**RIPHAH INTERNATIONAL UNIVERSITY**



**Submitted to:**

Mam Shazwa

**Submitted by:**

Maryam Sohail **(45773)**

Task:

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

// --- Model class representing a Task ---

class Task {

private int id;

private String title;

private String description;

private boolean isCompleted;

// Constructor to initialize a Task

public Task(int id, String title, String description) {

this.id = id;

this.title = title;

this.description = description;

this.isCompleted = false;

}

// Getters and setters for Task attributes

public int getId() {

return id;

}

public String getTitle() {

return title;

}

public void setTitle(String title) {

this.title = title;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

public boolean isCompleted() {

return isCompleted;

}

public void setCompleted(boolean completed) {

isCompleted = completed;

}

// Overridden toString method to display task information

@Override

public String toString() {

return "ID: " + id + ", Title: " + title + ", Description: " + description + ", Completed: " + isCompleted;

}

}

// --- View class responsible for displaying task information ---

class TaskView {

// Method to display all tasks

public void displayTasks(List<Task> tasks) {

if (tasks.isEmpty()) {

System.out.println("No tasks available.");

} else {

for (Task task : tasks) {

System.out.println(task);

}

}

}

// Method to display confirmation of a task being added

public void displayTaskAdded(Task task) {

System.out.println("Task added: " + task);

}

// Method to display confirmation of marking a task as completed

public void displayTaskMarkedCompleted(int taskId) {

System.out.println("Task with ID " + taskId + " marked as completed.");

}

}

// --- Controller class to manage task data and interactions ---

class TaskController {

private List<Task> tasks = new ArrayList<>();

private TaskView view;

private int nextId = 1;

// Constructor to initialize the controller with a view

public TaskController(TaskView view) {

this.view = view;

}

// Method to add a new task

public void addTask(String title, String description) {

Task task = new Task(nextId++, title, description);

tasks.add(task);

view.displayTaskAdded(task);

}

// Method to display all tasks via the view

public void displayAllTasks() {

view.displayTasks(tasks);

}

// Method to mark a task as completed by its ID

public void markTaskAsCompleted(int taskId) {

for (Task task : tasks) {

if (task.getId() == taskId) {

task.setCompleted(true);

view.displayTaskMarkedCompleted(taskId);

return;

}

}

System.out.println("Task with ID " + taskId + " not found.");

}

}

// --- Main application class to run the console-based application ---

public class Main {

public static void main(String[] args) {

TaskView view = new TaskView();

TaskController controller = new TaskController(view);

Scanner scanner = new Scanner(System.in);

System.out.println("Task Management Application");

while (true) {

System.out.println("\n1. Add Task");

System.out.println("2. View All Tasks");

System.out.println("3. Mark Task as Completed");

System.out.println("4. Exit");

System.out.print("Enter choice: ");

int choice = scanner.nextInt();

scanner.nextLine(); // Consume newline

switch (choice) {

case 1:

System.out.print("Enter task title: ");

String title = scanner.nextLine();

System.out.print("Enter task description: ");

String description = scanner.nextLine();

controller.addTask(title, description);

break;

case 2:

controller.displayAllTasks();

break;

case 3:

System.out.print("Enter task ID to mark as completed: ");

int taskId = scanner.nextInt();

controller.markTaskAsCompleted(taskId);

break;

case 4:

System.out.println("Exiting...");

scanner.close();

return;

default:

System.out.println("Invalid choice. Try again.");

}

}

}

}

Output:

Task Management Application

1. Add Task

2. View All Tasks

3. Mark Task as Completed

4. Exit

Enter choice: 1

Enter task title: abc

Enter task description: sleep

Task added: ID: 1, Title: abc, Description: sleep, Completed: false

1. Add Task

2. View All Tasks

3. Mark Task as Completed

4. Exit

Enter choice: 2

ID: 1, Title: abc, Description: sleep, Completed: false

1. Add Task

2. View All Tasks

3. Mark Task as Completed

4. Exit

Enter choice: 3

Enter task ID to mark as completed: 1

Task with ID 1 marked as completed.

1. Add Task

2. View All Tasks

3. Mark Task as Completed

4. Exit

Enter choice: 2

ID: 1, Title: abc, Description: sleep, Completed: true

1. Add Task

2. View All Tasks

3. Mark Task as Completed

4. Exit

Enter choice: