To-Do list

Razan Abdulrahman Alrashed
Rahaf Ali Alduaij
Sara Khalid Alshuwaier
Nada Abdalrahman Aldayel

Introduction

A to-do list is a systematic and organized way to document and manage tasks or activities that need to be completed. It serves as a reminder and helps individuals prioritize their work, ensuring that important tasks are addressed efficiently. The digital to-do list is considered a flexible and efficient means to stay organized, affirming that tasks can be easily accessed and updated in the fast-paced, ever-changing world of technology.

Benefits of a to-do list:

- It helps to get the task done on time.
- Keeps you organized.
- Increases your productivity.
- Relives stress anxiety and tension.
- Helps you stay responsible .
- Helps you practice good time management.
- Serves as a memory aid.
- Gives you a sense of accomplishment.

Python code

Importing the required modules

Motivation.txt

```
import tkinter as tk              # importing the tkinter module as tk
from tkinter import ttk           # importing the ttk module
from tkinter import messagebox    # importing the messagebox module
import sqlite3 as sql             # importing the sqlite3 module as sql
```

Adding a Task to the list

Motivation.txt

```
tasks = []     # defining an empty list
# defining the function to add tasks to the list
def add_task():
# getting the string from the entry field
    task_string = task_field.get()
# checking whether the string is empty or not
    if len(task_string) == 0:
# displaying a message box with 'Empty Field' message
        messagebox.showinfo('Error', 'Field is Empty.')
    else:
        tasks.append(task_string)   # adding the string to the tasks list
# using the execute() method to execute a SQL statement
        the_cursor.execute('insert into tasks values (?)', (task_string ,))
        list_update()   # calling the function to update the list
        task_field.delete(0, 'end')  # deleting the entry in the entry field
```

Deleting a task from the list

Motivation.txt

```
# defining the function to delete a task from the list
def delete_task():
   # using the try-except method
    try:
   # getting the selected entry from the list box
       the_value = task_listbox.get(task_listbox.curselection())
       # checking if the stored value is present in the tasks list
       if the_value in tasks:
           # removing the task from the list
           tasks.remove(the_value)
           # calling the function to update the list
           list_update()
           # using the execute() method to execute a SQL statement
           the_cursor.execute('delete from tasks where title = ?', (the_value,))
    except:
      # displaying the message box with 'No Item Selected' message for an exception
       messagebox.showinfo('Error', 'No Task Selected. Cannot Delete.')
```

Deleting all the entries from the list

Motivation.txt

```
#function to delete all tasks from the list
def delete_all_tasks():
# displaying a message box to ask user for confirmation
    message_box = messagebox.askyesno('Delete All', 'Are you sure?')
    # if the value turns to be True
    if message_box == True:
    # using while loop to iterate through the tasks list until it's empty
        while(len(tasks) != 0):
            # using the pop() method to pop out the elements from the list
            tasks.pop()
        # using the execute() method to execute a SQL statement
        the_cursor.execute('delete from tasks')
         # calling the function to update the list
        list_update()
```

Clearing the list

Motivation.txt

```
# function to clear the list
def clear_list():
    # using the delete method to delete all entries from the list box
    task_listbox.delete(0, 'end')
```

Closing the application

Motivation.txt

```
# function to close the application
def close():
    # printing the elements from the tasks list
    print(tasks)
     # using the destroy() method to close the application
    guiWindow.destroy()
```

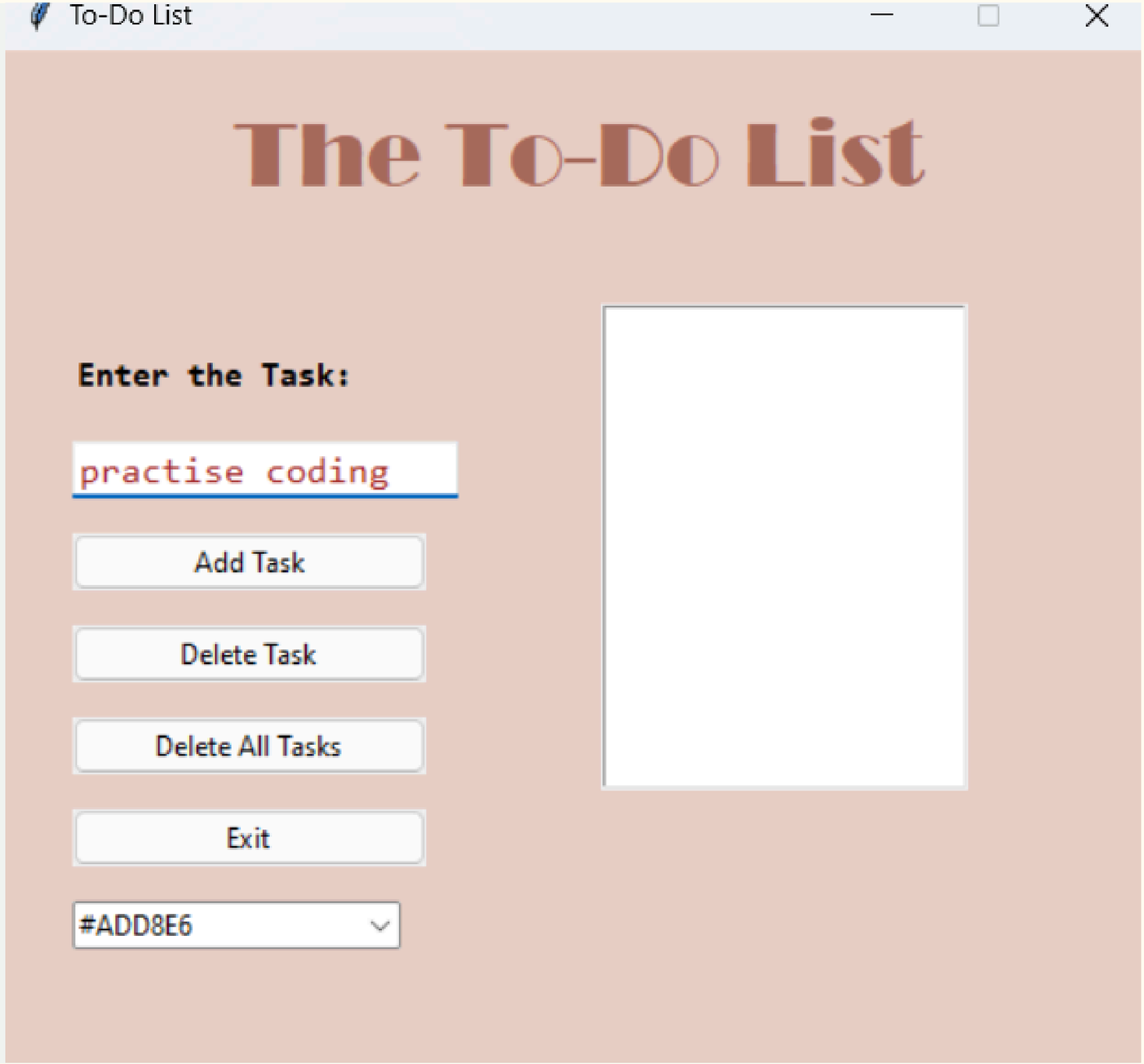
Change a background color

Motivation.txt

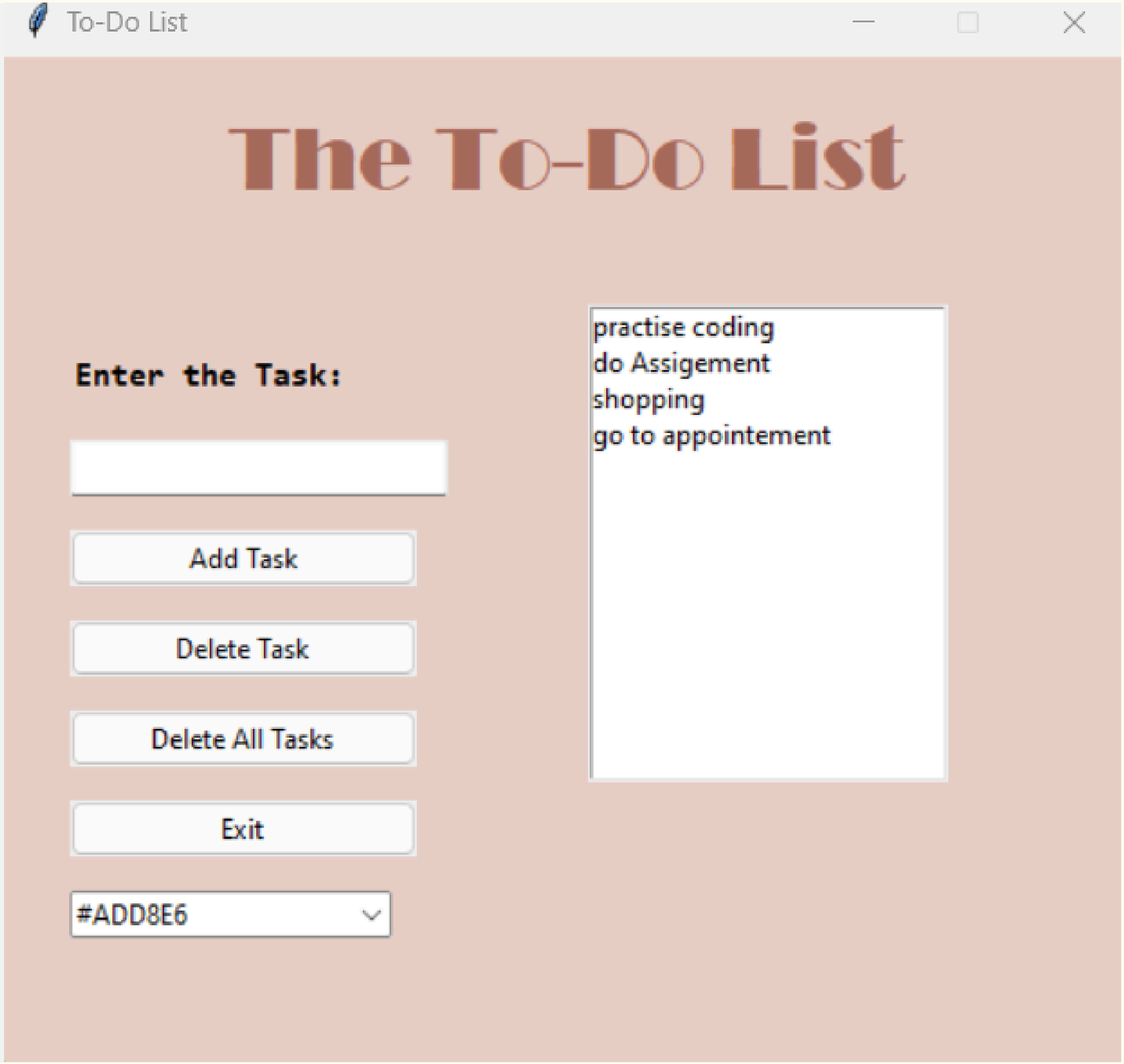
```
# function to change background color
def change_background_color(color):
# set the main window background color
  guiWindow.configure(bg=color)
# set the header frame background color
  header_frame.configure(bg=color)
# set the Function frame background color
  functions_frame.configure(bg=color)
# set the list box background color
  listbox_frame.configure(bg=color)
```

Instructions on how to use program?

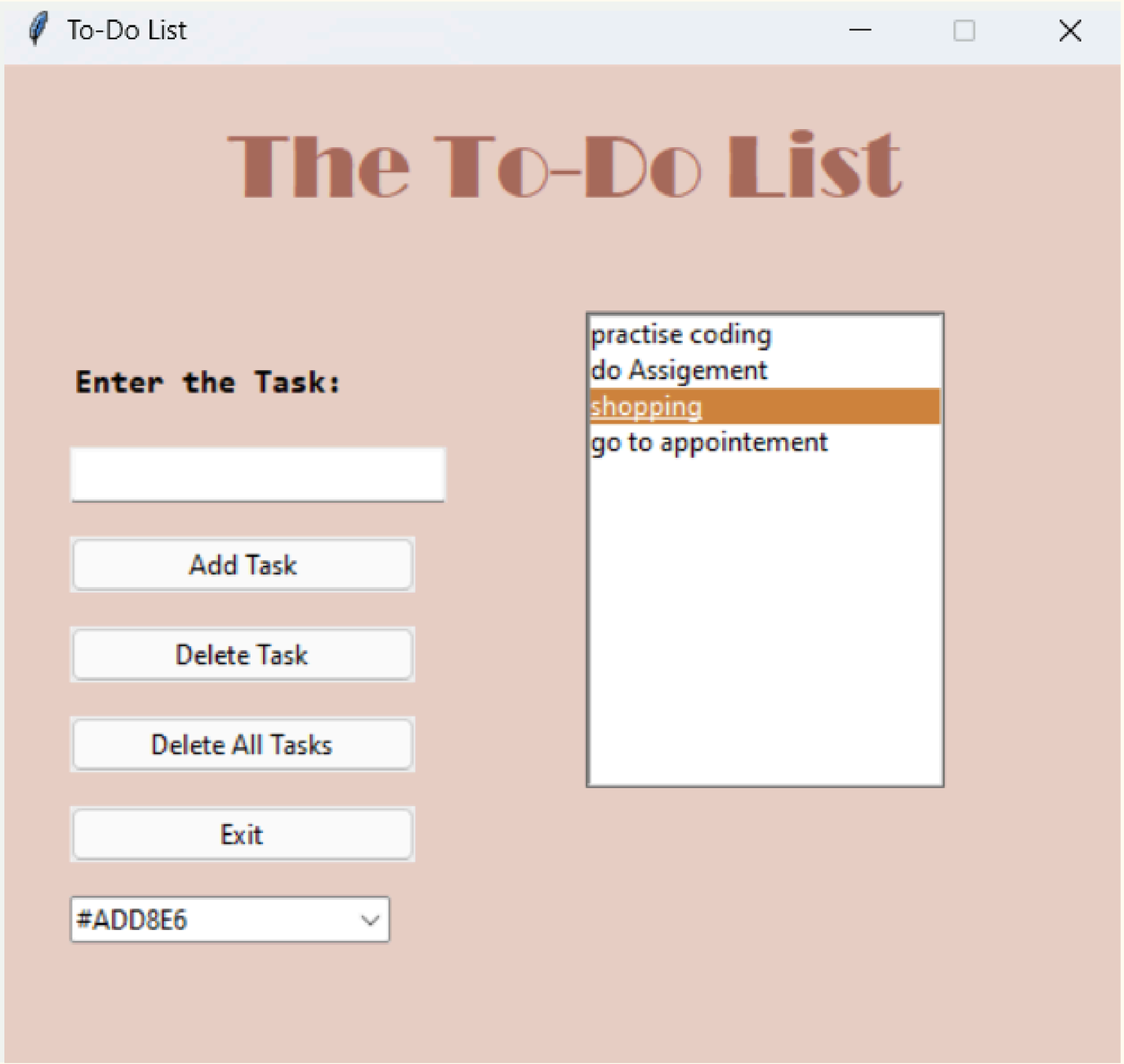
First when we want to Add New Task we click on the white Blank then write your Task then click on the Add Task Button



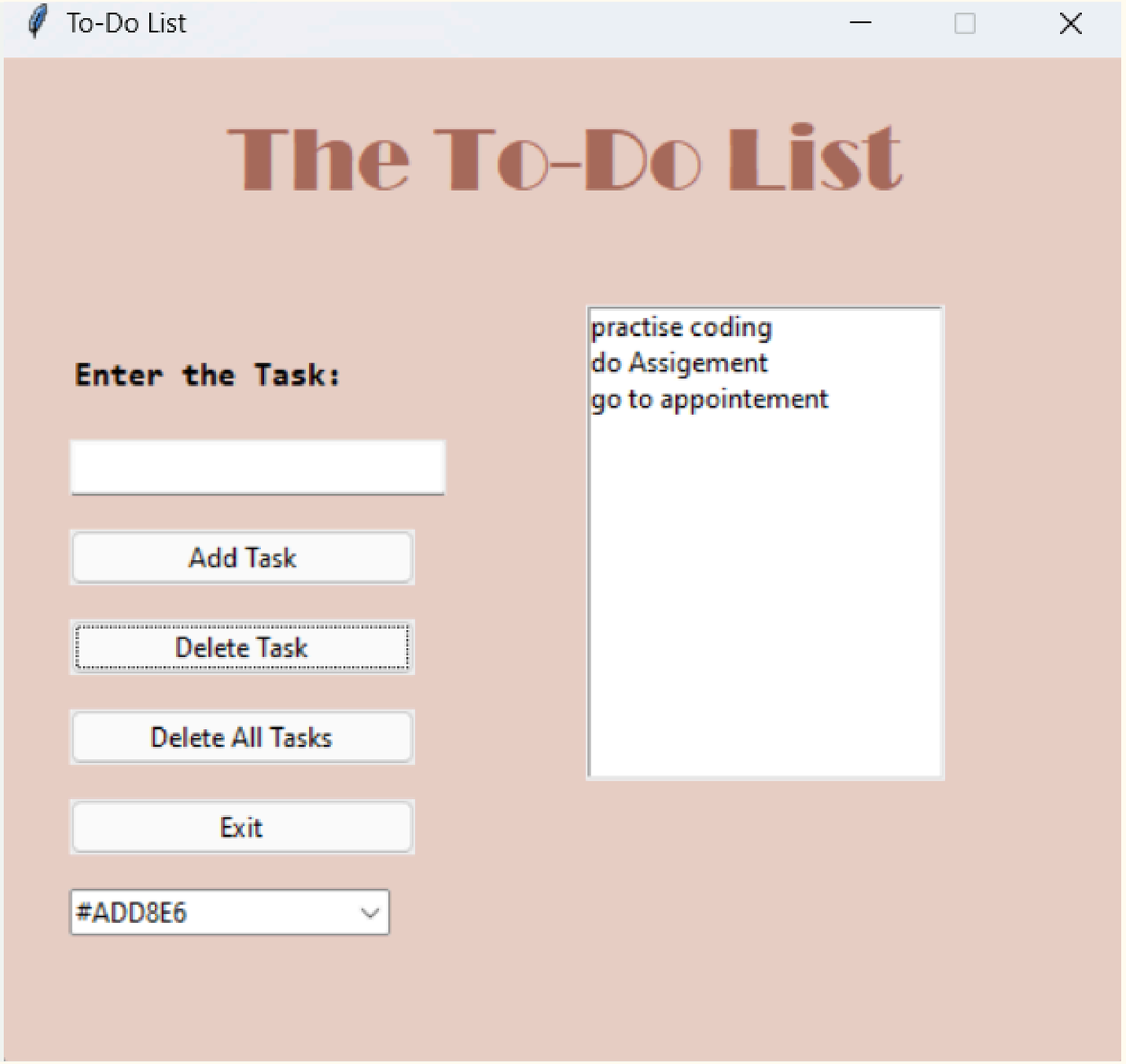
Write all your Tasks on The Same Way



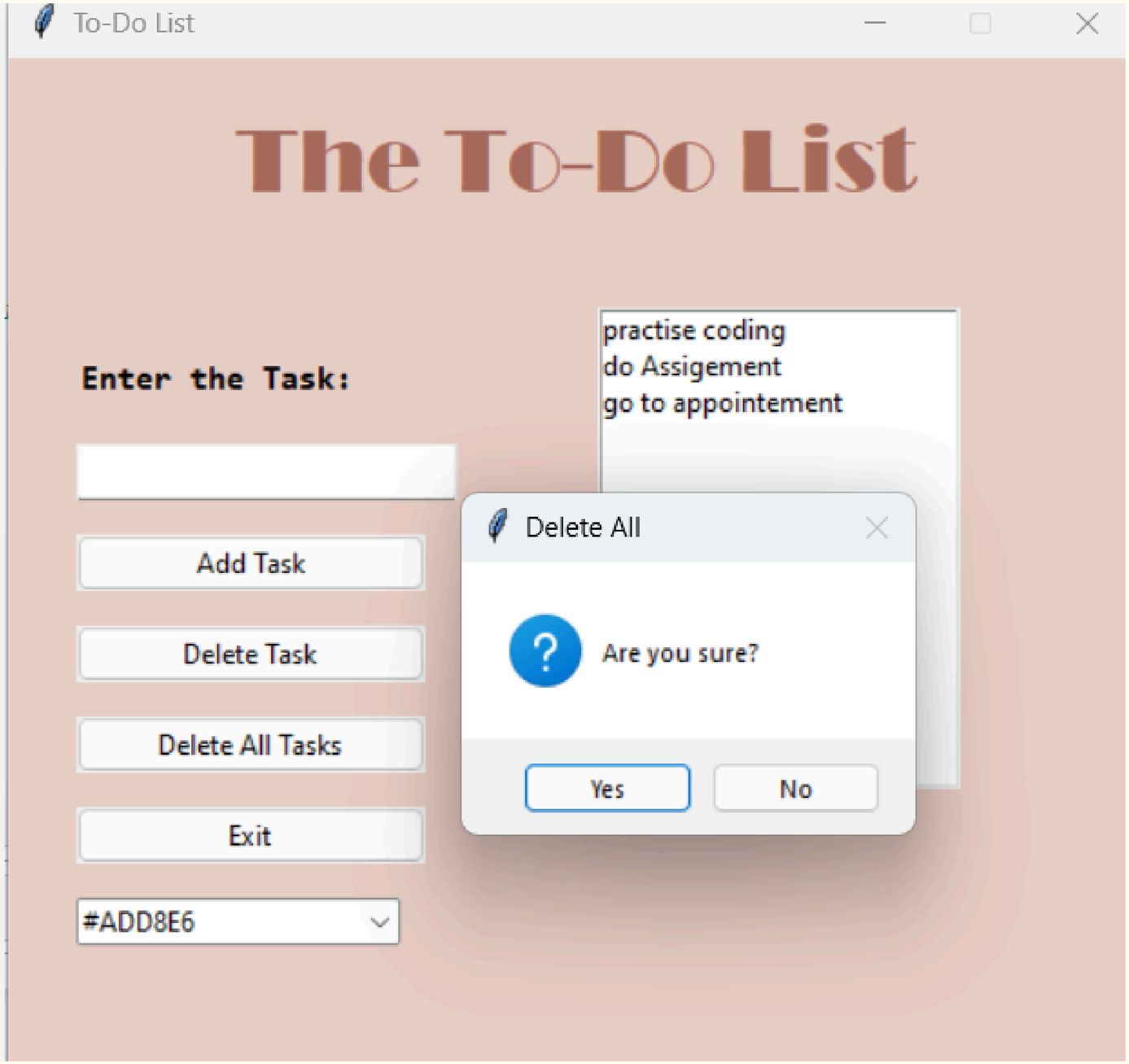
When we want to delete a task, we select the task we want to delete, then click the Delete Task button



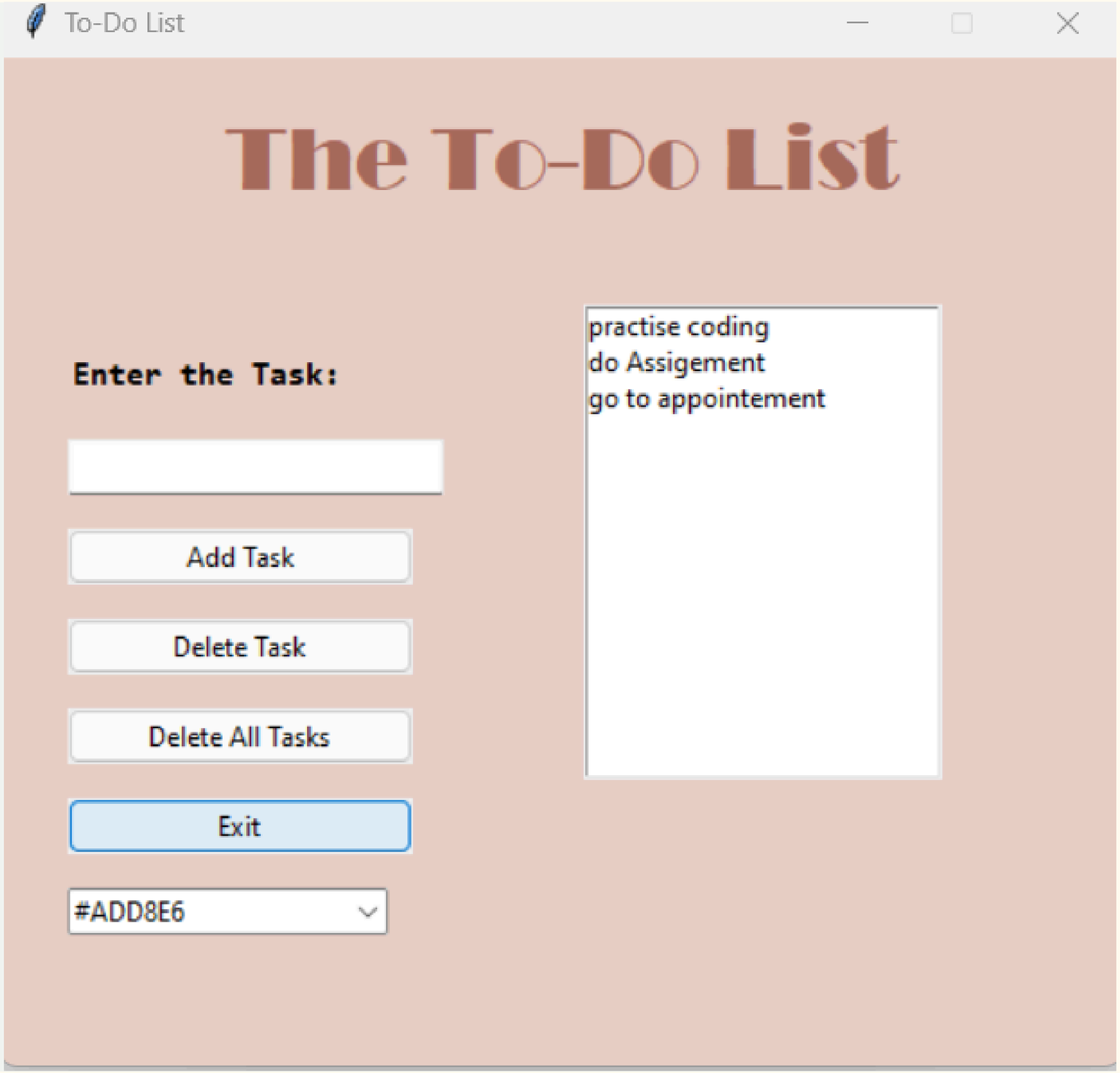
Deleted!


To-Do List
The To-Do List
Enter the Task:
practise coding
do Assigement
go to appointement
Add Task
Delete Task
Delete All Tasks
Exit
#ADD8E6

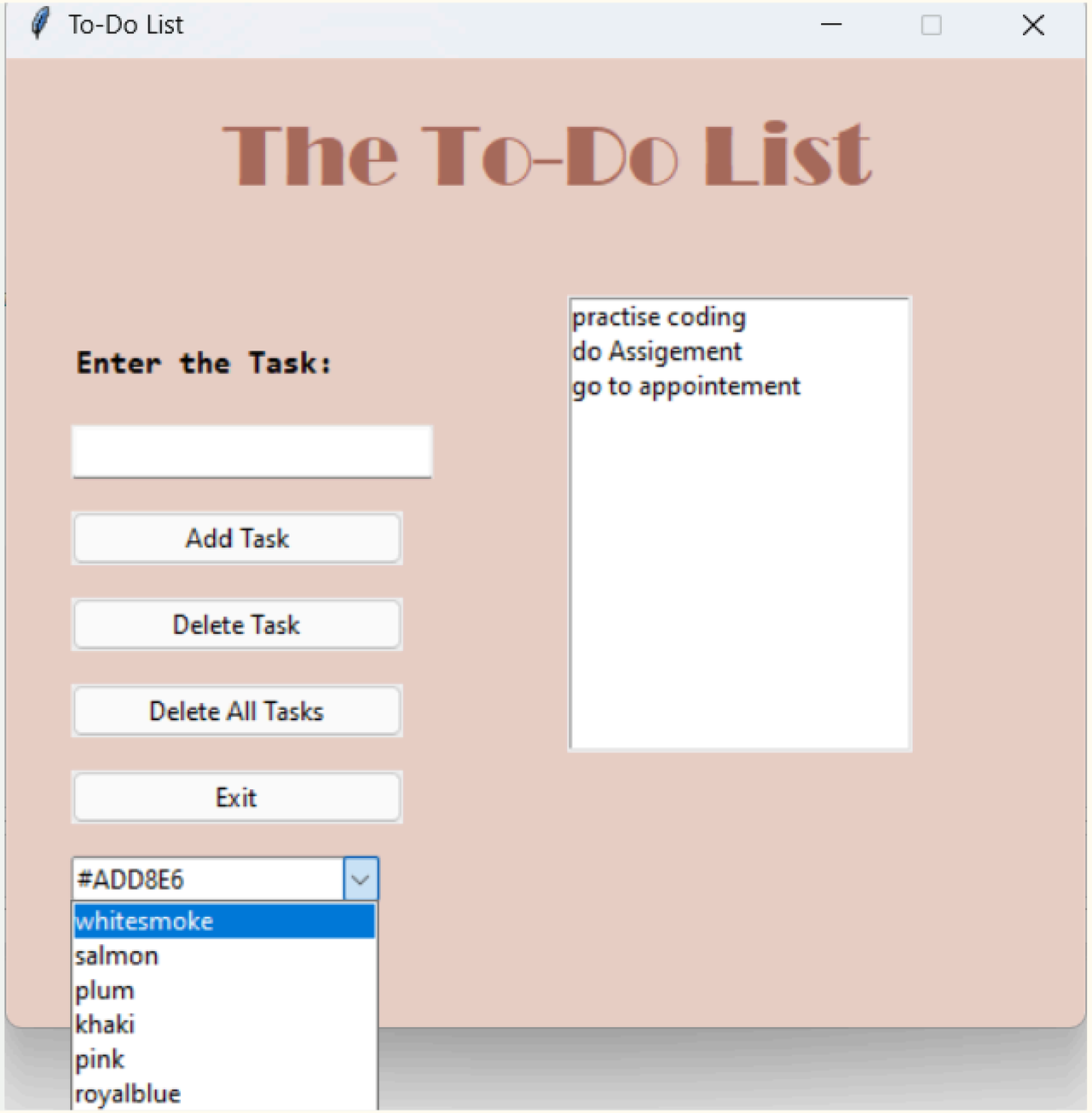
When you want to delete all tasks, click on the Delete All Tasks button After that, a window will appear with a message confirming the deletion. If the user presses Yes, all tasks will be deleted. If he presses No, the deletion will be undone.



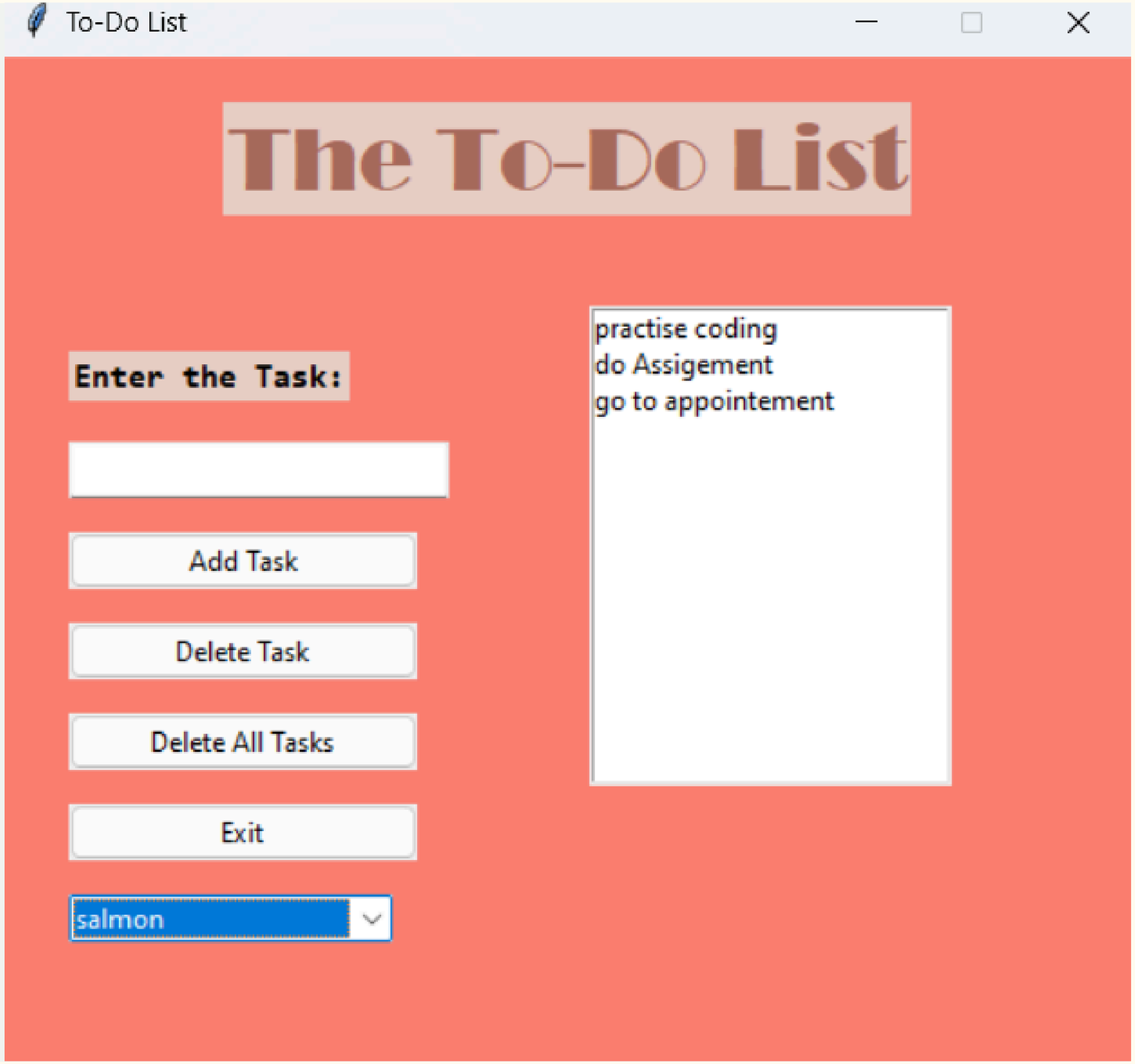
If the user presses the exit button, the application will exit



Also adding a feature to the program is that the user can change the background color



changed


To-Do List
The To-Do List
practise coding
do Assigement
go to appointement
Enter the Task:
Add Task
Delete Task
Delete All Tasks
Exit
salmon

Summarize of project

The **To-Do list** is a list of tasks that must be complete.

It is one of the simplest solutions for managing the tasks that a person wants to accomplish.

When you write the tasks what you need to do , you are more motivated.

In this sense, we designed a simple website for people to track their tasks using the **Python language** and help topics the

Interface, database and exceptions.

Finally, we tried to create a simple and effective to-do list that eliminates distractions to help people get things done easily and avoid hassles.

Suggest additions on project

1- Made it like application

2- Allowing the user to arrange tasks according to priority and give each task a specific time to complete

3- Add a button for each task to mark the completion of the task

4- Allow the user to add his number or email in Application to send reminder notifications to perform tasks in specific time

limitations on project

1-Lack of integration with other tools or platforms, user may want to want if their to-do list can sync with calendars or email.

2- No mobile support our to-do list is not suitable to run in mobile devices.

3- Limited customization, user may prefer to be able to prioritize tasks.

4- To-Do list doesn’t send notifications to remind you of tasks.

Thanks for Listening !

Any questions?