NAME: GANESH N MOROLIYA

Level 3 output:

Level 3

Task 1

Cognifyz
Where Data Meets Intelligence

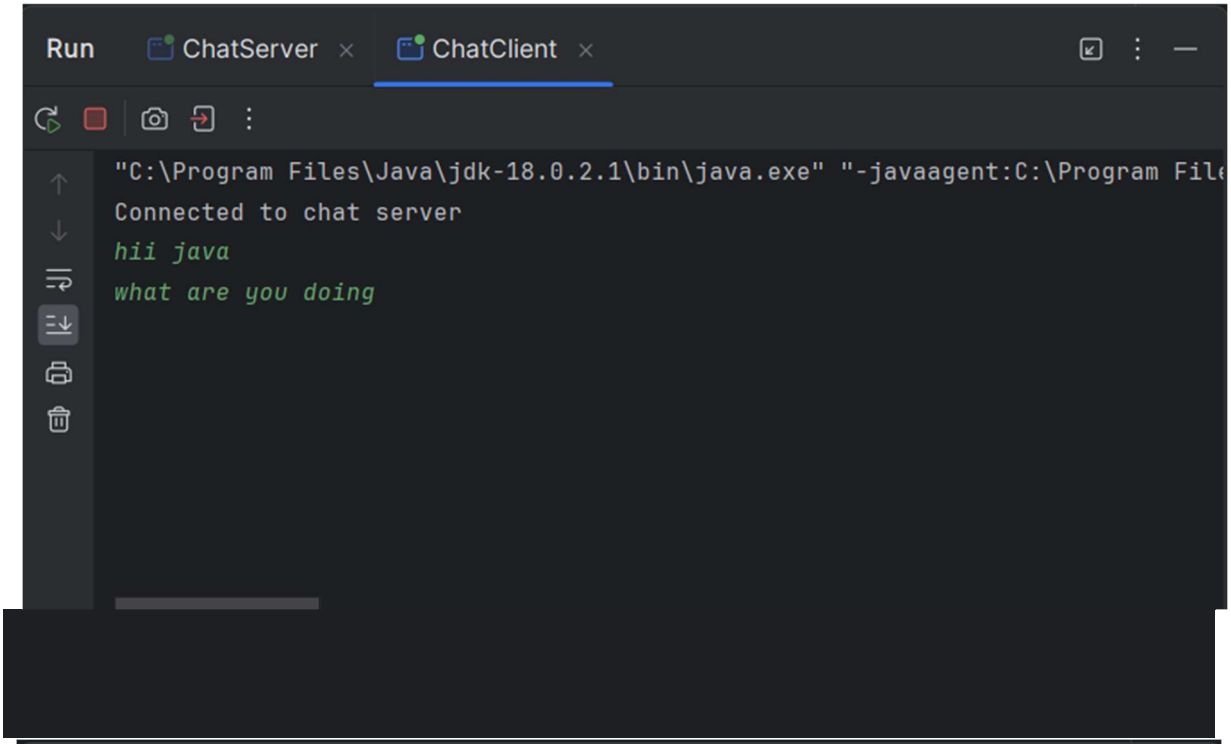
Task: Implement a Chat Application

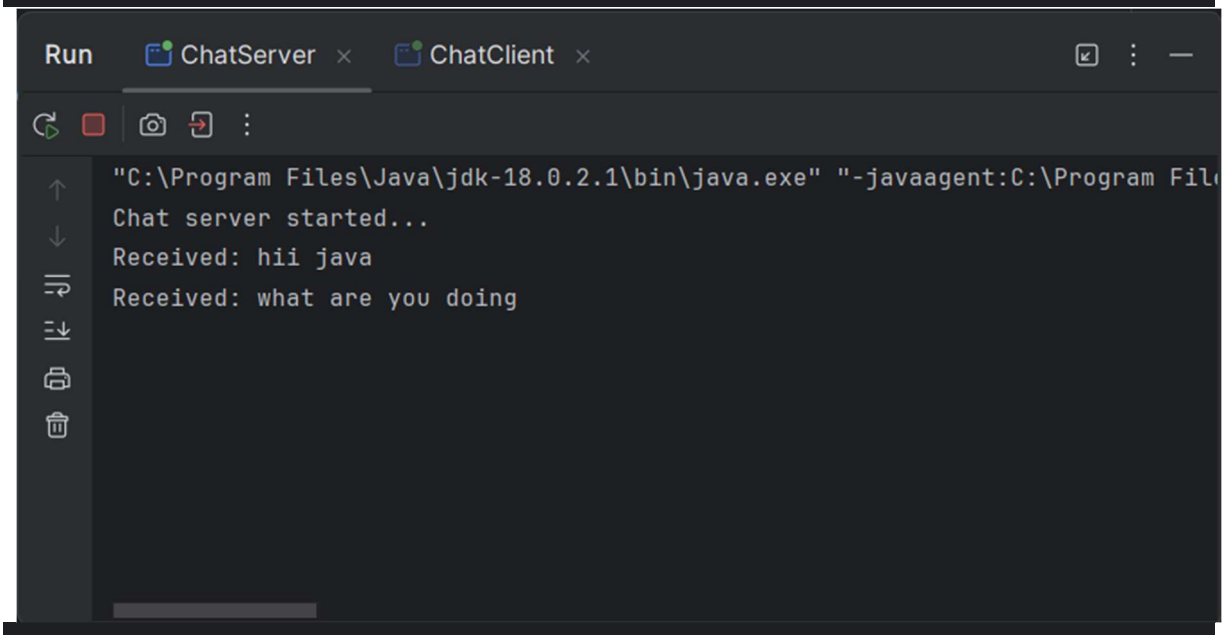
Develop a chat application using Java and networking libraries like Socket or Java NIO. This task will enhance their knowledge of network programming, client-server architecture, and message exchange.

```java
package levelTask1;

import java.io.*;
import java.net.*;

public class ChatClient {
    private static final String SERVER_ADDRESS = "localhost";
    private static final int SERVER_PORT = 12345;

    public static void main(String[] args) {
        try (Socket socket = new Socket(SERVER_ADDRESS, SERVER_PORT);
             BufferedReader consoleReader = new BufferedReader(new
InputStreamReader(System.in));
             PrintWriter out = new PrintWriter(socket.getOutputStream(),
true);
             BufferedReader in = new BufferedReader(new
InputStreamReader(socket.getInputStream()))) {

            System.out.println("Connected to chat server");

            Thread readThread = new Thread(() -> {
                try {
                    String message;
                    while ((message = in.readLine()) != null) {
                        System.out.println("Received: " + message);
                    }
                } catch (IOException e) {
                    e.printStackTrace();
                }
            });

            readThread.start();

            String userInput;
            while ((userInput = consoleReader.readLine()) != null) {
                out.println(userInput);
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}



package levelTask1;

import java.io.*;
import java.net.*;
import java.util.*;

public class ChatServer {
    private static final int PORT = 12345;
    private static Set<ClientHandler> clientHandlers = new HashSet<>();

    public static void main(String[] args) {
        System.out.println("Chat server started...");
        try (ServerSocket serverSocket = new ServerSocket(PORT)) {
            while (true) {
                Socket socket = serverSocket.accept();
                ClientHandler clientHandler = new ClientHandler(socket);
                clientHandlers.add(clientHandler);
```

```
                new Thread(clientHandler).start();
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    public static void broadcast(String message, ClientHandler
excludeClient) {
        for (ClientHandler client : clientHandlers) {
            if (client != excludeClient) {
                client.sendMessage(message);
            }
        }
    }

    public static void removeClient(ClientHandler clientHandler) {
        clientHandlers.remove(clientHandler);
    }
}

class ClientHandler implements Runnable {
    private Socket socket;
    private PrintWriter out;
    private BufferedReader in;

    public ClientHandler(Socket socket) {
        this.socket = socket;
    }

    @Override
    public void run() {
        try {
            in = new BufferedReader(new
InputStreamReader(socket.getInputStream()));
            out = new PrintWriter(socket.getOutputStream(), true);

            String message;
            while ((message = in.readLine()) != null) {
                System.out.println("Received: " + message);
                ChatServer.broadcast(message, this);
            }
        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            try {
                socket.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
            ChatServer.removeClient(this);
        }
    }

    public void sendMessage(String message) {
        out.println(message);
    }
}
```

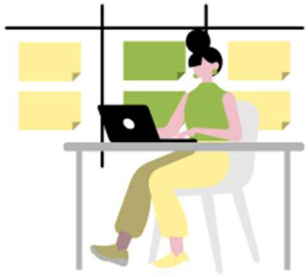
Output:


Run
ChatServer
ChatClient
"C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe" "-javaagent:C:\Program File
Connected to chat server
hii java
what are you doing


Run
ChatServer
ChatClient
"C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe" "-javaagent:C:\Program File
Chat server started...
Received: hii java
Received: what are you doing

Level 3

Task 2



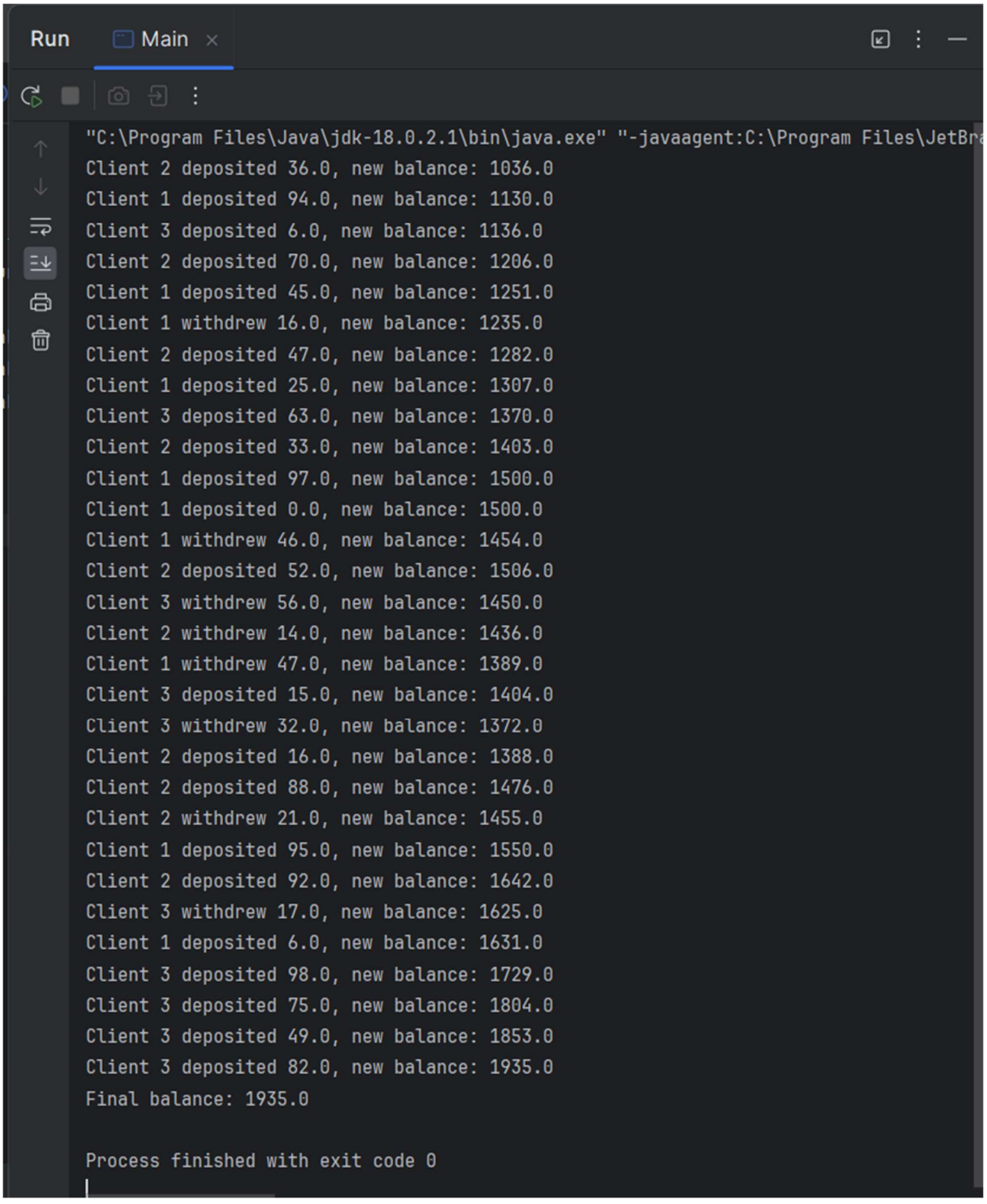
Task: Create a Multi-threaded Application

Build a multi-threaded Java application that demonstrates concurrency concepts like synchronization, locks, and threads. This task will help them understand concurrent programming and handling shared resources.

```
package leveltask2;

public class BankAccount {
    private double balance;

    public BankAccount(double initialBalance) {
        this.balance = initialBalance;
    }

    public synchronized void deposit(double amount) {
        balance += amount;
        System.out.println(Thread.currentThread().getName() + " deposited "
+ amount + ", new balance: " + balance);
    }

    public synchronized void withdraw(double amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println(Thread.currentThread().getName() + "
withdrew " + amount + ", new balance: " + balance);
        } else {
            System.out.println(Thread.currentThread().getName() + " tried
to withdraw " + amount + " but insufficient funds, balance: " + balance);
        }
    }

    public double getBalance() {
        return balance;
    }
}
```

```java
package leveltask2;

import java.util.Random;

    public class BankClient implements Runnable {
        private BankAccount account;

        public BankClient(BankAccount account) {
            this.account = account;
        }

        @Override
        public void run() {
            Random random = new Random();
            for (int i = 0; i < 10; i++) {
                if (random.nextBoolean()) {
                    account.deposit(random.nextInt(100));
                } else {
                    account.withdraw(random.nextInt(100));
                }
                try {
                    Thread.sleep(random.nextInt(500));
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        }
    }


package leveltask2;

public class Main {
    public static void main(String[] args) {
        BankAccount account = new BankAccount(1000);

        Thread client1 = new Thread(new BankClient(account), "Client 1");
        Thread client2 = new Thread(new BankClient(account), "Client 2");
        Thread client3 = new Thread(new BankClient(account), "Client 3");

        client1.start();
        client2.start();
        client3.start();

        try {
            client1.join();
            client2.join();
            client3.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }

        System.out.println("Final balance: " + account.getBalance());
    }
}
```

Output:


Run
Main
"C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe" "-javaagent:C:\Program Files\JetBra
Client 2 deposited 36.0, new balance: 1036.0
Client 1 deposited 94.0, new balance: 1130.0
Client 3 deposited 6.0, new balance: 1136.0
Client 2 deposited 70.0, new balance: 1206.0
Client 1 deposited 45.0, new balance: 1251.0
Client 1 withdrew 16.0, new balance: 1235.0
Client 2 deposited 47.0, new balance: 1282.0
Client 1 deposited 25.0, new balance: 1307.0
Client 3 deposited 63.0, new balance: 1370.0
Client 2 deposited 33.0, new balance: 1403.0
Client 1 deposited 97.0, new balance: 1500.0
Client 1 deposited 0.0, new balance: 1500.0
Client 1 withdrew 46.0, new balance: 1454.0
Client 2 deposited 52.0, new balance: 1506.0
Client 3 withdrew 56.0, new balance: 1450.0
Client 2 withdrew 14.0, new balance: 1436.0
Client 1 withdrew 47.0, new balance: 1389.0
Client 3 deposited 15.0, new balance: 1404.0
Client 3 withdrew 32.0, new balance: 1372.0
Client 2 deposited 16.0, new balance: 1388.0
Client 2 deposited 88.0, new balance: 1476.0
Client 2 withdrew 21.0, new balance: 1455.0
Client 1 deposited 95.0, new balance: 1550.0
Client 2 deposited 92.0, new balance: 1642.0
Client 3 withdrew 17.0, new balance: 1625.0
Client 1 deposited 6.0, new balance: 1631.0
Client 3 deposited 98.0, new balance: 1729.0
Client 3 deposited 75.0, new balance: 1804.0
Client 3 deposited 49.0, new balance: 1853.0
Client 3 deposited 82.0, new balance: 1935.0
Final balance: 1935.0
Process finished with exit code 0