

**Department of Computer Technology****Vision of the Department***To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.***Mission of the Department***To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.***Session 2025-2026****Vision:** Dream of where you want.**Mission:** Means to achieve Vision**Program Educational Objectives of the program (PEO):** (broad statements that describe the professional and career accomplishments)

PEO1	Preparation	P: Preparation	Pep-CL abbreviation pronounce as Pep-si-IL easy to recall
PEO2	Core Competence	E: Environment (Learning Environment)	
PEO3	Breadth	P: Professionalism	
PEO4	Professionalism	C: Core Competence	
PEO5	Learning Environment	L: Breadth (Learning in diverse areas)	

Program Outcomes (PO): (statements that describe what a student should be able to do and know by the end of a program)**Keywords of POs:**

Engineering knowledge, Problem analysis, Design/development of solutions, Conduct Investigations of Complex Problems, Engineering Tool Usage, The Engineer and The World, Ethics, Individual and Collaborative Team work, Communication, Project Management and Finance, Life-Long Learning

PSO Keywords: Cutting edge technologies, Research

“I am an engineer, and I know how to apply engineering knowledge to investigate, analyse and design solutions to complex problems using tools for entire world following all ethics in a collaborative way with proper management skills throughout my life.” to contribute to the development of cutting-edge technologies and Research.

Integrity: I will adhere to the Laboratory Code of Conduct and ethics in its entirety.**Name and Signature of Student and Date**

(Signature and Date in Handwritten)

**Department of Computer Technology****Vision of the Department***To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.***Mission of the Department***To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.*

Session	2025-26 (ODD)	Course Name	Operating System Lab
Semester	5	Course Code	23IOT1504
Roll No	36	Name of Student	Darshil D. Amalkar

Practical Number	03
Course Outcome	<ol style="list-style-type: none">1. Understand Computer System Configuration and Simulate system resources efficiently using Linux Commands (CO1)2. Analyse operating system functionalities utilizing system calls, thread programming and process scheduling algorithms (CO2)3. Apply Synchronization primitives to implement a Deadlock-free solution(CO3)4. Simulate Disk scheduling, Memory allocation, File allocation, page replacement algorithms (CO4)
Aim	Implement Thread life cycle : Thread Creation and Termination
Problem Definition	Quiz Game Simulation Create a thread that gives a countdown timer (10 to 1). Another thread prints the quiz question and answer check. Both terminate after the question is answered or time runs out.
Theory (100 words)	<p>Think of a thread in Java as a worker hired to complete a specific task list (its run() method). Like any worker, it goes through several distinct phases from the moment it's created until its job is done.</p> <p>1. New: Hired, but Waiting for the First Day</p> <p>When you first create a thread object (new Thread()), it's in the New state. At this point, it's like a newly hired employee. The paperwork is done, the employee exists, but they haven't actually started doing any work yet. The thread is just a potential worker holding its to-do list, waiting for the call to begin.</p>



Department of Computer Technology

Vision of the Department

To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.

Mission of the Department

To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.

2. Runnable: Ready and Waiting to Work

Once you call the start() method, the thread enters the **Runnable** state. This doesn't mean it's actively running at this very second. Instead, think of it as the worker arriving at the office, clocking in, and telling the manager, "I'm here and ready for an assignment!" It's now in a pool with all other "runnable" threads, and it's up to the system's thread scheduler (the manager) to give it a slice of CPU time to execute its task.

3. Waiting: Taking an Indefinite Pause

A thread enters the **Waiting** state when it needs to pause indefinitely until another thread gives it a specific signal. Imagine our worker has to stop because they're waiting for a colleague to deliver a critical document. They can't proceed on their own and have no idea how long it will take. They are effectively on standby, waiting for that "tap on the shoulder" from another thread to wake them up and send them back to the **Runnable** state.

4. Timed Waiting: Taking a Scheduled Break

This is a pause with a deadline. A thread enters the **Timed Waiting** state when it's told to wait for a specific amount of time (like using Thread.sleep()) or until a certain event occurs. This is like a worker taking a scheduled 15-minute coffee break. They will automatically wake up and return to the **Runnable** state once their break time is over, ready to get back to work.

5. Terminated (Dead): The Job is Done

A thread reaches the **Terminated** state when it has successfully completed every task on its to-do list (i.e., its run() method finishes). Its journey is over. The worker has finished the project, clocked out for the last time, and gone home. Once a thread is



Department of Computer Technology

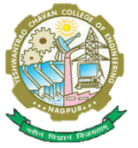
Vision of the Department

To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.

Mission of the Department

To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.

	<p>terminated, it cannot be restarted. Its life cycle is complete.</p> <pre>graph LR; NT(New Thread()) --> New; New -- Start() --> Runnable; Runnable -- run() --> Running; Running -- Sleep(), wait() --> Waiting; Waiting -- End of execution --> Dead; New --> Dead;</pre>
Procedure and Execution (100 Words)	<p>Step for Implementation:</p> <p>To create a quiz game where a timer runs while the user answers, you use two separate threads in Java.</p> <ul style="list-style-type: none">● Shared Signal (AtomicBoolean): A central "game over" flag that both threads can see. When the user answers, this flag is set to true.● Timer Thread: Counts down from 10. It stops either when it hits zero or when it sees the "game over" flag has been set by the other thread.● Quiz Thread: Prints the question and waits for the user's input. Once the user answers, it checks the result and immediately sets the "game over" flag to true, stopping the timer. <p>Code:</p> <pre>import java.util.Scanner; import java.util.concurrent.atomic.AtomicBoolean; public class QuizGame { static AtomicBoolean isAnswered = new AtomicBoolean(false);</pre>



Department of Computer Technology

Vision of the Department

To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.

Mission of the Department

To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.

```
public static void main(String[] args) {
    System.out.println("Quiz starts now! You have 10
seconds. ");

    Thread timerThread = new Thread(new Timer());
    Thread quizThread = new Thread(new Quiz());

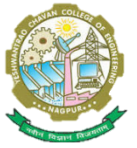
    timerThread.start();
    quizThread.start();
}

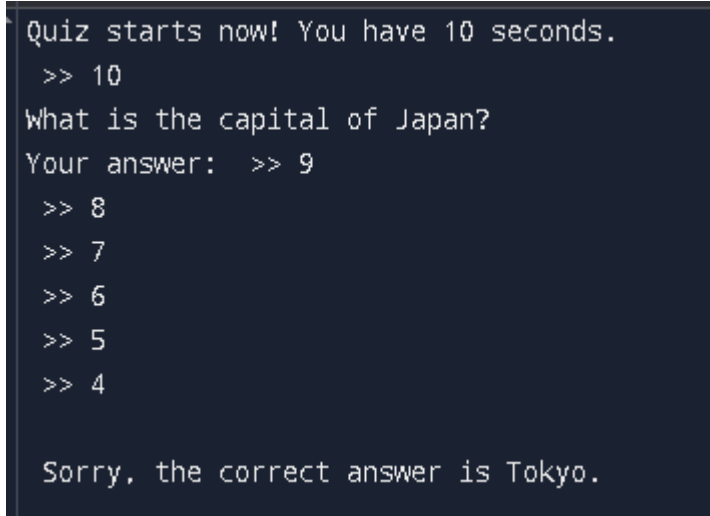
class Timer implements Runnable {
    @Override
    public void run() {
        try {
            for (int i = 10; i >= 1; i--) {
                if (QuizGame.isAnswered.get()) {
                    return; // Exit the run method
                }

                System.out.println(">> " + i);
                Thread.sleep(1000); // Wait for 1 second
            }
        } catch (InterruptedException e) {
            System.out.println("Timer was interrupted.");
        }

        if (!QuizGame.isAnswered.get()) {
            System.out.println("\n TIME'S UP! ");
            System.exit(0); // Terminate the whole program
        }
    }
}

class Quiz implements Runnable {
    @Override
    public void run() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("What is the capital of Japan?");
        System.out.print("Your answer: ");
```

**Department of Computer Technology****Vision of the Department***To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.***Mission of the Department***To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.*

	<pre>String answer = scanner.nextLine(); QuizGame.isAnswered.set(true); if (answer.equalsIgnoreCase("Tokyo")) { System.out.println(" Correct! You're a winner!"); } else { System.out.println(" Sorry, the correct answer is Tokyo."); } scanner.close(); System.exit(0); }</pre>
	<p>Output:</p> 
Output Analysis	The timer is ticking until I type something and once the time's up it gives the answer to the asked question.
Link of student Github profile where lab assignment has been uploaded	https://github.com/Darshil-yup/OS_Lab.git
Conclusion	In the end, viewing a thread's life cycle as a worker's journey makes it much easier to grasp. From being hired (New) to being ready for tasks (Runnable), taking breaks (Waiting),



Nagar Yuwak Shikshan Sanstha's

Yeshwantrao Chavan College of Engineering

(An Autonomous Institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University)

Hingna Road, Wanadongri, Nagpur - 441 110

NAAC A++

Ph.: 07104-237919, 234623, 329249, 329250 Fax: 07104-232376, Website: www.ycce.edu



Department of Computer Technology

Vision of the Department

To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.

Mission of the Department

To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.

	and finally retiring (Terminated), each state has a clear purpose. Understanding this flow is crucial because it allows you to manage how different parts of your program cooperate or pause without causing chaos. It's the key to writing efficient, predictable code where multiple tasks run in harmony.
Plag Report (Similarity index < 12%)	<div><div><div>5%</div><div>Plagiarism</div></div><div><div>Exact Match</div><div>5%</div></div><div><div>Partial Match</div><div>0%</div></div><div><div>95%</div><div>Unique</div></div></div>
Date	20-08-2025