|  |  |
| --- | --- |
| A picture of a winding road and trees  [Document title]  [Document subtitle] | Abstract  [Draw your reader in with an engaging abstract. It is typically a short summary of the document. When you’re ready to add your content, just click here and start typing.]  Shahid Abid  [Course title] |

Here's a simple To-Do list app in core Python:

# (link unavailable)

class Todo:

def \_\_in it\_\_ (self, title, description):

self. Title = title

self. Description = description

self. Completed = False

def toggle completed(self):

self. Completed = not self. Completed

def \_\_str\_\_(self):

status = "Completed" if self. Completed else "Not Completed"

return f"{self. Title} - {self. Description} ({status})"

class Todo List:

def \_\_in it\_\_(self):

self. Odos = []

def add Todo (self, title, description):

Todo ((Todo (title, description))

def delete Todo (self, index):

try:

del self. Odos[index]

except Index Error:

print ("Invalid index.")

def print ((self):

for I, Todo in enumerate (self. Odos):

print(f"{i+1}. {Todo}")

def toggle completed (self, index):

try:

self. Odos[index]. toggle completed ()

except Index Error:

print ("Invalid index.")

def main ():

Todo list = Todo List ()

while True:

print ("\n1. Add Todo")

print ("2. Delete Todo")

print ("3. Toggle Completed")

print ("4. Display Todos")

print ("5. Quit")

choice = input ("Choose an option: ")

if choice == "1":

title = input ("Enter title: ")

description = input ("Enter description: ")

input ((title, description)

elf choice == "2":

index = int (input ("Enter index to delete: ")) - 1

input ((index)

elf choice == "3":

index = int (input ("Enter index to toggle completed: ")) - 1

input ((index)

elf choice == "4":

elf ()

elf choice == "5":

break

else:

print ("Invalid option. Please choose again.")

if \_\_name\_\_ == "\_\_main\_\_":

main ()

``"

Run the script using `python (link unavailable) `.

This app allows you to:

1. Add to-dos with title and description

2. Delete to-dos by index

3. Toggle completed status of to-dos by index

4. Display all to-dos

5. Quit the app

**Note:**

* This implementation uses a simple text-based interface.
* It stores to-dos in memory, so data will be lost when the app closes.
* You can enhance this app by adding features like data persistence (e.g., using a database or file storage), user authentication, and more.
* To make the app more robust, consider adding:
* Error handling for invalid inputs
* Validation for Todo titles and descriptions
* Sorting and filtering options for to-dos
* Due dates and reminders for to-dos
* User authentication and authorization