

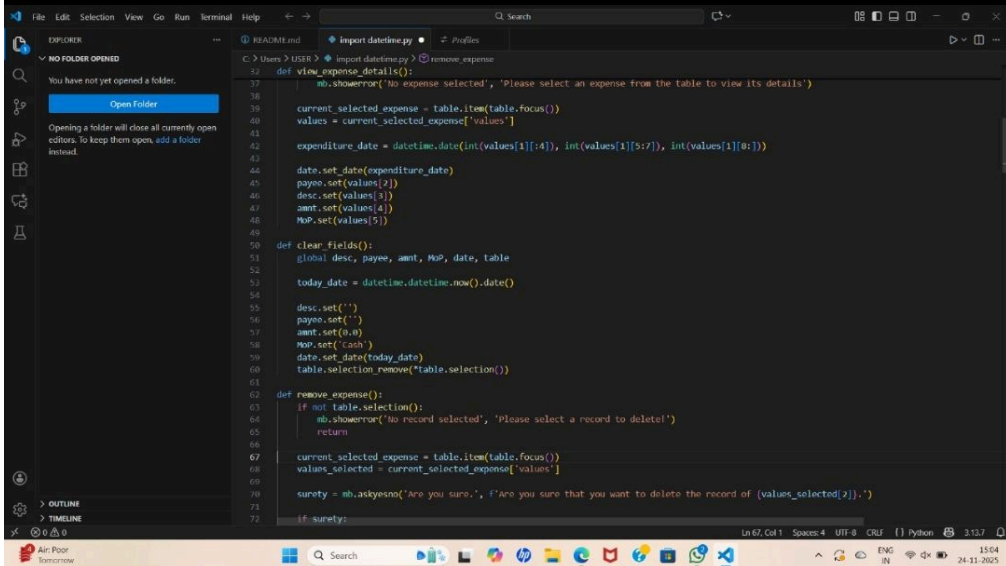
```
File Edit Selection View Go Run Terminal Help
import datetime
import sqlite3
from tkcalendar import DateEntry
from tkinter import *
import tkinter.messagebox as mb
import tkinter.ttk as ttk

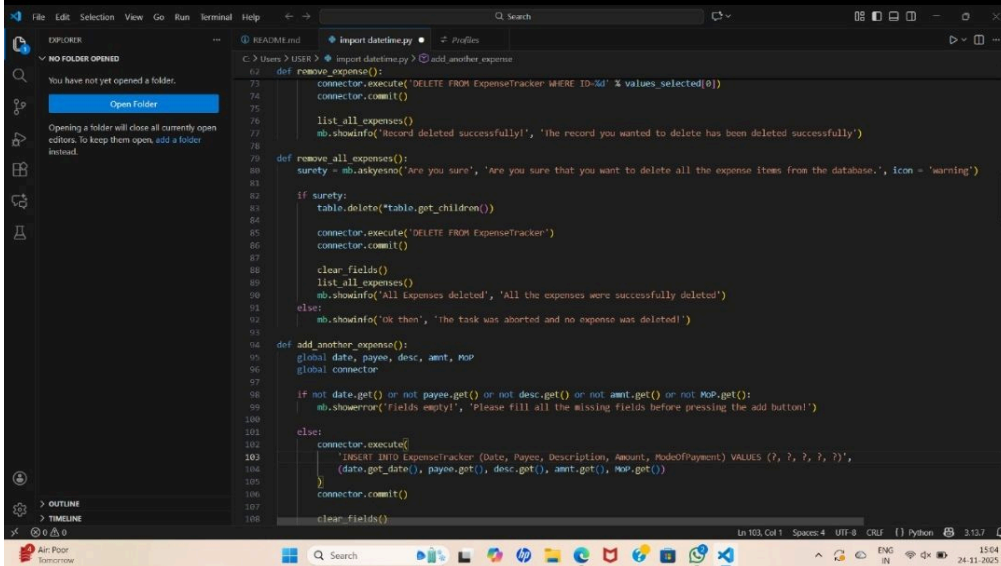
connector = sqlite3.connect('Expense Tracker.db')
cursor = connector.cursor()

connector.execute(
    'CREATE TABLE IF NOT EXISTS Expensetracker (ID INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, Date DATETIME, Payee TEXT, Description TEXT, Amount REAL)'
)
connector.commit()

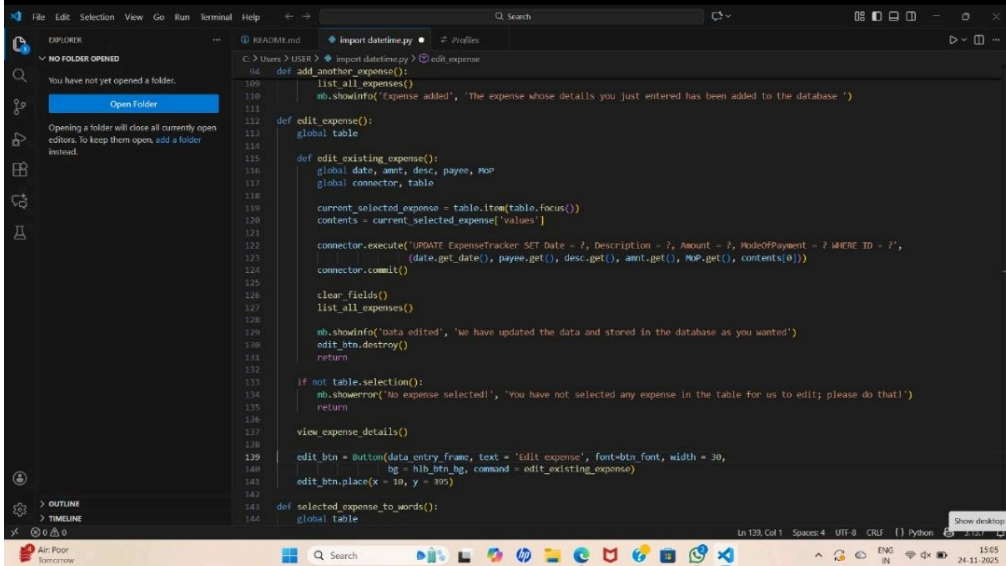
def list_all_expenses():
    global connector, table
    table.delete(*table.get_children())
    all_data = connector.execute('SELECT * FROM Expensetracker')
    data = all_data.fetchall()
    for values in data:
        table.insert('', END, values=values)

def view_expense_details():
    global table
    global date, payee, desc, amt, PUP
    if not table.selection():
        mb.showerror('No expense selected', 'Please select an expense from the table to view its details!')
```





```
62 def remove_expense():
73     connector.execute("DELETE FROM Expensetracker WHERE ID=?& % values_selected[0])
74     connector.commit()
75
76     list_all_expenses()
77     mb.showinfo("Record deleted successfully!", "The record you wanted to delete has been deleted successfully")
78
79 def remove_all_expenses():
80     surety = mb.askyesno("Are you sure", "Are you sure that you want to delete all the expense items from the database.", icon = "warning")
81
82     if surety:
83         table.delete(table.get_children())
84
85         connector.execute("DELETE FROM Expensetracker")
86         connector.commit()
87
88         clear_fields()
89         list_all_expenses()
90         mb.showinfo("All Expenses deleted", "All the expenses were successfully deleted")
91     else:
92         mb.showinfo("Ok then", "The task was aborted and no expense was deleted")
93
94 def add_another_expense():
95     global date, payee, desc, amt, mop
96     global connector
97
98     if not date.get() or not payee.get() or not desc.get() or not amt.get() or not mop.get():
99         mb.showerror("Fields empty!", "Please fill all the missing fields before pressing the add button!")
100
101     else:
102         connector.execute(
103             "INSERT INTO Expensetracker (Date, Payee, Description, Amount, ModeOfPayment) VALUES (?, ?, ?, ?, ?)",
104             (date.get_date(), payee.get(), desc.get(), amt.get(), mop.get())
105         )
106         connector.commit()
107
108     clear_fields()
```



```
File Edit Selection View Go Run Terminal Help
C:\Users\USER> import datetime.py
def selected_expense_to_words():
    if not table.selection():
        mb.showerror('No expense selected!', 'Please select an expense from the table for us to read')
        return
    current_selected_expense = table.item(table.focus())
    values = current_selected_expense['values']
    message = f'Your expense can be read like, \n" You paid {values[4]} to {values[2]} for {values[1]} via {values[5]}"'
    mb.showinfo('Here\'s how to read your expense', message)
def expense_to_words_before_adding():
    global date, desc, amt, payee, MoP
    if not date or not desc or not amt or not payee or not MoP:
        mb.showerror('Incomplete data!', 'The data is incomplete, meaning fill all the fields first!')
    message = f'Your expense can be read like, \n" You paid {amt.get()} to {payee.get()} for {desc.get()} on {date.get_date()} via {MoP.get()}'
    add_question = mb.askyesno('Read your record like .', f'{message}\n\nshould i add it to the database .')
    if add_question:
        add_another_expense()
    else:
        mb.showinfo('OK', 'Please take your time to add this record')
def entry_frame_bg = 'red'
def buttons_frame_bg = 'tomato'
def hlb_btn_bg = 'indianred'
hlb_font = ('Georgia', 14)
entry_font = ('Times 12 bold')
btn_font = ('Gill Sans MT', 13)
```

```
File Edit Selection View Go Run Terminal Help
C:\Users\USER> import datetime.py ...

180
181
182 root = Tk()
183 root.title("PythonGeeks Expense Tracker")
184 root.geometry("1200x550")
185 root.resizable(0,0)
186
187 Label(root, text= "EXPENSE TRACKER", font= ("Noto Sans CJK TC", 15, "bold"), bg = hlb_btn_bg).pack(side = TOP, fill = X)
188
189
190 desc = StringVar()
191 amt = DoubleVar()
192 payee = StringVar()
193 MOP = StringVar(value = "Cash")
194
195
196
197 data_entry_frame = Frame(root, bg = dataentry_frame_bg)
198 data_entry_frame.place(x=0, y=0, relheight=0.95, relwidth=0.25)
199
200 buttons_frame = Frame(root, bg = buttons_frame_bg)
201 buttons_frame.place(relx=0.25, rely=0.05, relwidth=0.75, relheight=0.21)
202
203 tree_frame = Frame(root)
204 tree_frame.place(relx=0.25, rely=0.26, relwidth=0.75, relheight=0.74)
205
206
207
208 Label(data_entry_frame, text = "Date (MM/DD/YYYY) :", font=lbl_font,bg=dataentry_frame_bg).place(x=10,y=230)
209 date_entry = Entry(data_entry_frame, text=date=datetime.datetime.now().date(), font=entry_font)
210 date_entry.place(x=160,y=50)
211 Label(data_entry_frame, text="Payee:", font=lbl_font, bg=dataentry_frame_bg).place(x=10,y=230)
212 entry(data_entry_frame, font=entry_font, width=41, textvariable=payee).place(x=10,y=260)
213 Label(data_entry_frame, text="Description :", font=lbl_font,bg=dataentry_frame_bg).place(x=10,y=100)
214 entry(data_entry_frame, font=entry_font, width=41, textvariable=desc).place(x=10,y=130)
215 Label(data_entry_frame, text="Amount:", font=lbl_font,bg=dataentry_frame_bg).place(x=10,y=180)
216 entry(data_entry_frame, font=entry_font, width=14, textvariable=amt).place(x=160,y=180)
```

```
File Edit Selection View Go Run Terminal Help
C:\Users\USER> import datetime.py
217 Label(data_entry_frame, text="Mode of payment:", font=lbl_font, bg=dataentry_frame.bg).place(x=10, y=310)
218 ddi.optionsmenu(data_entry_frame, POP, ["Cash", "Cheque", "Credit card", "Debit card", "Paytm", "Google pay", "Razor pay"])
219 ddi.place(x=100, y=300)
220 ddi.configure(width=10, font=entry_font)
221
222
223 Button(data_entry_frame, text="Add expense", command=add_another_expense, font=btn_font, width=30,
224         bg=hlb_btn_bg).place(x=10, y=85)
225 Button(data_entry_frame, text="Convert to words before adding", font=btn_font, width=30, bg=hlb_btn_bg).place(x=10, y=400)
226
227
228 Button(buttons_frame, text="Delete Expense", font=btn_font, width=25, bg=hlb_btn_bg, command=remove_expense).place(x=30, y=5)
229
230 Button(buttons_frame, text="Clear Fields in DataEntry Frame", font=btn_font, width=25, bg=hlb_btn_bg,
231         command=clear_fields).place(x=335, y=5)
232
233 Button(buttons_frame, text="Delete All Expenses", font=btn_font, width=25, bg=hlb_btn_bg, command=remove_all_expenses).place(x=640, y=5)
234
235 Button(buttons_frame, text="View Selected Expense's Details", font=btn_font, width=25, bg=hlb_btn_bg,
236         command=view_expense_details).place(x=30, y=65)
237
238 Button(buttons_frame, text="Edit Selected expense", command=edit_expense, font=btn_font, width=25, bg=hlb_btn_bg).place(x=335, y=65)
239
240 Button(buttons_frame, text="Convert Expense to a sentence", font=btn_font, width=25, bg=hlb_btn_bg,
241         command=selected_expense_to_words).place(x=640, y=65)
242
243
244 table = ttk.Treeview(tree_frame, selectmode=BROWSE, columns=('ID', 'Date', 'Payee', 'Description', 'Amount', 'Mode of Payment'))
245
246 X_Scroller = Scrollbar(tree_frame, orient=HORIZONTAL, command=table.xview)
247 Y_Scroller = Scrollbar(tree_frame, orient=VERTICAL, command=table.yview)
248 X_Scroller.pack(side=BOTTOM, fill=X)
249 Y_Scroller.pack(side=RIGHT, fill=Y)
250
251 table.config(yscrollcommand=Y_Scroller.set, xscrollcommand=X_Scroller.set)
252
253 table.heading('ID', text='S.No.', anchor=CENTER)
```

```
File Edit Selection View Go Run Terminal Help
C:\Users\USER> import datetime.py
command-selected_expense.to_words).place(x=640, y=65)

241
242
243
244 table = ttk.Treeview(tree_frame, selectmode=BROWSE, columns=('ID', 'Date', 'Payee', 'Description', 'Amount', 'Mode of Payment'))
245
246 X_Scroller = Scrollbar(tree_frame, orient=HORIZONTAL, command=table.xview)
247 Y_Scroller = Scrollbar(tree_frame, orient=VERTICAL, command=table.yview)
248 X_Scroller.pack(side=LEFT, fill=X)
249 Y_Scroller.pack(side=RIGHT, fill=Y)
250
251 table.config(yscrollcommand=Y_Scroller.set, xscrollcommand=X_Scroller.set)
252
253 table.heading('ID', text='S No.', anchor=CENTER)
254 table.heading('Date', text='Date', anchor=CENTER)
255 table.heading('Payee', text='Payee', anchor=CENTER)
256 table.heading('Description', text='Description', anchor=CENTER)
257 table.heading('Amount', text='Amount', anchor=CENTER)
258 table.heading('Mode of Payment', text='Mode of Payment', anchor=CENTER)
259
260 table.column('ID', width=50, stretch=NO)
261 table.column('Date', width=50, stretch=NO)
262 table.column('Payee', width=90, stretch=NO)
263 table.column('Description', width=150, stretch=NO)
264 table.column('Amount', width=125, stretch=NO)
265 table.column('Mode of Payment', width=125, stretch=NO)
266 table.column('Amount', width=125, stretch=NO)
267
268 table.place(rele=0, y=0, relheight=1, relwidth=1)
269
270 list_all_expenses()
271 root.update()
272 root.mainloop()
```



FileEditSelectionViewGoRunTerminalHelp

Pythonworks Expense Tracker

EXPENSE TRACKER

11/24/25

Description

Amount0.0

Payer

Mode of payment:Cash

Add expense

Convert to words before adding

Delete Expense

Clear Fields in DataEntry Frame

Delete All Expenses

View Selected Expense's Details

Edit Selected Expense

Convert Expense to a sentence

S.No.	Date	Payer	Description	Amount	Mode of Payment
1	2025-11-23	SHRADHA SHADMA	BISTRO	300.0	Cash
2	2025-11-22	ROHIT VERMA	MAYURI	250.0	Google pay
3	2025-11-21	SHREYA JAIN	UNDER BELLY	400.0	Paytm

Code is already running!

Ln 272, Col 16Spaces: 4UTF-8CSLFPython3.13.7