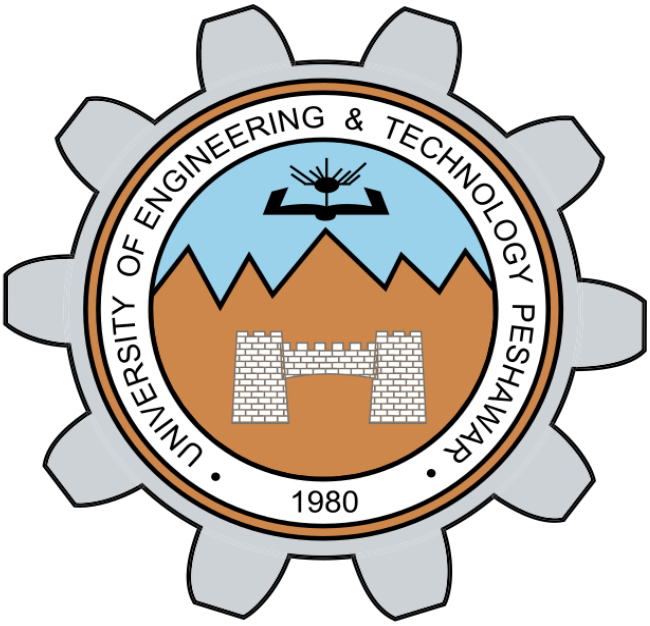
UNIVERSITY OF ENGINEERING AND TECHNOLOGY PESHAWAR



SUBMITTED BY: Waqas Khan

SUBMITTED TO: SIR ADEEL ALI SHAH

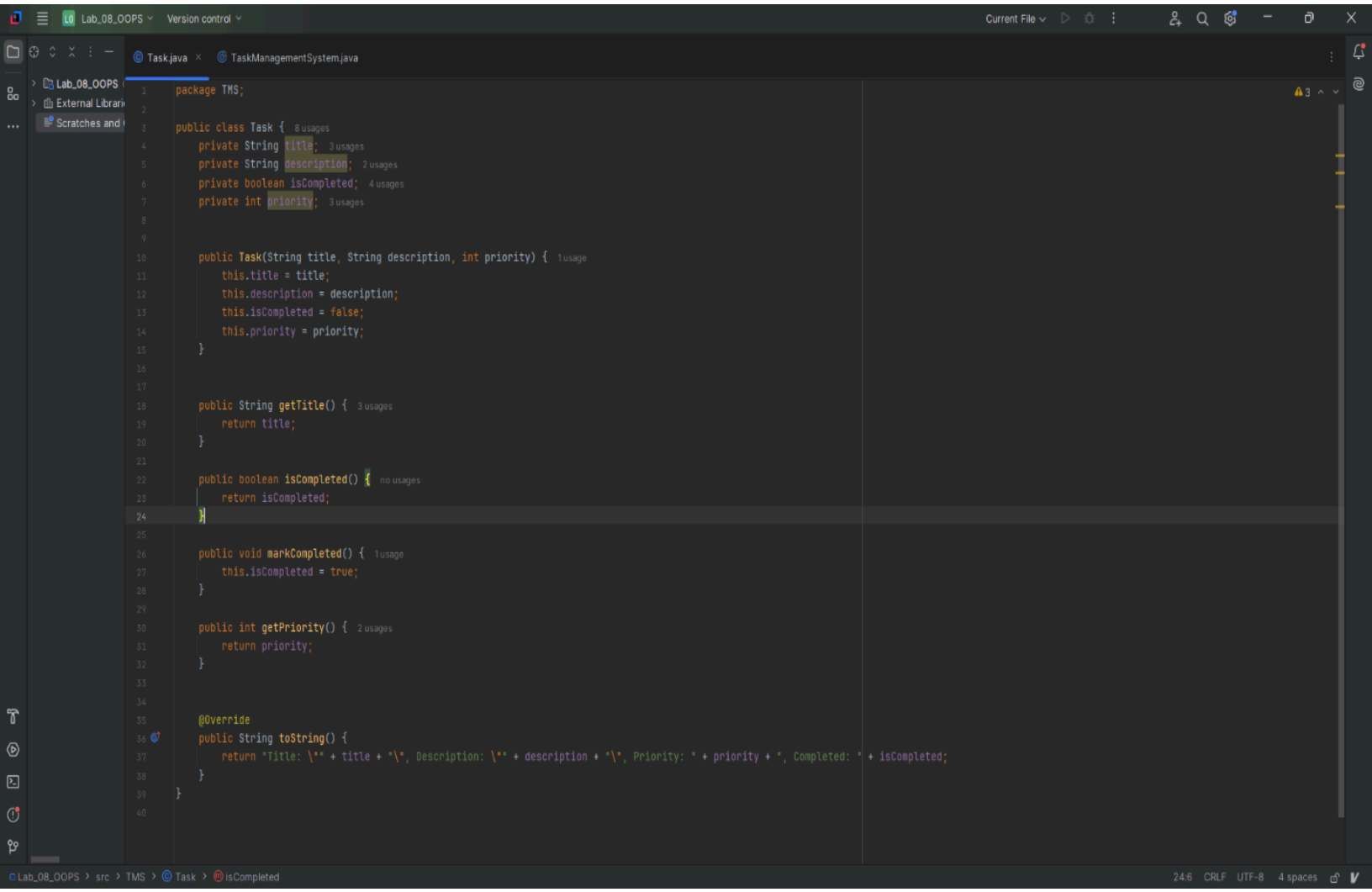
SECTION: "A"

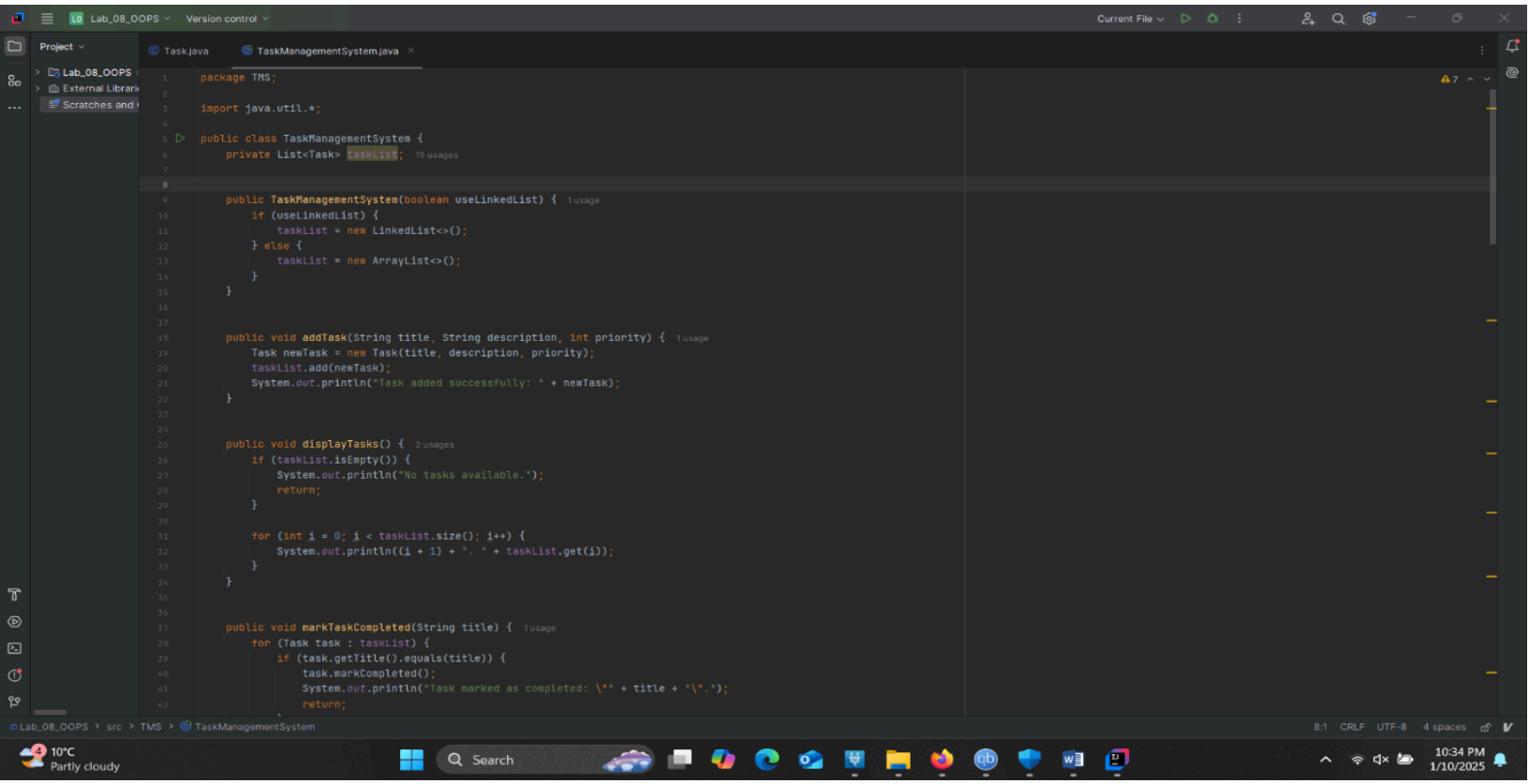
REG NO # 24PWBCS1141

DATE: 9/January/2024

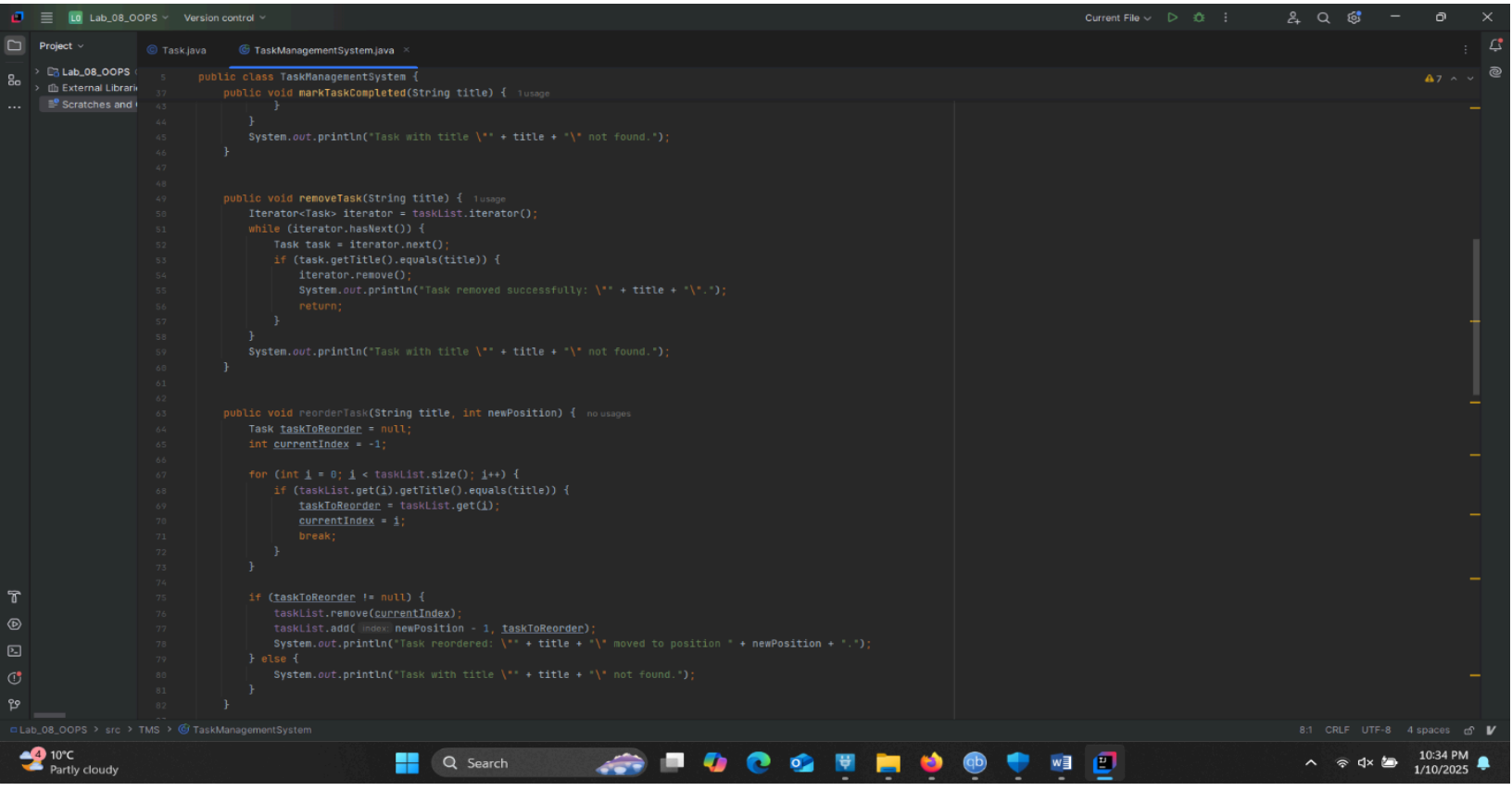
TASK NO: 2

DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY PESHAWAR

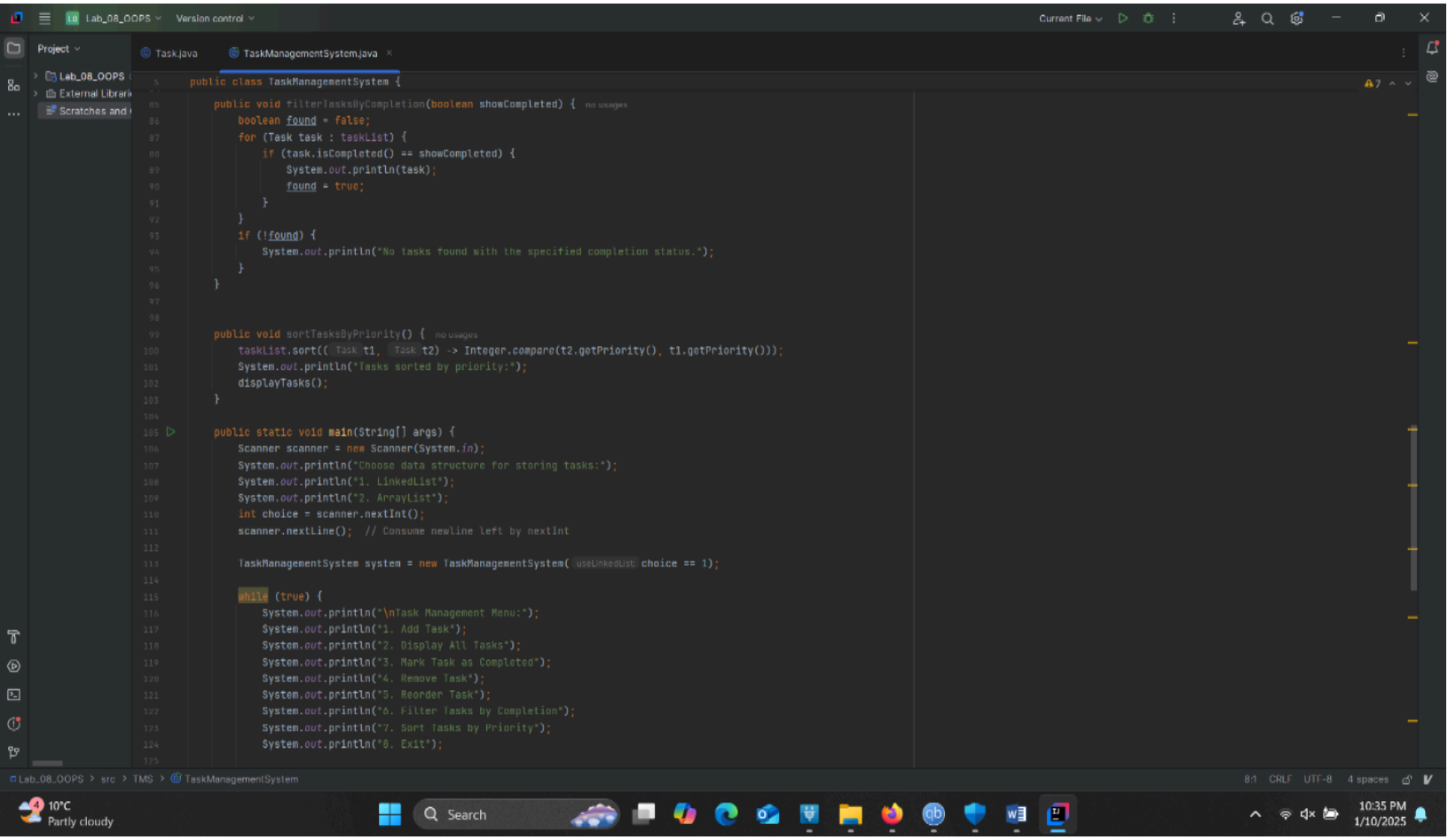

Lab_08_OOPS
Version control
Current File
Task.java
TaskManagementSystem.java
Lab_08_OOPS
External Librari
Scratches and
package TMS;
public class Task { 8 usages
private String title; 3 usages
private String description; 2 usages
private boolean isCompleted; 4 usages
private int priority; 3 usages
public Task(String title, String description, int priority) { 1 usage
this.title = title;
this.description = description;
this.isCompleted = false;
this.priority = priority;
}
public String getTitle() { 3 usages
return title;
}
public boolean isCompleted() { no usages
return isCompleted;
}
public void markCompleted() { 1 usage
this.isCompleted = true;
}
public int getPriority() { 2 usages
return priority;
}
@Override
public String toString() {
return "Title: \"" + title + "\", Description: \"" + description + "\", Priority: " + priority + ", Completed: " + isCompleted;
}
}
Lab_08_OOPS > src > TMS > Task > isCompleted
24:6 CRLF UTF-8 4 spaces

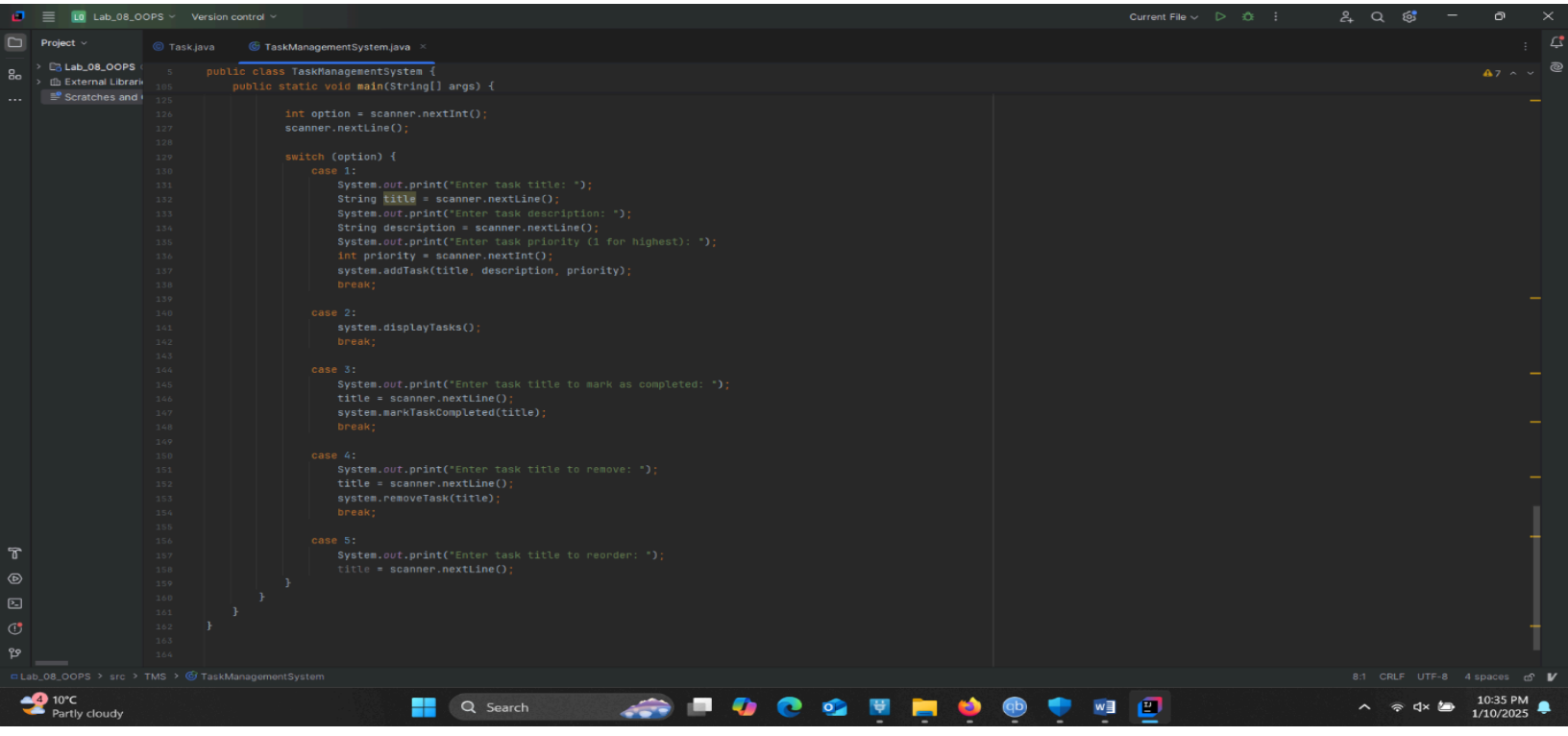


```java
package TMS;

import java.util.*;

public class TaskManagementSystem {
    private List<Task> taskList;  15 usages

    public TaskManagementSystem(boolean useLinkedList) {  1 usage
        if (useLinkedList) {
            taskList = new LinkedList<>();
        } else {
            taskList = new ArrayList<>();
        }
    }


    public void addTask(String title, String description, int priority) {  1 usage
        Task newTask = new Task(title, description, priority);
        taskList.add(newTask);
        System.out.println("Task added successfully: " + newTask);
    }


    public void displayTasks() {  2 usages
        if (taskList.isEmpty()) {
            System.out.println("No tasks available.");
            return;
        }

        for (int i = 0; i < taskList.size(); i++) {
            System.out.println((i + 1) + ". " + taskList.get(i));
        }
    }


    public void markTaskCompleted(String title) {  1 usage
        for (Task task : taskList) {
            if (task.getTitle().equals(title)) {
                task.markCompleted();
                System.out.println("Task marked as completed: \"" + title + "\".");
                return;
```



```java
public class TaskManagementSystem {
    public void markTaskCompleted(String title) {  1 usage
            }
        }
        System.out.println("Task with title \"" + title + "\" not found.");
    }


    public void removeTask(String title) {  1 usage
        Iterator<Task> iterator = taskList.iterator();
        while (iterator.hasNext()) {
            Task task = iterator.next();
            if (task.getTitle().equals(title)) {
                iterator.remove();
                System.out.println("Task removed successfully: \"" + title + "\".");
                return;
            }
        }
        System.out.println("Task with title \"" + title + "\" not found.");
    }


    public void reorderTask(String title, int newPosition) {  no usages
        Task taskToReorder = null;
        int currentIndex = -1;

        for (int i = 0; i < taskList.size(); i++) {
            if (taskList.get(i).getTitle().equals(title)) {
                taskToReorder = taskList.get(i);
                currentIndex = i;
                break;
            }
        }

        if (taskToReorder != null) {
            taskList.remove(currentIndex);
            taskList.add( index: newPosition - 1, taskToReorder);
            System.out.println("Task reordered: \"" + title + "\" moved to position " + newPosition + ".");
        } else {
            System.out.println("Task with title \"" + title + "\" not found.");
        }
    }
```


Lab_08_OOPS
Version control
Current File
Project
Task.java
TaskManagementSystem.java
Lab_08_OOPS
External Librari
Scratches and
public class TaskManagementSystem {
public void filterTasksByCompletion(boolean showCompleted) { no usages
boolean found = false;
for (Task task : taskList) {
if (task.isCompleted() == showCompleted) {
System.out.println(task);
found = true;
}
}
if (!found) {
System.out.println("No tasks found with the specified completion status.");
}
}
public void sortTasksByPriority() { no usages
taskList.sort((Task t1, Task t2) -> Integer.compare(t2.getPriority(), t1.getPriority()));
System.out.println("Tasks sorted by priority:");
displayTasks();
}
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Choose data structure for storing tasks:");
System.out.println("1. LinkedList");
System.out.println("2. ArrayList");
int choice = scanner.nextInt();
scanner.nextLine(); // Consume newline left by nextInt
TaskManagementSystem system = new TaskManagementSystem(useLinkedList: choice == 1);
while (true) {
System.out.println("\nTask Management Menu:");
System.out.println("1. Add Task");
System.out.println("2. Display All Tasks");
System.out.println("3. Mark Task as Completed");
System.out.println("4. Remove Task");
System.out.println("5. Reorder Task");
System.out.println("6. Filter Tasks by Completion");
System.out.println("7. Sort Tasks by Priority");
System.out.println("8. Exit");
Lab_08_OOPS > src > TMS > TaskManagementSystem
8:1 CRLF UTF-8 4 spaces
10°C
Partly cloudy
Search
10:35 PM
1/10/2025


Lab_08_OOPS
Version control
Current File
Project
Task.java
TaskManagementSystem.java
Lab_08_OOPS
External Librari
Scratches and
public class TaskManagementSystem {
public static void main(String[] args) {
int option = scanner.nextInt();
scanner.nextLine();
switch (option) {
case 1:
System.out.print("Enter task title: ");
String title = scanner.nextLine();
System.out.print("Enter task description: ");
String description = scanner.nextLine();
System.out.print("Enter task priority (1 for highest): ");
int priority = scanner.nextInt();
system.addTask(title, description, priority);
break;
case 2:
system.displayTasks();
break;
case 3:
System.out.print("Enter task title to mark as completed: ");
title = scanner.nextLine();
system.markTaskCompleted(title);
break;
case 4:
System.out.print("Enter task title to remove: ");
title = scanner.nextLine();
system.removeTask(title);
break;
case 5:
System.out.print("Enter task title to reorder: ");
title = scanner.nextLine();
}
}
}
Lab_08_OOPS > src > TMS > TaskManagementSystem
8:1 CRLF UTF-8 4 spaces
10°C
Partly cloudy
Search
10:35 PM
1/10/2025

Run TaskManagementSystem

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.3.1.1\lib\idea_rt.jar=58451:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.3.1.1\bin" -Dfile.enc
Choose data structure for storing tasks:
1. LinkedList
2. ArrayList
1

Task Management Menu:
1. Add Task
2. Display All Tasks
3. Mark Task as Completed
4. Remove Task
5. Reorder Task
6. Filter Tasks by Completion
7. Sort Tasks by Priority
8. Exit
1
Enter task title: Task1
Enter task description: No
Enter task priority (1 for highest): 1
Task added successfully: Title: "Task1", Description: "No", Priority: 1, Completed: false

Task Management Menu:
1. Add Task
2. Display All Tasks
3. Mark Task as Completed
4. Remove Task
5. Reorder Task
6. Filter Tasks by Completion
7. Sort Tasks by Priority
8. Exit
2
1. Title: "Task1", Description: "No", Priority: 1, Completed: false

Task Management Menu:
1. Add Task
2. Display All Tasks
3. Mark Task as Completed
4. Remove Task
5. Reorder Task
6. Filter Tasks by Completion
7. Sort Tasks by Priority
8. Exit
3
Enter task title to mark as completed: 1
```