



Computer Science Project

Kaizer Futuristic

Development Team:

- Mohammed Daiyaan
- Salman
- Vidyakar



Kaizer Futuristic

Introduction: We are going to create a application Named ultimate database manager which can create modify update and delete database. This app will be useful for those people who don't have any knowledge about database. This app will get the input from the user for the following query and will execute the query.

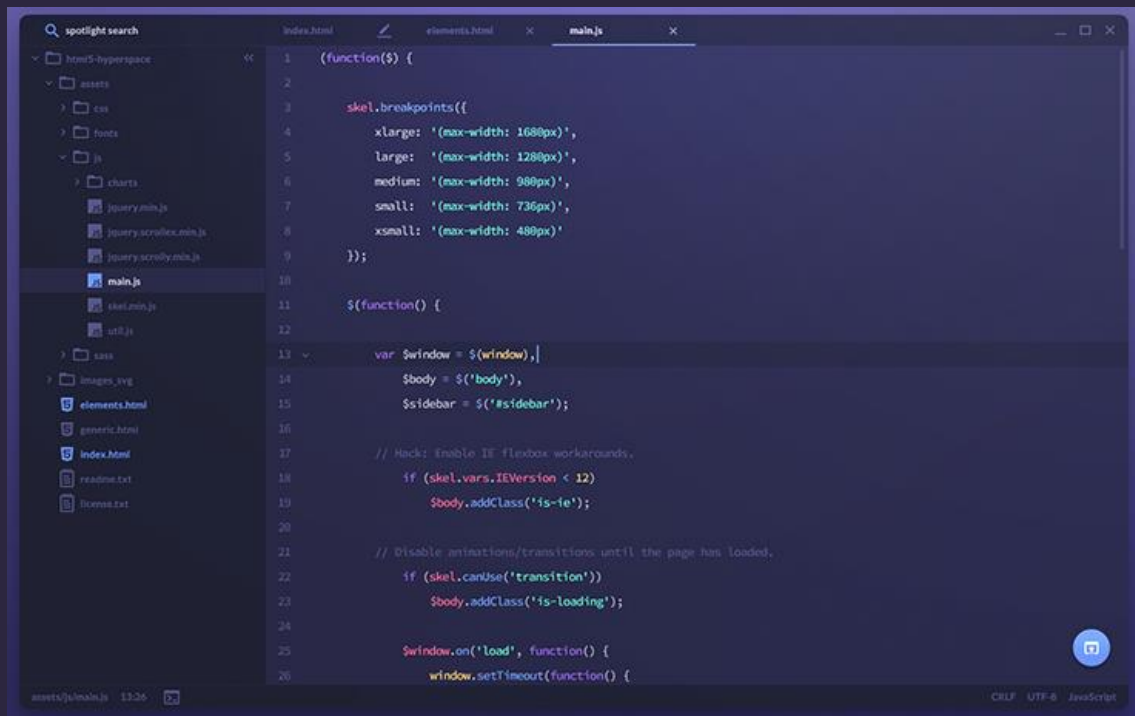


Description: We will use the popular programming language Python And many SQL database for that developing this application Python's thinter module will be used for creating the graphical user interface for

better XP user experience. My SQL connector module for connecting Python And my SQL this is used for creating connecting back end and front end

Development: We will use the python's thinter module for creating the GUI.

- ❖ First we have to import the required module which is thinter and mysql.connector
- ❖ Using thinter we will create the root window. A window in which our application will be displayed.
- ❖ We will design basic buttons, textbox, labels and input box according to our design for the application.
- ❖ By using mysql.connector, we will connect the front and backend
- ❖ We will create options for the user, if they want to create database, update database, alter database or delete database.
- ❖ If they want to create database, we will get the name of the database by using the thinter's input box and the command :
"CREATE DATABASE <input>;"
Will be executed at the backend
- Similarly we will develop for other queries..



Usage: In this modern world where people are creating apps, websites without having the knowledge of programming language by using apps like Figma, WordPress, BlogSpot etc. Similarly, we want to create an application which can create, modify and delete databases without having any prerequisite of programming language.



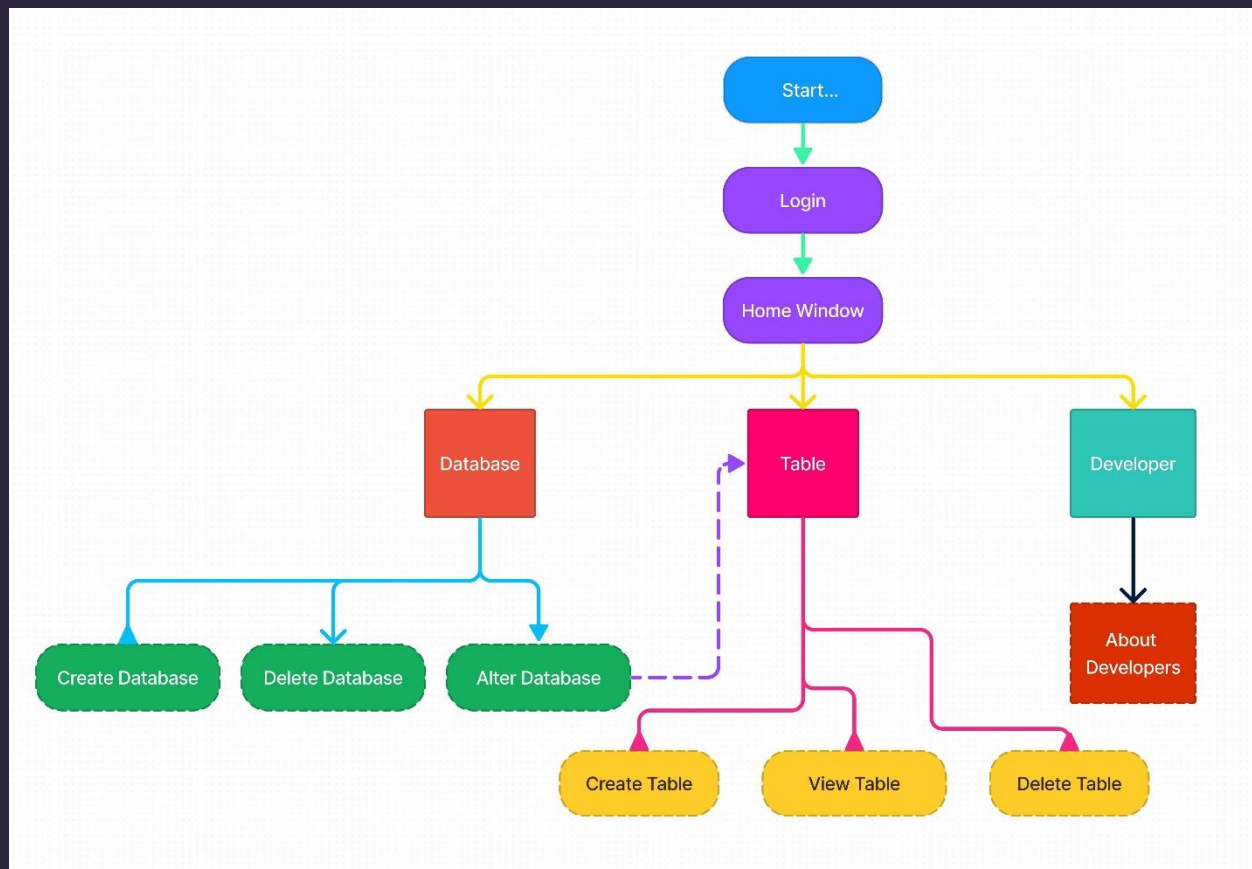
Software requirement:

- Python 3
- MySQL
- Any latest Operating System
- Other plugins

Hardware requirements:

- 1GB Ram
- 100MB Storage
- Any processor

Flow Chart



Source Code:

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py theme.json
KAIZER > app.py > App > alterDataBaseWin
1 import customtkinter as CT
2 from tkinter import *
3 import mysql.connector as m
4 from tkinter import messagebox
5 from PIL import Image, ImageTk
6 import time
7 import os
8
9
10 PATH = os.path.dirname(os.path.realpath(__file__))
11 CT.set_default_color_theme(PATH + "/theme.json")
12 CT.set_appearance_mode("dark")
13 imgBg = Image.open(PATH + "/img/image.jpg")
14 imageBackBtn = Image.open(PATH + "/img/back.PNG").resize((30, 30))
15 appIcon = PATH + "/img/icon.ico"
16
17
18 class App(CT.CTk):
19     def __init__(self):
20         super().__init__()
21         self.ButtonFont = "Aerial"
22         self.navBarFont = "monospace", 20
23         self.title("KAIZER")
24         self.minsize(800, 600)
25         self.bg_image = ImageTk.PhotoImage(imgBg)
26         self.photoBackBtn = ImageTk.PhotoImage(imageBackBtn)
27         self.iconbitmap(appIcon)
28         self.bg = CT.CTkLabel(image=self.bg_image).place(
29             relx=0.5, rely=0.5, anchor=CT.CENTER
30         )
31
32
33 @classmethod
34 def back(self, currentFrm, destinedFrm):
35     currentFrm.destroy()
36
37 def progress(self):
38     self.progressBar.set(0.2)
39     self.update_idletasks()
40     time.sleep(1)
41     self.progressBar.set(0.4)
42     self.update_idletasks()
43     time.sleep(1)
44     self.progressBar.set(0.6)
45     self.update_idletasks()
46     time.sleep(1)
47     self.progressBar.set(0.8)
48     self.update_idletasks()
49     time.sleep(1)
50     self.progressBar.set(1)
51
52 def sliderFun(self):
53     self.sliderFrm = CT.CTkFrame(self)
54     self.progressBar = CT.CTkProgressBar(self.sliderFrm, height=20, width=400)
55     self.progressBar.grid(row=1, column=0, padx=200, pady=(30))
56     self.progressBar.set(0)
57
58 queryLabel = CT.CTkLabel(
59     self.sliderFrm, text="QUERY I N PROCESS", text_font=("Anurati", 28)
60 )
61 queryLabel.grid(row=0, column=0, padx=0, pady=0)
```



```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py x {} theme.json
KAIZER > app.py > App > progress
62 backBtn = CT.CTkButton(
63     self.sliderFrm,
64     text="HOME",
65     height=30,
66     width=30,
67     text_font=("Anurati", 18),
68     command=lambda: self.back(self.sliderFrm, self.homeWindow()),
69     grid(row=3, column=0, columnspan=3, pady=(50, 80))
70 )
71 self.sliderFrm.place(relx=0.5, rely=0.5, anchor=CENTER)
72
73 def deleteTableFun(self):
74     try:
75         DataBase = self.delTableComboobx.get()
76         self.cur.execute(f"use {DataBase};")
77         table = self.deleteTableEntry.get()
78         self.cur.execute(f"drop table {table};")
79         self.deleteTableFrm.destroy()
80         self.sliderFun()
81         self.progress()
82         messagebox.showinfo(
83             "Table Deleted Sucessfully",
84             f"Table {table} has been sucessfully deleted",
85         )
86     except:
87         messagebox.showerror("Error 404", "Please check your input")
88
89 def viewTableFun(self):
90     try:
91         DataBase = self.viewTableComboobx.get()
92         table = self.viewTableEntry.get()
93         self.cur.execute(f"use {DataBase};")
94         self.cur.execute(f"select * from {table};")
95         records = self.cur.fetchall()
96         with open(PATH + "/records.txt", "w") as f:
97             for i in records:
98                 f.write(str(i) + "\n")
99         self.viewTableFrm.destroy()
100         self.sliderFun()
101         self.progress()
102         messagebox.showinfo(
103             "Record Extracted Sucessfully",
104             f"Records from the Table {table} has been sucessfully extracted and stored in {PATH} \\records.txt ",
105         )
106     except:
107         messagebox.showerror("Error 404", "Please check your input")
108
109 def viewTable(self):
110     self.viewTableFrm = CT.CTkFrame(self)
111
112     viewTableHeading = CT.CTkLabel(
113         self.viewTableFrm, text="V I E W T A B L E", text_font=("Anurati", 28)
114     ).grid(row=0, column=0, columnspan=3, pady=(30, 20), padx=(10, 10))
115
116     viewTableSubHeading = CT.CTkLabel(
117         self.viewTableFrm
118     )
```



```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py x theme.json
KAIZER > app.py > App > progress
117 viewTableSubHeading = CT.CTkLabel(
118     self.viewTableFrm,
119     text="Enter the following details to view the records of a Table",
120     text_font=("Arial", 12),
121 ).grid(row=1, column=0, columnspan=3, pady=(0, 30))
122
123 selectDB = CT.CTkLabel(
124     self.viewTableFrm, text="DATABASE", text_font=("anurati", 18)
125 ).grid(row=2, column=0, pady=(20), padx=(60, 0))
126
127 dataBaseList = []
128 self.cur.execute("show databases;")
129 a = self.cur.fetchall()
130
131 for i in a:
132     dataBaseList.append(str(i))
133
134 self.viewTableComboBox = CT.CTkComboBox(
135     self.viewTableFrm, values=dataBaseList, height=30, width=250
136 )
137 self.viewTableComboBox.grid(row=2, column=1, pady=(20), padx=(20, 40))
138 self.viewTableComboBox.set("Select DataBase")
139
140 viewTableLab = CT.CTkLabel(
141     self.viewTableFrm, text="NAME", text_font=("anurati", 18)
142 ).grid(row=3, column=0, pady=10, padx=(20, 20))
143
144 self.viewTableEntry = CT.CTkEntry(
145     self.viewTableFrm, text_font=("Arial", 16), width=250, corner_radius=20
146 )
147
148 self.viewTableEntry.grid(row=3, column=1, pady=10, padx=(30, 60))
149
150 viewTableButton = CT.CTkButton(
151     self.viewTableFrm,
152     text="View Records",
153     command=self.viewTableFun,
154     text_font=("Arial", 16),
155 ).grid(row=4, column=0, columnspan=3, pady=(30, 40))
156
157 backBtn = CT.CTkButton(
158     self.viewTableFrm,
159     text="",
160     fg_color="#171520",
161     height=30,
162     width=30,
163     image=self.photoBackBtn,
164     text_font=("aerial", 18),
165     command=lambda: self.back(self.viewTableFrm, self.tableHomeWin()),
166 ).place(x=0, y=0)
167
168 self.viewTableFrm.place(relex=0.5, rely=0.5, anchor=CENTER)
169
170 def deleteTable(self):
171     self.deleteTableFrm = CT.CTkFrame(self)
172
173     delTableHeading = CT.CTkLabel(
174         self.deleteTableFrm, text="D E L E T E   T A B L E", text_font=("Anurati", 28)
175     ).grid(row=0, column=0, columnspan=3, pady=(30, 20), padx=(10, 10))
176
177     delTableSubHeading = CT.CTkLabel(
```

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py x theme.json
KAIZER > app.py > App > progress
177 delTableSubHeading = CT.CTkLabel(
178     self.deleteTableFrm,
179     text="Enter the following details to delete the Table",
180     text_font=("Arial", 12),
181 ).grid(row=1, column=0, columnspan=3, pady=(0, 30))
182
183 selectDB = CT.CTkLabel(
184     self.deleteTableFrm, text="DATABASE", text_font=("anurati", 18)
185 ).grid(row=2, column=0, pady=(20), padx=(60, 0))
186
187 dataBaselist = []
188 self.cur.execute("show databases;")
189 a = self.cur.fetchall()
190
191 for i in a:
192     dataBaselist.append(str(i))
193
194 self.delTableComboobx = CT.CTkComboBox(
195     self.deleteTableFrm, values=dataBaselist, height=30, width=250
196 )
197 self.delTableComboobx.grid(row=2, column=1, pady=(20), padx=(20, 40))
198 self.delTableComboobx.set("Select DataBase")
199
200 delTableLab = CT.CTkLabel(
201     self.deleteTableFrm, text="NAME", text_font=("anurati", 18)
202 ).grid(row=3, column=0, pady=10, padx=(20, 20))
203
204 self.deleteTableEntry = CT.CTkEntry(
205     self.deleteTableFrm, text_font=("Arial", 16), width=250, corner_radius=20
206 )
207
208 self.deleteTableEntry.grid(row=3, column=1, pady=10, padx=(30, 60))
209
210 delTableButton = CT.CTkButton(
211     self.deleteTableFrm,
212     text="Delete Table",
213     command=self.deleteTableFun,
214     text_font=("Arial", 16),
215 ).grid(row=4, column=0, columnspan=3, pady=(30, 40))
216
217 backBtn = CT.CTkButton(
218     self.deleteTableFrm,
219     text="",
220     fg_color="#171520",
221     height=30,
222     width=30,
223     image=self.photoBackBtn,
224     text_font=("aerial", 18),
225     command=lambda: self.back(self.deleteTableFrm, self.tableHomeWin()),
226 ).place(x=0, y=0)
227
228 self.deleteTableFrm.place(relx=0.5, rely=0.5, anchor=CENTER)
229
230 def tableHomeWin(self):
231     self.navBarDestroy()
232     self.tableHomeFrame = CT.CTkFrame(self)
233
234 createTableBtn = CT.CTkButton(
235     self.tableHomeFrame,
236     text="Create Table",
237     text_font=(self.ButtonFont, 20)
238 )
```

The screenshot displays a Visual Studio Code editor window titled "app.py - Ultimate Database Manager 2 - Visual Studio Code". The editor is open to a file named "app.py", which contains Python code for a database application. The code is organized into several sections, including a main loop, a function to create a database, and a function to delete a database. The code uses the Tkinter library for the GUI and the MySQL connector for database operations. The interface also shows a file explorer on the left with files like "a.py", "a.html", "app.py", and "theme.json". The bottom status bar indicates the current line and column (Ln 39, Col 32) and the Python version (3.9.5 64-bit).

```

236         text="Create Table",
237         text_font=(self.ButtonFont, 20),
238         state="disabled",
239     ).grid(row=0, column=0, columnspan=3, pady=(60, 40), padx=80)
240
241     alterTableBtn = CT.CTkButton(
242         self.tableHomeFrame,
243         text="View Records",
244         text_font=(self.ButtonFont, 20),
245         command=lambda: self.back(self.tableHomeFrame, self.viewTable()),
246     ).grid(row=1, column=0, columnspan=3, pady=40, padx=80)
247
248     deleteTableBtn = CT.CTkButton(
249         self.tableHomeFrame,
250         text="Delete Table",
251         text_font=(self.ButtonFont, 20),
252         command=lambda: self.back(self.tableHomeFrame, self.deleteTable()),
253     ).grid(row=2, column=0, columnspan=3, padx=80, pady=(40, 60))
254
255     backBtn = CT.CTkButton(
256         self.tableHomeFrame,
257         text="",
258         fg_color="#171520",
259         height=30,
260         width=30,
261         image=self.photoBackBtn,
262         text_font=("aerial", 18),
263         command=lambda: self.back(self.tableHomeFrame, self.homeWindow()),
264     ).place(x=0, y=0)
265
266     self.tableHomeFrame.place(relx=0.5, rely=0.5, anchor=CENTER)
267
268     def createDataBaseFun(self):
269         databaseEntry = createDBEntry.get()
270         try:
271             cur = mysql_con.cursor()
272             cur.execute(f"create database {databaseEntry};")
273             self.createDBWin.destroy()
274             self.sliderFun()
275             self.progress()
276             messagebox.showinfo(
277                 "Database Created",
278                 f"Database {databaseEntry} has been successfully created!",
279             )
280         except:
281             messagebox.showerror(
282                 "Invalid Input",
283                 f"Database {databaseEntry} cannot be created. {databaseEntry} database already exists or check your input",
284             )
285
286     def deleteDataBaseFun(self):
287         try:
288             databaseEntry = deleteDBEntry.get()
289             self.cur.execute(f"drop database {databaseEntry};")
290             self.deleteDBFrame.destroy()
291             self.sliderFun()
292             self.progress()
293             messagebox.showinfo(

```



```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py {} theme.json
KAIZER > app.py > App > progress
294     messagebox.showinfo(
295         "Database Deleted",
296         f"Database {databaseEntry} has been sucessfully deleted!😄😄😄",
297     )
298
299     except:
300         messagebox.showerror(
301             "Error 404", f"Error while deleteting database {databaseEntry}!"
302         )
303
304     def createDataBaseWin(self):
305         global createDBEntry
306
307         self.dataBaseOptionFrame.destroy()
308         self.createDBWin = CT.CTkFrame(self)
309
310         create_db_heading_label = CT.CTkLabel(
311             self.createDBWin,
312             text="C R E A T E   D A T A B A S E",
313             text_font=("Anurati", 26),
314             ).grid(row=0, column=0, columnspan=3, pady=(60, 20), padx=(60, 60))
315
316         create_db_subheading_label = CT.CTkLabel(
317             self.createDBWin,
318             text="Enter the following details to create the database",
319             text_font=("aerial", 16),
320             ).grid(row=1, column=0, columnspan=3, pady=(20, 30))
321
322         create_db_database_name = CT.CTkLabel(
323             self.createDBWin, text="NAME" text_font=("Anurati", 22))
```

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py {} theme.json
KAIZER > app.py > App > progress
326     createDBEntry = CT.CTkEntry(
327         self.createDBWin, text_font=("aeial", 20), width=280, corner_radius=20
328     )
329
330     createDBEntry.grid(row=2, column=1, pady=30, padx=(0, 60))
331
332     create_db_button = CT.CTkButton(
333         self.createDBWin,
334         text="Create Database",
335         command=self.createDataBaseFun,
336         text_font=("aerial", 20),
337     ).grid(row=3, column=0, columnspan=3, pady=(30, 60))
338
339     backBtn = CT.CTkButton(
340         self.createDBWin,
341         text="",
342         fg_color="#171520",
343         height=30,
344         width=30,
345         image=self.photoBackBtn,
346         text_font=("aerial", 18),
347         command=lambda: self.back(self.createDBWin, self.dataBaseWindow()),
348     ).place(x=0, y=0)
349     self.createDBWin.place(relx=0.5, rely=0.5, anchor=CENTER)
350
351     def alterDataBaseWin(self):
352         self.dataBaseOptionFrame.destroy()
353         self.alterDBFrame = CT.CTkFrame(self)
354
355         createTableBtn = CT.CTkButton(
```

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py x {} theme.json
KAIZER > app.py > App > progress
355 createTableBtn = CT.CTkButton(
356     self.alterDBFrame,
357     text="Create Table",
358     text_font=(self.ButtonFont, 20),
359     state="disabled",
360 ).grid(row=0, column=0, columnspan=3, pady=(60, 40), padx=80)
361
362 alterTableBtn = CT.CTkButton(
363     self.alterDBFrame,
364     text="View Table",
365     text_font=(self.ButtonFont, 20),
366     command=lambda: self.back(self.alterDBFrame, self.viewTable()),
367 ).grid(row=1, column=0, columnspan=3, pady=40, padx=80)
368
369 deleteTableBtn = CT.CTkButton(
370     self.alterDBFrame,
371     text="Delete Table",
372     text_font=(self.ButtonFont, 20),
373     command=lambda: self.back(self.alterDBFrame, self.deleteTable()),
374 ).grid(row=2, column=0, columnspan=3, padx=80, pady=(40, 60))
375
376 backBtn = CT.CTkButton(
377     self.alterDBFrame,
378     text="",
379     fg_color="#171520",
380     height=30,
381     width=30,
382     image=self.photoBackBtn,
383     text_font=("aerial", 18),
384     command=lambda: self.back(self.alterDBFrame, self.dataBaseWindow())
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
```

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py x {} theme.json
KAIZER > app.py > App > progress
385 ).place(x=0, y=0)
386
387 self.alterDBFrame.place(relx=0.5, rely=0.5, anchor=CENTER)
388
389 def deleteDataBaseWin(self):
390     global deleteDBEntry
391
392     self.dataBaseOptionFrame.destroy()
393     self.deleteDBFrame = CT.CTkFrame(self)
394
395     delete_db_heading_label = CT.CTkLabel(
396         self.deleteDBFrame,
397         text="DELETED DATABASE",
398         text_font=("Anurati", 26),
399     ).grid(row=0, column=0, columnspan=3, pady=(60, 20), padx=(60, 60))
400
401     delete_db_subheading_label = CT.CTkLabel(
402         self.deleteDBFrame,
403         text="Enter the name of the Database to be deleted",
404         text_font=("aerial", 16),
405     ).grid(row=1, column=0, columnspan=3, padx=(10, 10), pady=(20, 30))
406
407     delete_db_name = CT.CTkLabel(
408         self.deleteDBFrame, text="NAME", text_font=("Anurati", 22)
409     ).grid(row=2, column=0, pady=30, padx=(20, 0))
410
411     deleteDBEntry = CT.CTkEntry(
412         self.deleteDBFrame, text_font=("aerial", 20), width=290, corner_radius=20
413     )
414     deleteDBEntry.grid(row=3, column=0, width=290, height=30, padx=(0, 40))
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
```

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py theme.json
KAIZER > app.py > App > progress
416 delete_db_button = CT.CTkButton(
417     self.deleteDBFrame,
418     text="Delete Database",
419     text_font=("aerial", 20),
420     corner_radius=15,
421     command=self.deleteDataBaseFun,
422 ) .grid(row=3, column=0, columnspan=3, pady=(30, 60))
423
424 backBtn = CT.CTkButton(
425     self.deleteDBFrame,
426     text="",
427     fg_color="#171520",
428     height=30,
429     width=30,
430     image=self.photoBackBtn,
431     command=lambda: self.back(self.deleteDBFrame, self.dataBaseWindow()),
432 ) .place(x=0, y=0)
433 self.deleteDBFrame.place(relx=0.5, rely=0.5, anchor=CENTER)
434
435 def dataBaseWindow(self):
436     self.dataBaseOptionFrame = CT.CTkFrame(self)
437     self.navBarDestroy()
438
439     createDataBaseBtn = CT.CTkButton(
440         self.dataBaseOptionFrame,
441         text="Create Database",
442         text_font=(self.ButtonFont, 20),
443         command=self.createDataBaseWin,
444     ) .grid(row=0, column=0, columnspan=3, padx=(60, 40), pady=80)
445
446 alterDataBaseBtn = CT.CTkButton(
447     self.dataBaseOptionFrame,
448     text="Alter Database",
449     text_font=(self.ButtonFont, 20),
450     command=self.alterDataBaseWin,
451 ) .grid(row=1, column=0, columnspan=3, padx=40, pady=60)
452
453 deleteDataBaseBtn = CT.CTkButton(
454     self.dataBaseOptionFrame,
455     text="Delete Database",
456     text_font=(self.ButtonFont, 20),
457     command=self.deleteDataBaseWin,
458 ) .grid(row=2, column=0, columnspan=3, padx=80, pady=(40, 60))
459
460 backBtn = CT.CTkButton(
461     self.dataBaseOptionFrame,
462     text="",
463     fg_color="#171520",
464     height=30,
465     width=30,
466     image=self.photoBackBtn,
467     text_font=("aerial", 16),
468     command=lambda: self.back(self.dataBaseOptionFrame, self.homeWindow()),
469 ) .place(x=0, y=0)
470
471 self.dataBaseOptionFrame.place(relx=0.5, rely=0.5, anchor=CENTER)
472
473 def aboutDeveloper(self):
474     self.navBarDestroy()
475     self.aboutDeveloperWin = CT.CTkFrame(self)
Ln 39, Col 32 Spaces: 4 UTF-8 CRLF Python 3.9.5 64-bit Go Live Prettier
```



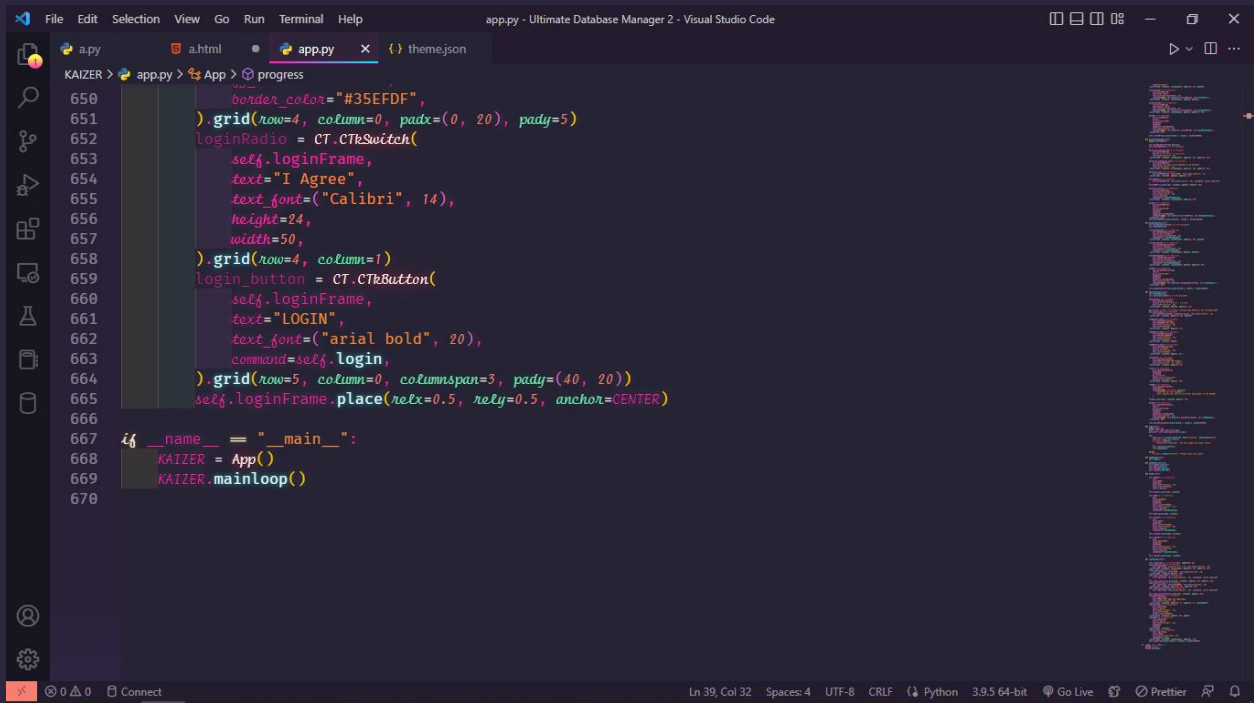
```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py theme.json
KAIZER > app.py > App > progress
477 headingLabel = CT.CTkLabel(
478     self.aboutDeveloperWin,
479     text="D E V E L O P M E N T   T E A M",
480     text_font=("anurati", 28),
481 ).grid(row=0, column=0, padx=40, pady=(50, 15))
482
483 description = "This is a Project created using Python by the following DEVS"
484 descriptionLabel = CT.CTkLabel(
485     self.aboutDeveloperWin, text=description, text_font=("aerial", 14)
486 ).grid(row=1, column=0, pady=(10, 30), padx=20)
487
488 teamMembersLabel1 = CT.CTkLabel(
489     self.aboutDeveloperWin,
490     text="MOHAMMED DAI YAAN",
491     text_font=("anurati", 20),
492     text_color="#ef233c",
493 ).grid(row=2, column=0, pady=(10, 5))
494
495 teamMembersLabel2 = CT.CTkLabel(
496     self.aboutDeveloperWin,
497     text="SALMAN GURSHID",
498     text_font=("anurati", 20),
499     text_color="#ef233c",
500 ).grid(row=3, column=0, pady=5)
501
502 teamMembersLabel3 = CT.CTkLabel(
503     self.aboutDeveloperWin,
504     text="V I DYAKAR",
505     text_font=("anurati", 20),
506     text_color="#ef233c"
507 ).grid(row=4, column=0, pady=(5, 10))
508
509 rateUsLabel = CT.CTkLabel(
510     self.aboutDeveloperWin,
511     text="RATE US USING THE SLIDER!",
512     text_font=("aerial", 16, "bold"),
513 ).grid(row=5, column=0, pady=(25, 0))
514
515 slider = CT.CTkSlider(
516     self.aboutDeveloperWin,
517     height=30,
518     width=400,
519     fg_color="red",
520     button_hover_color="pink",
521     progress_color="gold",
522 ).grid(row=6, column=0, pady=(5, 10))
523
524 rateBtn = CT.CTkButton(
525     self.aboutDeveloperWin,
526     text="Rate!",
527     command=lambda: messagebox.showinfo(
528         "Thank You For Your Feedback",
529         "Your response will help us in further development of the KAIZER",
530     ),
531 ).grid(row=7, column=0, pady=(0, 40))
532
533 backBtn = CT.CTkButton(
534     self.aboutDeveloperWin
```

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py theme.json
KAIZER > app.py > App > progress
536 text= "",
537 fg_color="#171520",
538 height=30,
539 width=30,
540 image=self.photoBackBtn,
541 text_font=("aerial", 18),
542 command=lambda: self.back(self.aboutDeveloperWin, self.homeWindow()),
543 ).place(x=0, y=0)
544
545 self.aboutDeveloperWin.place(relx=0.5, rely=0.5, anchor=CENTER)
546
547 def login(self):
548     global mysql_con
549     name = self.login_name_entry.get()
550     password = self.login_passwd_entry.get()
551
552     try:
553         mysql_con = m.connect(user=name, host="localhost", password=password)
554         self.cur = mysql_con.cursor()
555         messagebox.showinfo(
556             "Sucessfully connected", "You have logged into mysql server"
557         )
558         self.loginFrame.destroy()
559         self.homeWindow()
560
561     except:
562         messagebox.showerror("Error", "Please check your input")
563
564 def homeWindow(self):
565     self.navBar()
566
567 def navBarDestroy(self):
568     self.homeBtn.destroy()
569     self.dbBtn.destroy()
570     self.tableBtn.destroy()
571     self.loginBtn.destroy()
572
573 def navBar(self):
574
575     self.homeBtn = CT.CTkButton(
576         self,
577         text="HOME",
578         height=60,
579         text_font=("anurati", 16),
580         hover_color="#41lead4",
581         corner_radius=0,
582     )
583     self.homeBtn.grid(row=0, column=0)
584
585     self.dbBtn = CT.CTkButton(
586         self,
587         text="DATABASE",
588         height=60,
589         width=180,
590         hover_color="#ff006e",
591         text_font=("anurati", 16),
592         corner_radius=0,
593         command=self.dataBaseWindow,
594     )
595     self.dbBtn.grid(row=0, column=1)
```

Ln 39, Col 32 Spaces: 4 UTF-8 CRLF Python 3.9.5 64-bit Go Live Prettier

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py theme.json
KAIZER > app.py > App > progress
597 self.tableBtn = CT.CTkButton(
598     self,
599     text="TABLE",
600     height=60,
601     hover_color="#ffd60a",
602     text_font=("anurati", 16),
603     corner_radius=0,
604     command=self.tableHomeWin,
605 )
606 self.tableBtn.grid(row=0, column=2)
607
608 self.loginBtn = CT.CTkButton(
609     self,
610     text="DEVELOPERS",
611     height=60,
612     width=180,
613     text_font=("anurati", 16),
614     hover_color="#ef233c",
615     corner_radius=0,
616     command=self.aboutDeveloper,
617 )
618 self.loginBtn.grid(row=0, column=3)
619
620 def loginWindow(self):
621
622     self.loginFrame = CT.CTkFrame(self, image=self.bg)
623     login_heading_label = CT.CTkLabel(
624         self.loginFrame, text="SIGN IN", text_font=("Anurati", 28)
625     ).grid(row=0, column=0, columnspan=3, pady=(30, 40), padx=(10, 10))
626     login_name_label = CT.CTkLabel(
```

```
File Edit Selection View Go Run Terminal Help app.py - Ultimate Database Manager 2 - Visual Studio Code
a.py a.html app.py theme.json
KAIZER > app.py > App > progress
626 login_name_label = CT.CTkLabel(
627     self.loginFrame, text="NAME", text_font=("Anurati", 20)
628 ).grid(row=1, column=0, pady=(0, 30))
629 self.login_name_entry = CT.CTkEntry(
630     self.loginFrame, text_font=("Anurati", 20), width=250, corner_radius=20
631 )
632 self.login_name_entry.grid(row=1, column=1, pady=(0, 30), padx=(0, 30))
633 login_passwr_label = CT.CTkLabel(
634     self.loginFrame, text="PASSWORD", text_font=("Anurati", 20)
635 ).grid(row=2, column=0, padx=(30, 20), pady=(20, 20))
636 self.login_passwr_entry = CT.CTkEntry(
637     self.loginFrame, text_font=("Anurati", 20), width=250, corner_radius=20
638 )
639 self.login_passwr_entry.grid(row=2, column=1, padx=(0, 30))
640 loginCheckBoxLabel = CT.CTkLabel(
641     self.loginFrame,
642     text="AGREE OUR TERMS AND CONDITION",
643     text_font=("Calibri", 12),
644 ).grid(row=3, column=0, padx=(20, 0), pady=(30, 5), columnspan=3)
645 loginCheckBox = CT.CTkCheckBox(
646     self.loginFrame,
647     text="I Agree",
648     text_font=("Calibri", 14),
649     fg_color="#e71d36",
650     border_color="#35EFDF",
651 ).grid(row=4, column=0, padx=(0, 20), pady=5)
652 loginRadio = CT.CTkSwitch(
653     self.loginFrame,
654     text="I Agree",
655     text_font=("Calibri", 14)
```



```
File Edit Selection View Go Run Terminal Help
app.py - Ultimate Database Manager 2 - Visual Studio Code

a.py a.html app.py x {} theme.json
KAIZER > app.py > App > progress

650         border_color="#35EFDF",
651     ).grid(row=4, column=0, padx=(0, 20), pady=5)
652     loginRadio = CT.CTkSwitch(
653         self.loginFrame,
654         text="I Agree",
655         text_font=("Calibri", 14),
656         height=24,
657         width=50,
658     ).grid(row=4, column=1)
659     login_button = CT.CTkButton(
660         self.loginFrame,
661         text="LOGIN",
662         text_font=("arial bold", 20),
663         command=self.login,
664     ).grid(row=5, column=0, columnspan=3, pady=(40, 20))
665     self.loginFrame.place(relx=0.5, rely=0.5, anchor=CENTER)
666
667 if __name__ == "__main__":
668     KAIZER = App()
669     KAIZER.mainloop()
670
```

Ln 39, Col 32 Spaces: 4 UTF-8 CRLF Python 3.9.5 64-bit Go Live Prettier

Connect

Thank You