

K.RAMAKRISHNAN
COLLEGE OF TECHNOLOGY
(AN AUTONOMOUS INSTITUTION)
SAMAYAPURAM, TRICHY-621 112



logo

Practical Record Note

Name : THARUN KUMAR M
Register Number : 2303811710621114
Subject code/name : Laboratory
Programme :

Page

COURSE: Python Programming - I Year - II Sem - Project Module

ID: 2303811710621114>

NAME: THARUN KUMAR M

No. 1

K.RAMAKRISHNAN
COLLEGE OF TECHNOLOGY
(AN AUTONOMOUS INSTITUTION)
SAMAYAPURAM, TRICHY-621 112



logo

Page

COURSE: Python Programming - I Year - II Sem - Project Module

ID: 2303811710621114>

NAME: THARUN KUNAR M

No. 2

Certified that this is a bonafide record of work done by
THARUN KUMAR M of _____
Semester in **Python Programming - I Year - II Sem - Project**
Module Laboratory during the academic year 2023-2024

His/Her University Register Number is **2303811710621114**



signature

Staff Incharge

 signature

Head of the Department

Submitted for the Practical exam held on:



Signature

Page

COURSE: Python Programming - I Year - II Sem - Project Module

ID: 2303811710621114>

NAME: THARUN KUNAR M

No. 4

Internal Examiner
Date:



sign
External Examiner
Date:

Page

COURSE: Python Programming - I Year - II Sem - Project Module

ID: 2303811710621114>

NAME: THARUN KUNAR M

No. 5

Aim:

Project Module.

Program:

CTP28132.py

CodeTantira

Page

COURSE: Python Programming - I Year - II Sem - Project Module

ID: 230381710621114>

NAME: THARUN KUMAR M

No. 6

```

import datetime

class TodoItem:
    def __init__(self, description, due_date=None):
        self.description = description
        self.completed = False
        self.due_date = due_date

    def __str__(self):
        status = "Completed" if self.completed else "Pending"
        due_date_str = self.due_date.strftime("%Y-%m-%d") if self.due_date else
        "No due date"
        return f"{self.description} - {status} - Due: {due_date_str}"

class TodoApp:
    def __init__(self):
        self.tasks = []

    def add_task(self, description, due_date=None):
        task = TodoItem(description, due_date)
        self.tasks.append(task)
        print("Task added!")

    def delete_task(self, index):
        try:
            self.tasks.pop(index)
            print("Task deleted!")
        except IndexError:
            print("Invalid index!")

    def edit_task(self, index, new_description):
        try:
            self.tasks[index].description = new_description
            print("Task edited!")
        except IndexError:
            print("Invalid index!")

    def mark_as_completed(self, index):
        try:
            self.tasks[index].completed = True
            print("Task marked as completed!")
        except IndexError:
            print("Invalid index!")

    def set_due_date(self, index, due_date):
        try:
            self.tasks[index].due_date = due_date
            print("Due date set!")
        except IndexError:
            print("Invalid index!")

    def show_tasks(self):
        if not self.tasks:
            print("No tasks to show.")
        for i, task in enumerate(self.tasks):
            print(f"{i}: {task}")

```

```

def parse_date(date_str):
    return datetime.datetime.strptime(date_str, "%Y-%m-%d")

def main():
    app = TodoApp()

    while True:
        print("\nMenu:")
        print("1. Add task")
        print("2. Delete task")
        print("3. Edit task")
        print("4. Mark task as completed")
        print("5. Set task due date")
        print("6. Show tasks")
        print("7. Quit")

        option = input("Choose an option (1-7): ").strip()

        if option == "1":
            description = input("Task description: ").strip()
            due_date_str = input("Due date (YYYY-MM-DD) [optional]: ").strip()
            due_date = parse_date(due_date_str) if due_date_str else None
            app.add_task(description, due_date)

        elif option == "2":
            index = int(input("Task index to delete: ").strip())
            app.delete_task(index)

        elif option == "3":
            index = int(input("Task index to edit: ").strip())
            new_description = input("New task description: ").strip()
            app.edit_task(index, new_description)

        elif option == "4":
            index = int(input("Task index to mark as completed: ").strip())
            app.mark_as_completed(index)

        elif option == "5":
            index = int(input("Task index to set due date: ").strip())
            due_date_str = input("New due date (YYYY-MM-DD): ").strip()
            due_date = parse_date(due_date_str)
            app.set_due_date(index, due_date)

        elif option == "6":
            app.show_tasks()

        elif option == "7":
            break

        else:
            print("Invalid option! Please choose a number between 1 and 7.")

if __name__ == "__main__":
    main()

```

Output:

Test case - 1
User Output
Hello World
Hello World

Result:

Thus the above program is executed successfully and the output has been verified

CodeTantira

