self.cosmetic_price, font=("times
new roman", 10, "bold"), bd=7,
relief=SUNKEN).grid(row=0, column=1, padx=10, pady=1)

m2_lbl = Label(F6, text="Total Grocery Price", bg=bg_color, fg="black",
font=("times new roman", 14, "bold")).grid(row=1, column=0,
padx=20, pady=1, sticky="w")
m2_txt = Entry(F6, width=18, textvariable=self.grocery_price, font=("times
new roman", 10, "bold"), bd=7,
relief=SUNKEN).grid(row=1, column=1, padx=10, pady=1)

m3_lbl = Label(F6, text="Total Cold Drink Price", bg=bg_color, fg="black",
font=("times new roman", 14, "bold")).grid(row=2, column=0,
padx=20, pady=1, sticky="w")
m3_txt = Entry(F6, width=18, textvariable=self.cold_drink_price,
font=("times new roman", 10, "bold"), bd=7,
relief=SUNKEN).grid(row=2, column=1, padx=10, pady=1)

c1_lbl = Label(F6, text="Cosmetic Tax", bg=bg_color, fg="black",
font=("times new roman", 14, "bold")).grid(row=0, column=2,
padx=20, pady=1, sticky="w")
c1_txt = Entry(F6, width=18, textvariable=self.cosmetic_tax, font=("times
new roman", 10, "bold"), bd=7,
relief=SUNKEN).grid(row=0, column=3, padx=10, pady=1)

c2_lbl = Label(F6, text="Grocery Tax", bg=bg_color, fg="black",
font=("times new roman", 14, "bold")).grid(row=1, column=2,
padx=20, pady=1, sticky="w")
c2_txt = Entry(F6, width=18, textvariable=self.grocery_tax, font=("times
new roman", 10, "bold"), bd=7,
relief=SUNKEN).grid(row=1, column=3, padx=10, pady=1)

```

```

c3_lbl = Label(F6, text="Cold Drink Tax", bg=bg_color, fg="black",
               font=("times new roman", 14, "bold")).grid(row=2, column=2,
padx=20, pady=1, sticky="w")
c3_txt = Entry(F6, width=18, textvariable=self.cold_drink_tax, font=("times
new roman", 10, "bold"), bd=7,
               relief=SUNKEN).grid(row=2, column=3, padx=10, pady=1)

btn_F = Frame(F6, bd=7, relief=GROOVE)
btn_F.place(x=750, width=700, height=105)

total_btn = Button(btn_F, command=self.total, text="Total",
bg="rosybrown2", fg="black", pady=15, width=10,
               font=("times new roman", 14, "bold")).grid(row=0,
column=0, padx=5, pady=5)
generate_bill_btn = Button(btn_F, text="Generate Bill",
command=self.bill_area, bg="rosybrown2", fg="black",
               pady=15, width=10, font=("times new roman", 14,
"bold")).grid(row=0, column=1, padx=5,

               pady=5)

delete_btn = Button(btn_F, text="Delete Bill", command=self.delete_bill,
bg="rosybrown2", fg="black", pady=15,
               width=10, font=("times new roman", 14, "bold"))
delete_btn.grid(row=0, column=5, padx=5, pady=5)

clear_btn = Button(btn_F, text="Clear", command=self.clear_data,
bg="rosybrown2", fg="black", pady=15, width=10,
               font=("times new roman", 14, "bold")).grid(row=0,
column=2, padx=5, pady=5)

exit_btn = Button(btn_F, text="Exit", command=self.exit_app,
bg="rosybrown2", fg="black", pady=15, width=10,
               font=("times new roman", 14, "bold")).grid(row=0,
column=4, padx=5, pady=5)

self.welcome_bill()

def total(self):
    self.c_s_p = self.soap.get() * 40
    self.c_fc_p = self.face_cream.get() * 120
    self.c_fw_p = self.face_wash.get() * 60
    self.c_hs_p = self.hair_spray.get() * 180
    self.c_hg_p = self.hair_gel.get() * 140
    self.c_bl_p = self.body_lotion.get() * 180

    self.total_cosmetic_price = float(
        self.c_s_p +
        self.c_fc_p +

```



```

        self.c_fw_p +
        self.c_hs_p +
        self.c_hg_p +
        self.c_bl_p
    )
    self.cosmetic_price.set("Rs. " + str(self.total_cosmetic_price))
    self.c_tax = round((self.total_cosmetic_price * 0.05), 2)
    self.cosmetic_tax.set("Rs. " + str(self.c_tax))

    self.g_r_p = self.rice.get() * 80
    self.g_fo_p = self.food_oil.get() * 180
    self.g_d_p = self.daal.get() * 60
    self.g_w_p = self.wheat.get() * 240
    self.g_s_p = self.sugar.get() * 45
    self.g_t_p = self.tea.get() * 150

    self.total_grocery_price = float(
        self.g_r_p +
        self.g_fo_p +
        self.g_d_p +
        self.g_w_p +
        self.g_s_p +
        self.g_t_p
    )
    self.grocery_price.set("Rs. " + str(self.total_grocery_price))
    self.g_tax = round((self.total_grocery_price * 0.1), 2)
    self.grocery_tax.set("Rs. " + str(self.g_tax))

    self.cd_m_p = self.maza.get() * 30
    self.cd_c_p = self.coke.get() * 30
    self.cd_f_p = self.frooti.get() * 25
    self.cd_t_p = self.thumbs_up.get() * 40
    self.cd_l_p = self.limca.get() * 25
    self.cd_s_p = self.sprite.get() * 40

    self.total_cold_drink_price = float(
        self.cd_m_p +
        self.cd_c_p +
        self.cd_f_p +
        self.cd_t_p +
        self.cd_l_p +
        self.cd_s_p
    )
    self.cold_drink_price.set("Rs. " + str(self.total_cold_drink_price))
    self.cd_tax = round((self.total_cold_drink_price * 0.05), 2)
    self.cold_drink_tax.set("Rs. " + str(self.cd_tax))

def welcome_bill(self):
    self.txtarea.delete('1.0', END)
    self.txtarea.insert(END, f"    --Welcome to Retail Billing Software-- \n")

```

```

self.txtarea.insert(END, f" *****\n")
self.txtarea.insert(END, f"\n Bill Number : {self.bill_no.get()}")
self.txtarea.insert(END, f"\n Customer Name : {self.cust_name.get()}")
self.txtarea.insert(END, f"\n Phone Number : {self.cust_phone.get()}")
self.txtarea.insert(END, f"\n *****\n")
self.txtarea.insert(END, f"\n Products\t\tQty\t\tPrice")
self.txtarea.insert(END, f"\n*****")

def bill_area(self):
    if self.cust_name.get() == "" or self.cust_phone.get() == "":
        messagebox.showerror("Error", "Customer details are incomplete")
    elif self.cosmetic_price.get() == "Rs. 0.0" and self.grocery_price.get() ==
"Rs. 0.0" and self.cold_drink_price.get() == "Rs. 0.0":
        messagebox.showerror("Error", "No product selected")
    else:
        self.welcome_bill()
        # Cosmetic Bill
        if self.soap.get() != 0:
            self.txtarea.insert(END, f"\n Bath
Soap\t\t{self.soap.get()}\t\t{self.c_s_p}")
            if self.face_cream.get() != 0:
                self.txtarea.insert(END, f"\n Face
Cream\t\t{self.face_cream.get()}\t\t{self.c_fc_p}")
            if self.face_wash.get() != 0:
                self.txtarea.insert(END, f"\n Face
Wash\t\t{self.face_wash.get()}\t\t{self.c_fw_p}")
            if self.hair_spray.get() != 0:
                self.txtarea.insert(END, f"\n Hair
Spray\t\t{self.hair_spray.get()}\t\t{self.c_hs_p}")
            if self.hair_gel.get() != 0:
                self.txtarea.insert(END, f"\n Hair
Gel\t\t{self.hair_gel.get()}\t\t{self.c_hg_p}")
            if self.body_lotion.get() != 0:
                self.txtarea.insert(END, f"\n Body
Lotion\t\t{self.body_lotion.get()}\t\t{self.c_bl_p}")

        # Grocery Bill
        if self.rice.get() != 0:
            self.txtarea.insert(END, f"\n
Rice\t\t{self.rice.get()}\t\t{self.g_r_p}")
            if self.food_oil.get() != 0:
                self.txtarea.insert(END, f"\n Food
Oil\t\t{self.food_oil.get()}\t\t{self.g_fo_p}")
            if self.daal.get() != 0:
                self.txtarea.insert(END, f"\n
Daal\t\t{self.daal.get()}\t\t{self.g_d_p}")
            if self.wheat.get() != 0:
                self.txtarea.insert(END, f"\n
Wheat\t\t{self.wheat.get()}\t\t{self.g_w_p}")
            if self.sugar.get() != 0:

```

```

        self.txtarea.insert(END, f"\n
Sugar\t\t{self.sugar.get()}\t\t{self.g_s_p}")
        if self.tea.get() != 0:
            self.txtarea.insert(END, f"\n
Tea\t\t{self.tea.get()}\t\t{self.g_t_p}")

        # Cold Drink Bill
        if self.maza.get() != 0:
            self.txtarea.insert(END, f"\n
Maza\t\t{self.maza.get()}\t\t{self.cd_m_p}")
        if self.coke.get() != 0:
            self.txtarea.insert(END, f"\n
Coke\t\t{self.coke.get()}\t\t{self.cd_c_p}")
        if self.frooti.get() != 0:
            self.txtarea.insert(END, f"\n
Frooti\t\t{self.frooti.get()}\t\t{self.cd_f_p}")
        if self.thumbs_up.get() != 0:
            self.txtarea.insert(END, f"\n Thumbs
Up\t\t{self.thumbs_up.get()}\t\t{self.cd_t_p}")
        if self.limca.get() != 0:
            self.txtarea.insert(END, f"\n
Limca\t\t{self.limca.get()}\t\t{self.cd_l_p}")
        if self.sprite.get() != 0:
            self.txtarea.insert(END, f"\n
Sprite\t\t{self.sprite.get()}\t\t{self.cd_s_p}")

        self.txtarea.insert(END, "\n-----")
        if self.cosmetic_tax.get() != "Rs. 0.0":
            self.txtarea.insert(END, f"\n Cosmetic
Tax\t\t\t{self.cosmetic_tax.get()}")
        if self.grocery_tax.get() != "Rs. 0.0":
            self.txtarea.insert(END, f"\n Grocery
Tax\t\t\t{self.grocery_tax.get()}")
        if self.cold_drink_tax.get() != "Rs. 0.0":
            self.txtarea.insert(END, f"\n Cold Drink
Tax\t\t\t{self.cold_drink_tax.get()}")

        self.txtarea.insert(END, "\n*****")
        self.txtarea.insert(END, f"\n Total Bill : \t\t\tRs.
{str(self.total_cosmetic_price + self.total_grocery_price +
self.total_cold_drink_price + self.c_tax + self.g_tax + self.cd_tax)}")
        self.txtarea.insert(END, "\n*****")
        self.save_bill()

    def save_bill(self):
        # Get the bill data from the text widget
        bill_data = self.txtarea.get("1.0", "end-1c")

        # Insert the bill into the 'bills' table
        cursor.execute("INSERT INTO bills (bill_data) VALUES (%s)", (bill_data,))

```

```

db_connection.commit()
messagebox.showinfo("Success", "Bill has been saved successfully.")

def find_bill(self):
    present = "no"
    try:
        mycursor = db_connection.cursor()
        mycursor.execute("SELECT * FROM bills WHERE bill_number = %s",
(self.search_bill.get(),))
        result = mycursor.fetchone()

        if result:
            self.txtarea.delete('1.0', END)
            self.txtarea.insert(END, result[1]) # Assuming the bill data is
stored in the second column of the table
            present = "yes"
        except mysql.connector.Error as err:
            # Handle the database error
            print(f"Database error: {err}")
            messagebox.showerror("Database Error", "An error occurred while
fetching the bill data")

        if present == "no":
            messagebox.showerror("Error", "Invalid Bill Number")

def clear_data(self):
    op = messagebox.askyesno("Clear", "Do you really want to clear the data?")
    if op > 0:
        self.soap.set(0)
        self.face_cream.set(0)
        self.face_wash.set(0)
        self.hair_spray.set(0)
        self.hair_gel.set(0)
        self.body_lotion.set(0)

        self.rice.set(0)
        self.food_oil.set(0)
        self.daal.set(0)
        self.wheat.set(0)
        self.sugar.set(0)
        self.tea.set(0)

        self.maza.set(0)
        self.coke.set(0)
        self.frooti.set(0)
        self.thumbs_up.set(0)
        self.limca.set(0)
        self.sprite.set(0)

```

```

        self.cosmetic_price.set("")
        self.grocery_price.set("")
        self.cold_drink_price.set("")

        self.cosmetic_tax.set("")
        self.grocery_tax.set("")
        self.cold_drink_tax.set("")

        self.cust_name.set("")
        self.cust_phone.set("")
        x = random.randint(1000, 9999)
        self.bill_no.set(str(x))
        self.search_bill.set("")
        self.welcome_bill()

def delete_bill(self):
    bill_number = self.search_bill.get()
    if bill_number:
        try:
            # Delete the bill from the 'bills' table using the bill number
            cursor.execute("DELETE FROM bills WHERE bill_number = %s",
(bill_number,))
            db_connection.commit()
            messagebox.showinfo("Success", "Bill has been deleted
successfully.")
            self.clear_data() # Clear the bill data after deleting
        except mysql.connector.Error as err:
            # Handle the database error
            print(f"Database error: {err}")
            messagebox.showerror("Database Error", "An error occurred while
deleting the bill")
        else:
            messagebox.showerror("Error", "Please enter a valid Bill Number to
delete.")

def on_exit(self):
    # Thank you message
    messagebox.showinfo("Thank You", "Thank you for using this Software! ....
🙏")
    root.destroy()

def exit_app(self):
    op = messagebox.askyesno("Exit", "Do you really want to exit?")
    if op > 0:
        self.root.destroy()

if __name__ == "__main__":
    root = Tk()
    obj = Bill_App(root)
    root.mainloop()

```