**Management Education and Research Institute  
52-55, INSTITUTIONAL AREA, JANAKPURI, NEW DELHI – 110058  
Tel: 11-28522201-04 • E-mail: meribs@meri.edu.in • Web: https://meri.edu.in**

Contact No.: 8368448315

E-mail:

lavishagupta67@gmail.com

Address: E- 32-A

Sudershan park, Moti nagar

**Education: B.Tech,(Computer Science Engineering)**

**MERI College Of Engineering and Technology**

**2021 - 2025**

**Skills: Java (oops, swing- developing GUI, JDBC, Junit, servlets, JSP),**

**HTML, CSS**

**Languages: English, Hindi**

**Hobbies: Craft work**

|  |
| --- |
| Course Name: B tech • Semester: 7th • Batch: 2021-25  • Roll No.: 21CET0063 |

**Lavisha gupta**



**Profile: A motivated and skilled Java developer with hands-on experience in Java. I have successfully completed 2-3 projects, demonstrating my ability to write clean, efficient, and maintainable code. I am passionate about leveraging my technical skills to solve real-world problems and am eager to contribute to challenging projects.**

**Internship Experience: Data Analytics**

**IBM SKILLBUILD INTERNSHIP, Virtual**

**Internship • 24th June2024 to 5thAugust 2024**

**Analysed large datasets, created visualizations, and generated reports to support data-driven decisions. Improved data quality, streamlined data collection processes, and collaborated on key projects.**

**Project:**

**1. Text Editor**

**Nov 2023**

**In this project, we will create a simple text editor with a menu bar that contains two menus: File and Edit. The File menu will allow users to create new text files, open existing files, and save files to disk. The Edit menu will provide options for copying, cutting, and pasting text.**

**2. Student Management System**

**Jul 2023**

**Create a student management system that uses arrays and regular expressions to manage student records. The system should allow the user to add, sort, view, and search for student records.**

**3. Online Chatting application**

**Feb 2024**

**Create a chat application that allows multiple users to connect and chat with each other in real-time.  
  
Networking:  
  
Implement a server that listens for incoming connections from clients and routes messages between connected clients.  
  
Implement a client that can connect to the server, send and receive messages from server.  
Multithreading:  
  
Use a separate thread to listen incoming messages from server so the GUI remains responsive and doesn't block while waiting for incoming messages.  
  
Use a separate thread to send messages to the server so GUI remains responsive and doesn't block while sending messages.  
  
In server implementation, use a thread for each client that receives messages from client and reroutes it to the server to broadcast it  
  
GUI using Swing:  
  
Create a main window that allows users to enter their username and connect to the chat room.  
  
Use Swing components such as JTextArea and JTextField to implement the chat window and allow users to enter messages.**