A Project Report

on

TO DO List Using Python

Submitted in partial fulfillment of the requirement for the award of the degree of

Program name: B. Tech CSE

Under The Guidance of

Dr.Vimal Kumar

Submitted By

Katyayani Singh 21SCSE1011519

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING DEPARTMENT OFCOMPUTER SCIENCE AND ENGINEERING GALGOTIAS UNIVERSITY, GREATER NOIDA

Introduction

Title: Design and Implementation of a Python based To-Do List Project

In today's fast-paced world, effective task management is crucial for personal and professional success. This report delves into the design and implementation of a Python based ToDo List project, focusing on features such as adding tasks, displaying tasks, marking tasks as done, and providing a seamless user experience. The project aims to offer a simple yet powerful tool for individuals seeking an organized approach to their daily responsibilities

Project Overview

The To-Do List project is built using Python, leveraging its versatility and ease of use. The core features include:

Add Task: Users can add tasks to their to-do list with a title, description, and due date.

Show Tasks: The system allows users to view their existing tasks, providing a comprehensive overview of their to-do list.

Mark Task as Done: Once a task is completed, users have the option to mark it as done, providing a sense of accomplishment.

Exit: The project offers a clean exit mechanism, ensuring a user-friendly experience.

Implementation Details

The To-Do List project employs object-oriented programming (OOP) principles for a modular and organized code structure. The user interface is developed using the Python input() function for simplicity. Task data is stored using data structures such as lists or dictionaries, allowing efficient task management.

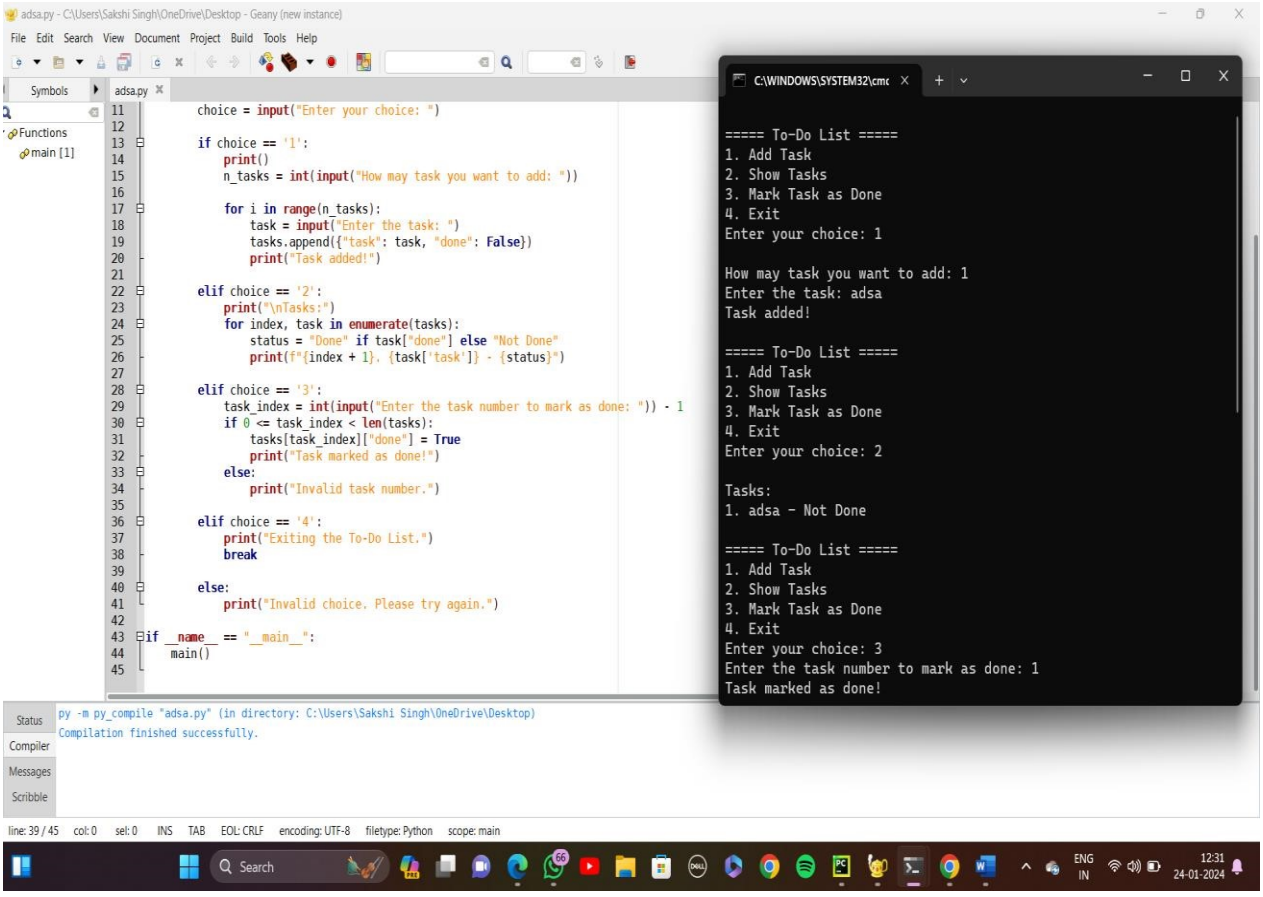
The "Add Task" feature prompts the user to input task details, creating a new task object that is added to the task list. The "Show Tasks" feature iterates through the task list, displaying each task's details. The "Mark Task as Done" functionality updates the task status, and the "Exit" option gracefully terminates the program.

User Experience

The To-Do List project prioritizes a positive user experience. Input validation ensures that users provide relevant information when adding tasks. The project utilizes clear prompts and messages to guide users through each feature, making it accessible to individuals with varying levels of technical expertise. Additionally, error handling mechanisms are implemented to address unforeseen issues gracefully.

Future Enhancements

While the current version of the To-Do List project fulfils basic task management needs, there is ample room for future enhancements. Potential features include task prioritization, deadline reminders, and the ability to categorize tasks. Integrating a graphical user interface (GUI) could further enhance the project's accessibility and visual appeal.


adsa.py - C:\Users\Sakshi Singh\OneDrive\Desktop - Geany (new instance)
File Edit Search View Document Project Build Tools Help
Symbols
Functions
main [1]
adsa.py
choice = input("Enter your choice: ")
if choice == '1':
print()
n_tasks = int(input("How may task you want to add: "))
for i in range(n_tasks):
task = input("Enter the task: ")
tasks.append({"task": task, "done": False})
print("Task added!")
elif choice == '2':
print("\nTasks:")
for index, task in enumerate(tasks):
status = "Done" if task["done"] else "Not Done"
print(f"{index + 1}. {task['task']} - {status}")
elif choice == '3':
task_index = int(input("Enter the task number to mark as done: ")) - 1
if 0 <= task_index < len(tasks):
tasks[task_index]["done"] = True
print("Task marked as done!")
else:
print("Invalid task number.")
elif choice == '4':
print("Exiting the To-Do List.")
break
else:
print("Invalid choice. Please try again.")
if __name__ == "__main__":
main()
C:\WINDOWS\SYSTEM32\cmd
===== To-Do List =====
1. Add Task
2. Show Tasks
3. Mark Task as Done
4. Exit
Enter your choice: 1
How may task you want to add: 1
Enter the task: adsa
Task added!
===== To-Do List =====
1. Add Task
2. Show Tasks
3. Mark Task as Done
4. Exit
Enter your choice: 2
Tasks:
1. adsa - Not Done
===== To-Do List =====
1. Add Task
2. Show Tasks
3. Mark Task as Done
4. Exit
Enter your choice: 3
Enter the task number to mark as done: 1
Task marked as done!
Status
Compiler
Messages
Scribble
py -m py_compile "adsa.py" (in directory: C:\Users\Sakshi Singh\OneDrive\Desktop)
Compilation finished successfully.
line: 39 / 45 col: 0 sel: 0 INS TAB EOL: CRLF encoding: UTF-8 filetype: Python scope: main
Search
ENG
IN
12:31
24-01-2024

Conclusion

In conclusion, the Python-based To-Do List project offers a robust solution for effective task management. By providing features such as adding tasks, displaying tasks, marking tasks as done, and a smooth exit option, the project caters to users seeking a straightforward yet powerful tool. The implementation details, focus on user experience, and considerations for future enhancements position this project as a valuable asset for individuals looking to streamline their daily responsibilities.