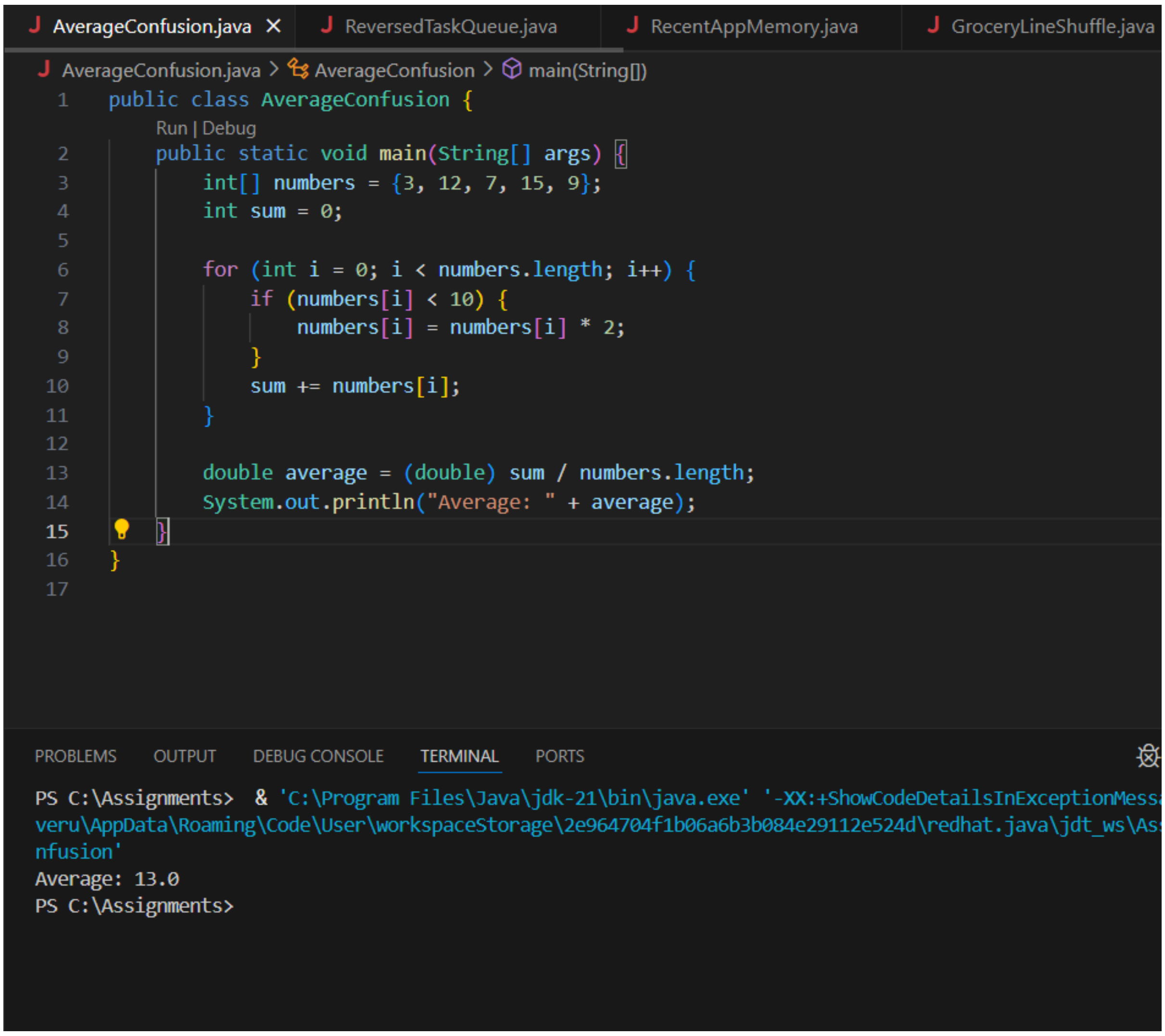
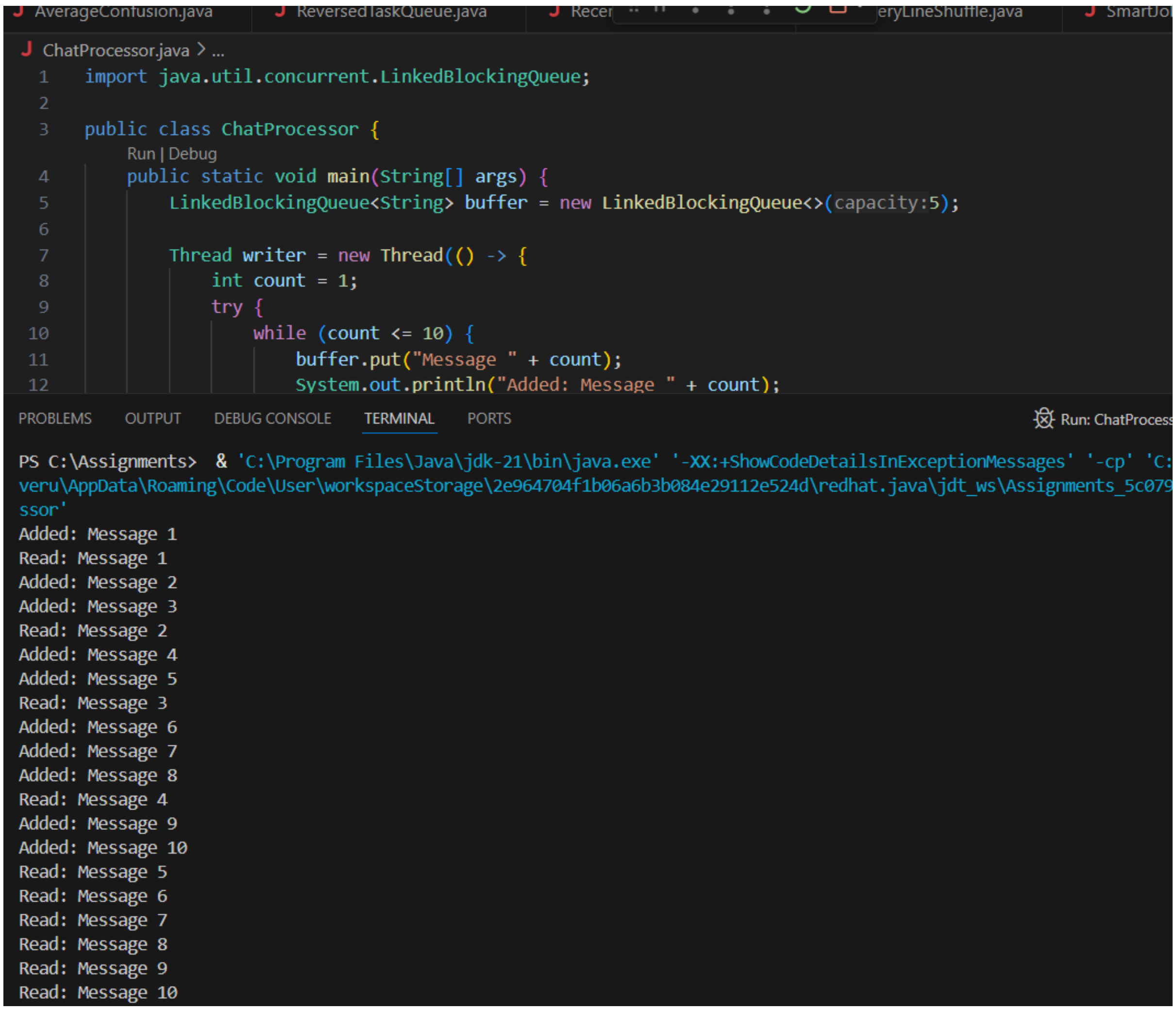


```java
public class AverageConfusion {
    public static void main(String[] args) {
        int[] numbers = {3, 12, 7, 15, 9};
        int sum = 0;

        for (int i = 0; i < numbers.length; i++) {
            if (numbers[i] < 10) {
                numbers[i] = numbers[i] * 2;
            }
            sum += numbers[i];
        }

        double average = (double) sum / numbers.length;
        System.out.println("Average: " + average);
    }
}
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMess
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\jdt_ws\As
nfusion'
Average: 13.0
PS C:\Assignments>
```

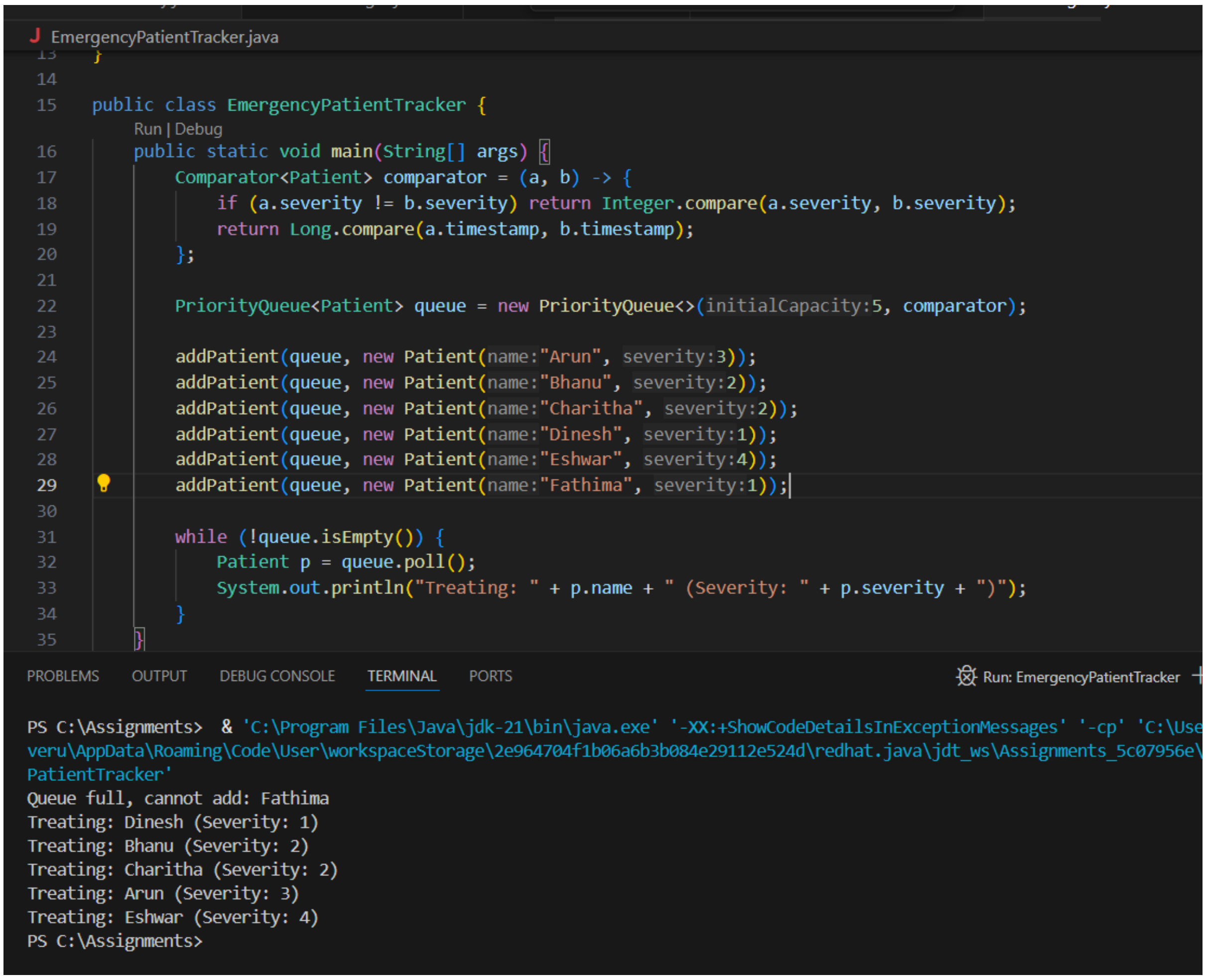
Figure 1AverageConfusion



```java
import java.util.concurrent.LinkedBlockingQueue;

public class ChatProcessor {
    public static void main(String[] args) {
        LinkedBlockingQueue<String> buffer = new LinkedBlockingQueue<>(capacity:5);

        Thread writer = new Thread(() -> {
            int count = 1;
            try {
                while (count <= 10) {
                    buffer.put("Message " + count);
                    System.out.println("Added: Message " + count);
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\jdt_ws\Assignments_5c079
ssor'
Added: Message 1
Read: Message 1
Added: Message 2
Added: Message 3
Read: Message 2
Added: Message 4
Added: Message 5
Read: Message 3
Added: Message 6
Added: Message 7
Added: Message 8
Read: Message 4
Added: Message 9
Added: Message 10
Read: Message 5
Read: Message 6
Read: Message 7
Read: Message 8
Read: Message 9
Read: Message 10
```

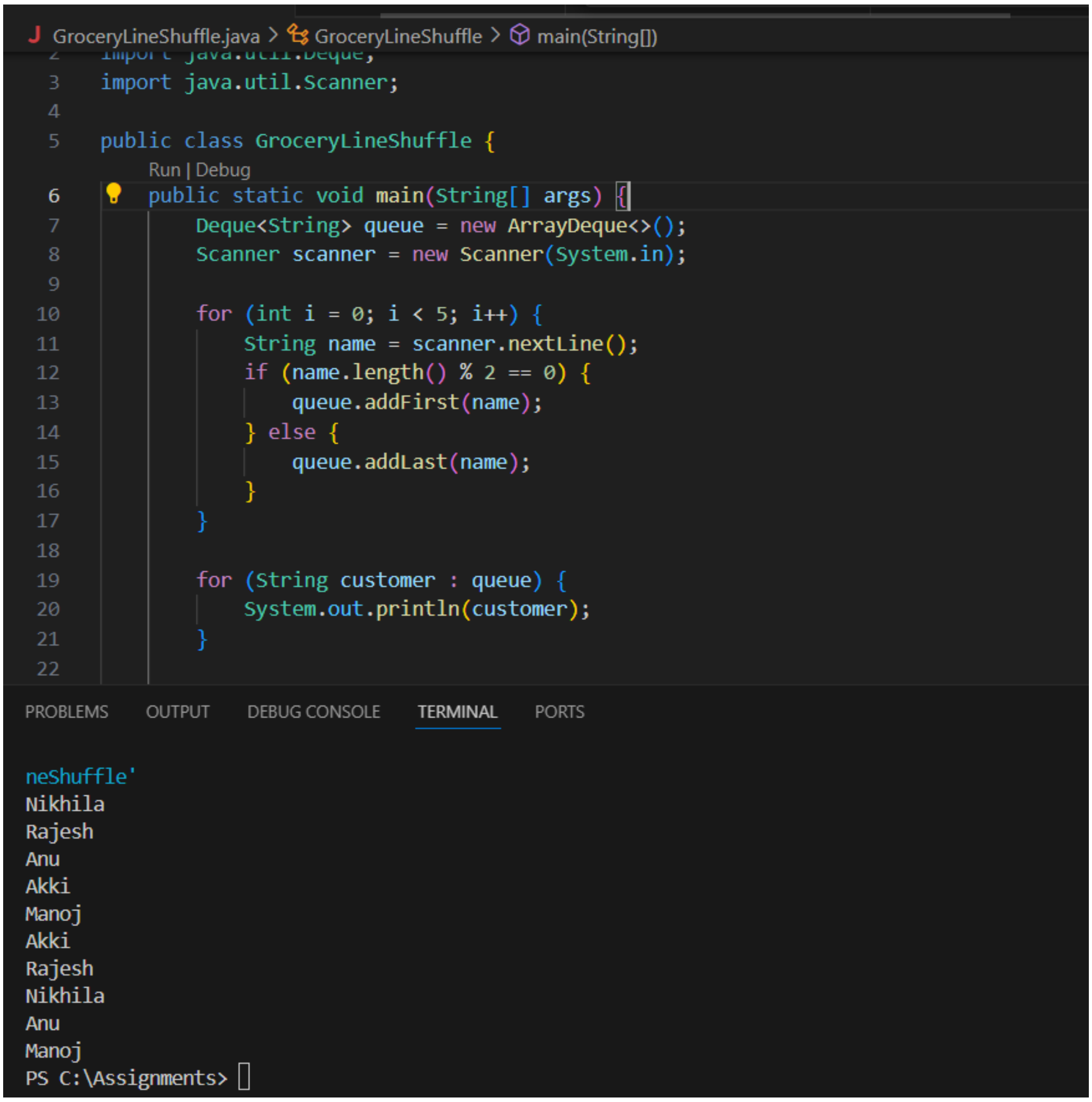
Figure 2ChatProcessor



```java
}

public class EmergencyPatientTracker {
    public static void main(String[] args) {
        Comparator<Patient> comparator = (a, b) -> {
            if (a.severity != b.severity) return Integer.compare(a.severity, b.severity);
            return Long.compare(a.timestamp, b.timestamp);
        };

        PriorityQueue<Patient> queue = new PriorityQueue<>(initialCapacity:5, comparator);

        addPatient(queue, new Patient(name:"Arun", severity:3));
        addPatient(queue, new Patient(name:"Bhanu", severity:2));
        addPatient(queue, new Patient(name:"Charitha", severity:2));
        addPatient(queue, new Patient(name:"Dinesh", severity:1));
        addPatient(queue, new Patient(name:"Eshwar", severity:4));
        addPatient(queue, new Patient(name:"Fathima", severity:1));

        while (!queue.isEmpty()) {
            Patient p = queue.poll();
            System.out.println("Treating: " + p.name + " (Severity: " + p.severity + ")");
        }
    }
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Use
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\jdt_ws\Assignments_5c07956e\
PatientTracker'
Queue full, cannot add: Fathima
Treating: Dinesh (Severity: 1)
Treating: Bhanu (Severity: 2)
Treating: Charitha (Severity: 2)
Treating: Arun (Severity: 3)
Treating: Eshwar (Severity: 4)
PS C:\Assignments>
```

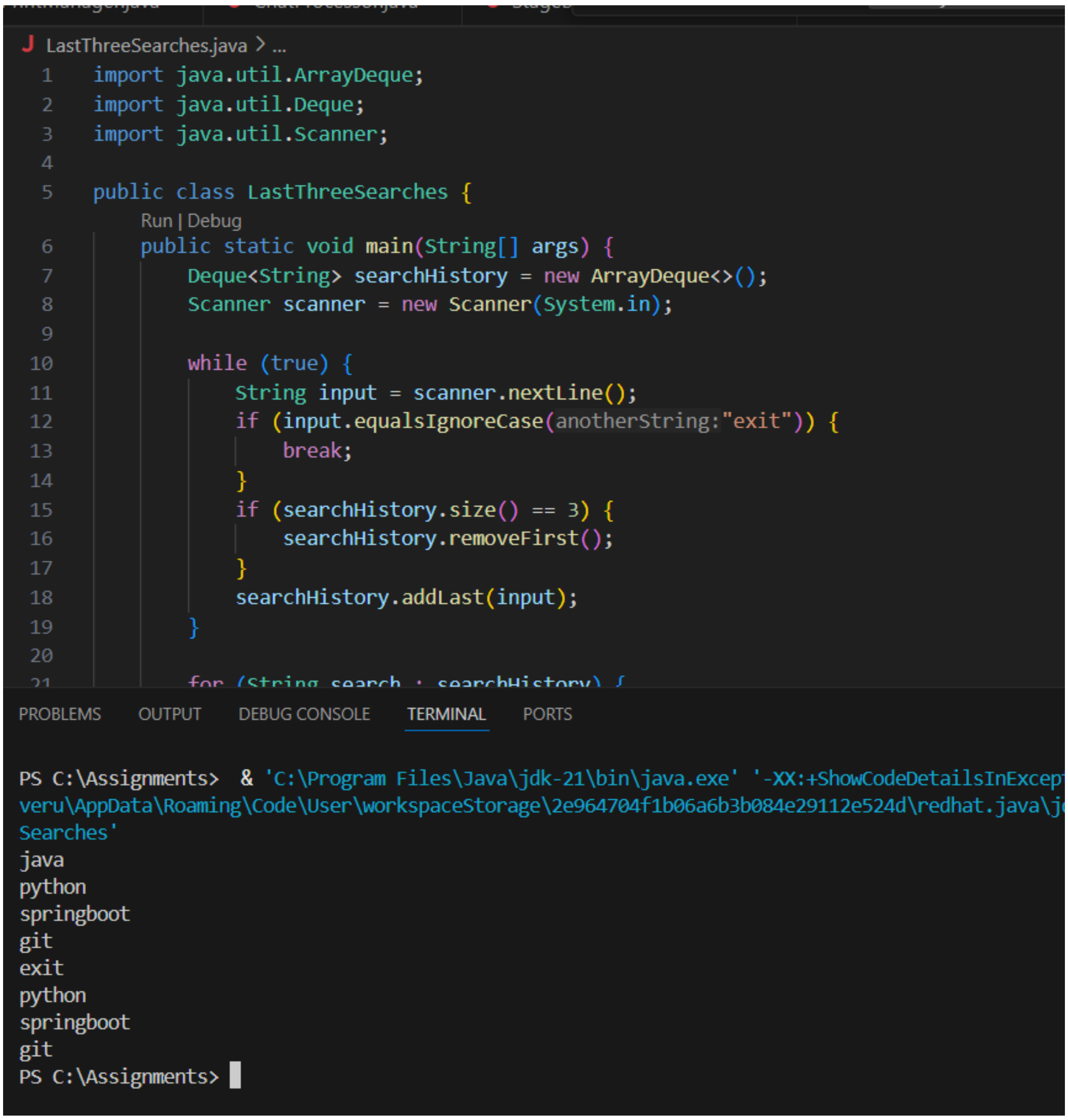
Figure 3EmergencyPatientTracker


GroceryLineShuffle.java > GroceryLineShuffle > main(String[])
Run | Debug
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS

```java
import java.util.Deque;
import java.util.Scanner;

public class GroceryLineShuffle {
    public static void main(String[] args) {
        Deque<String> queue = new ArrayDeque<>();
        Scanner scanner = new Scanner(System.in);

        for (int i = 0; i < 5; i++) {
            String name = scanner.nextLine();
            if (name.length() % 2 == 0) {
                queue.addFirst(name);
            } else {
                queue.addLast(name);
            }
        }

        for (String customer : queue) {
            System.out.println(customer);
        }

```

```
neShuffle'
Nikhila
Rajesh
Anu
Akki
Manoj
Akki
Rajesh
Nikhila
Anu
Manoj
PS C:\Assignments>
```

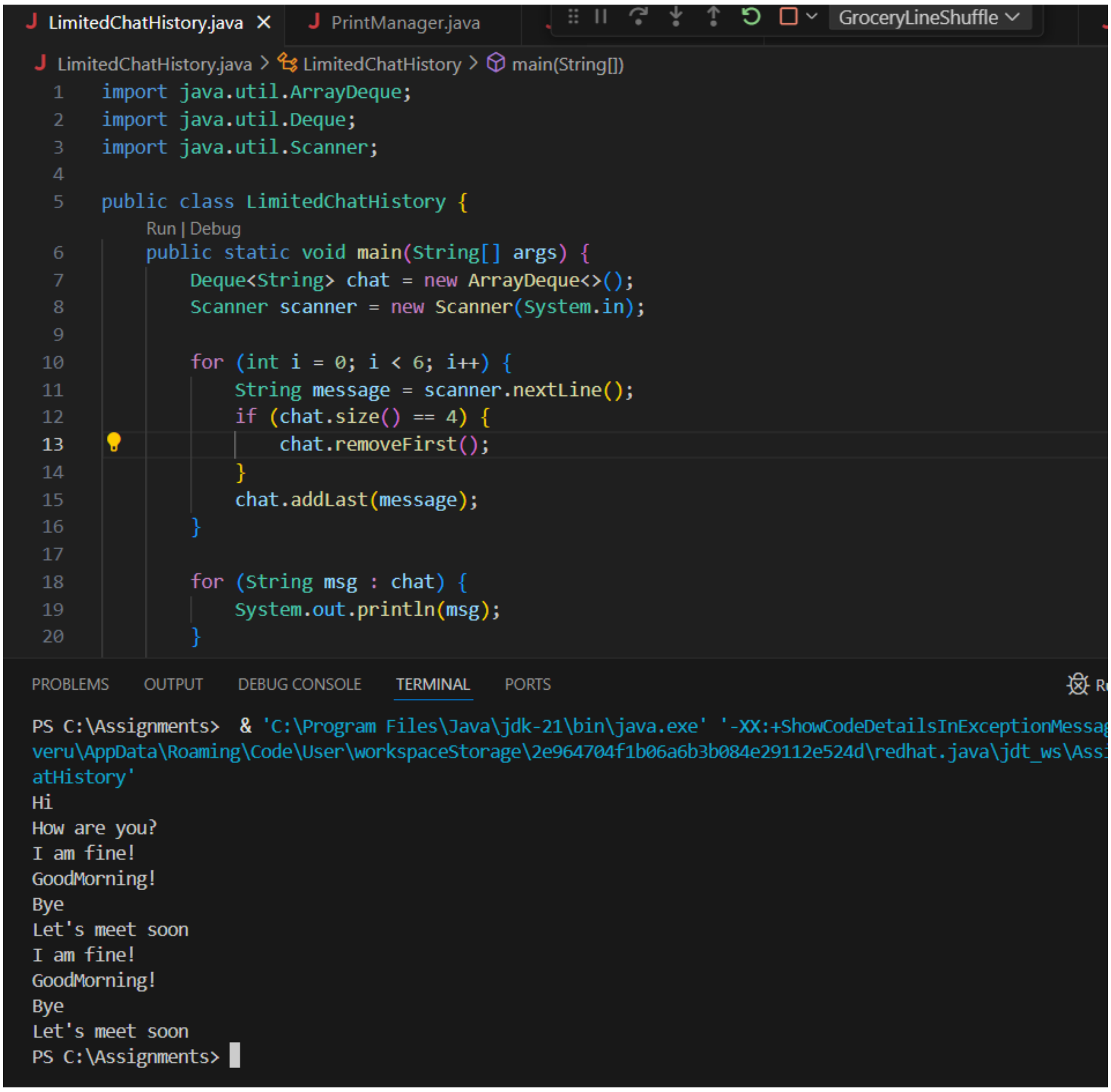
Figure 4GroceryLineShuffle



```java
import java.util.ArrayDeque;
import java.util.Deque;
import java.util.Scanner;

public class LastThreeSearches {
    public static void main(String[] args) {
        Deque<String> searchHistory = new ArrayDeque<>();
        Scanner scanner = new Scanner(System.in);

        while (true) {
            String input = scanner.nextLine();
            if (input.equalsIgnoreCase(anotherString:"exit")) {
                break;
            }
            if (searchHistory.size() == 3) {
                searchHistory.removeFirst();
            }
            searchHistory.addLast(input);
        }

        for (String search : searchHistory) {
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExcep
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\j
Searches'
java
python
springboot
git
exit
python
springboot
git
PS C:\Assignments> 
```

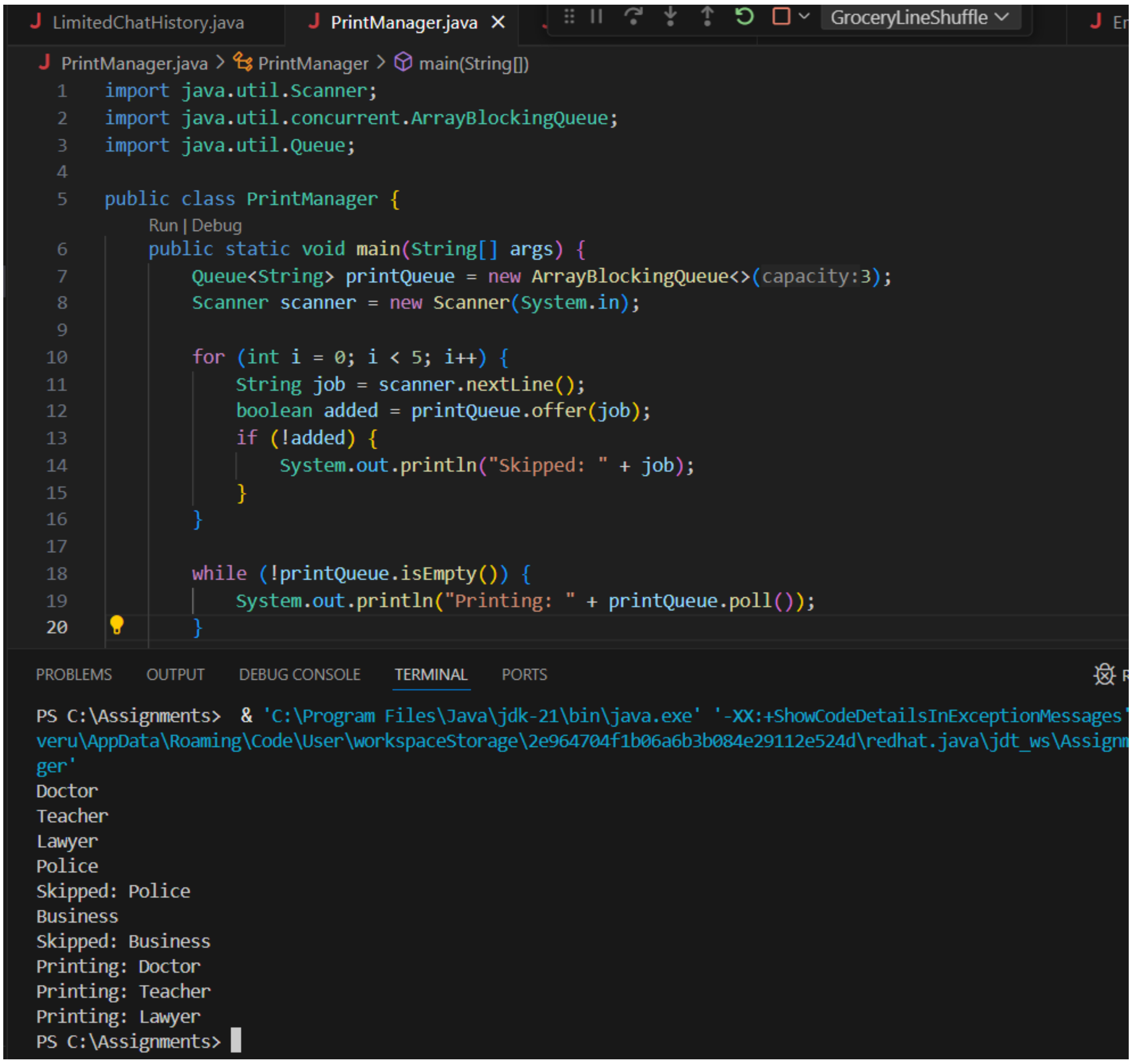
Figure 5LastThreeSearches



```java
import java.util.ArrayDeque;
import java.util.Deque;
import java.util.Scanner;

public class LimitedChatHistory {
    public static void main(String[] args) {
        Deque<String> chat = new ArrayDeque<>();
        Scanner scanner = new Scanner(System.in);

        for (int i = 0; i < 6; i++) {
            String message = scanner.nextLine();
            if (chat.size() == 4) {
                chat.removeFirst();
            }
            chat.addLast(message);
        }

        for (String msg : chat) {
            System.out.println(msg);
        }
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessag
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\jdt_ws\Assi
atHistory'
Hi
How are you?
I am fine!
GoodMorning!
Bye
Let's meet soon
I am fine!
GoodMorning!
Bye
Let's meet soon
PS C:\Assignments> 
```

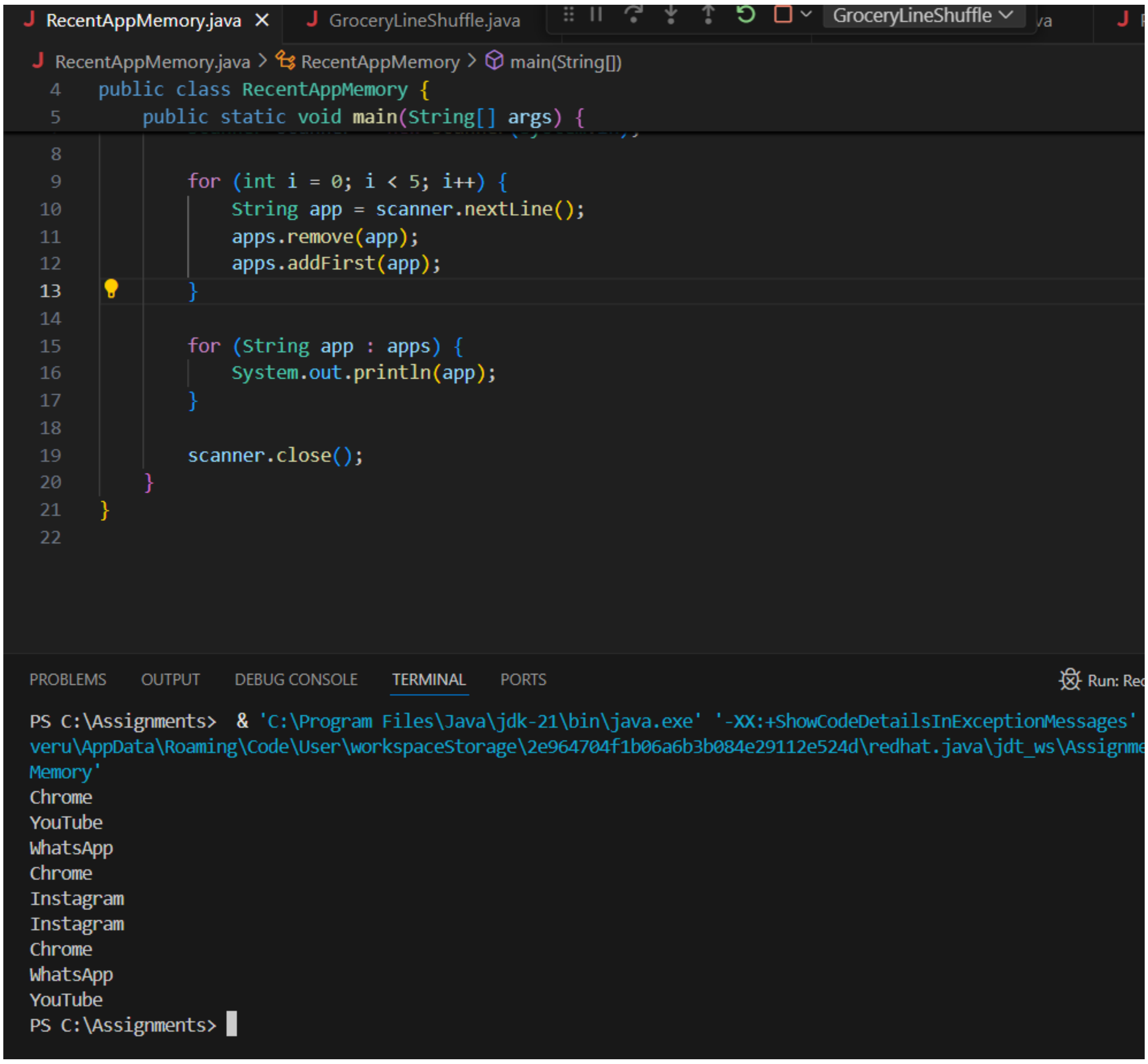
Figure 6LimitedChatHistory



```java
import java.util.Scanner;
import java.util.concurrent.ArrayBlockingQueue;
import java.util.Queue;

public class PrintManager {
    Run | Debug
    public static void main(String[] args) {
        Queue<String> printQueue = new ArrayBlockingQueue<>(capacity:3);
        Scanner scanner = new Scanner(System.in);

        for (int i = 0; i < 5; i++) {
            String job = scanner.nextLine();
            boolean added = printQueue.offer(job);
            if (!added) {
                System.out.println("Skipped: " + job);
            }
        }

        while (!printQueue.isEmpty()) {
            System.out.println("Printing: " + printQueue.poll());
        }
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages'
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\jdt_ws\Assignm
ger'
Doctor
Teacher
Lawyer
Police
Skipped: Police
Business
Skipped: Business
Printing: Doctor
Printing: Teacher
Printing: Lawyer
PS C:\Assignments>
```

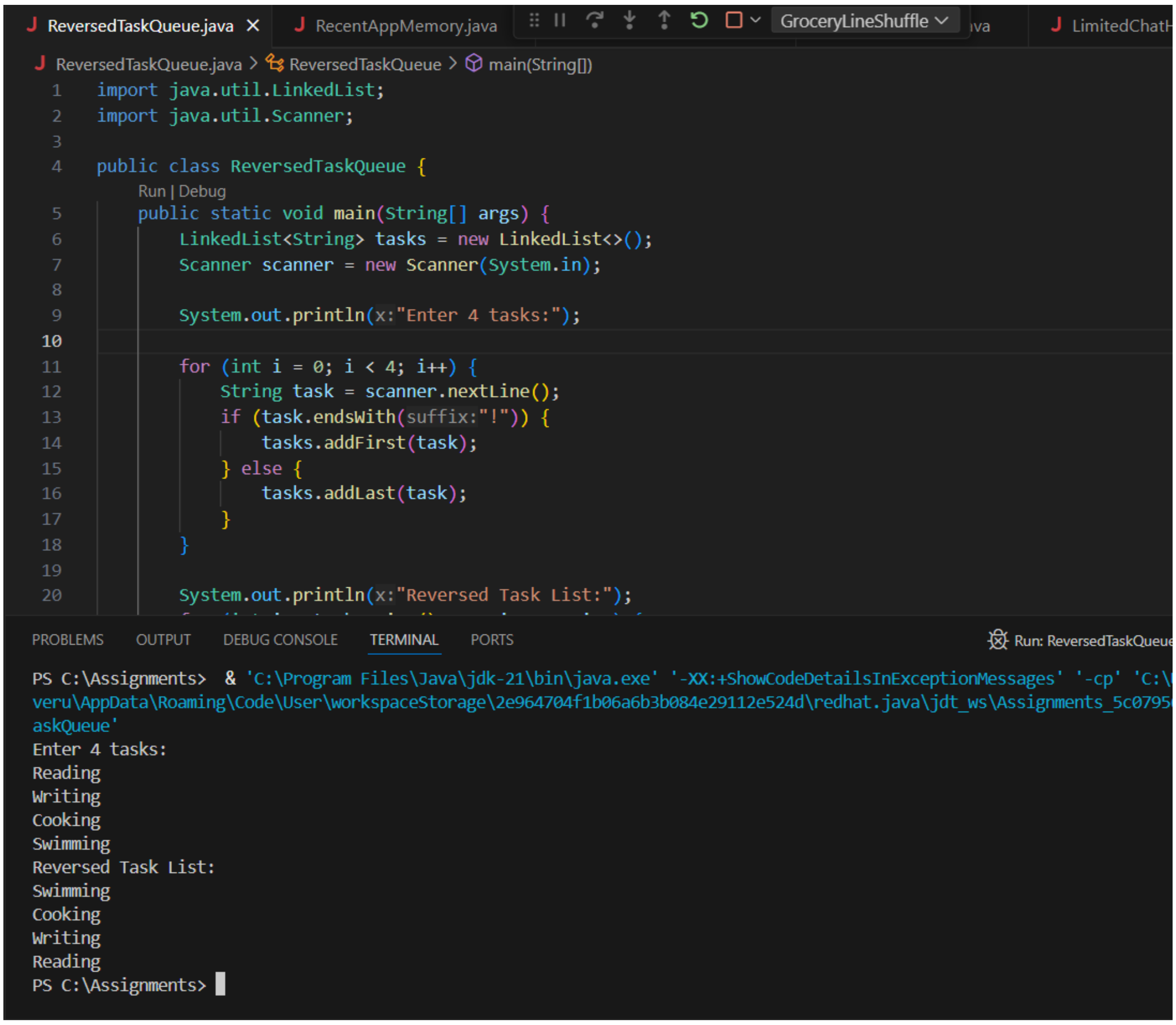
Figure 7PrintManager



```java
public class RecentAppMemory {
    public static void main(String[] args) {

        for (int i = 0; i < 5; i++) {
            String app = scanner.nextLine();
            apps.remove(app);
            apps.addFirst(app);
        }

        for (String app : apps) {
            System.out.println(app);
        }

        scanner.close();
    }
}
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages'
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\jdt_ws\Assignme
Memory'
Chrome
YouTube
WhatsApp
Chrome
Instagram
Instagram
Chrome
WhatsApp
YouTube
PS C:\Assignments>
```

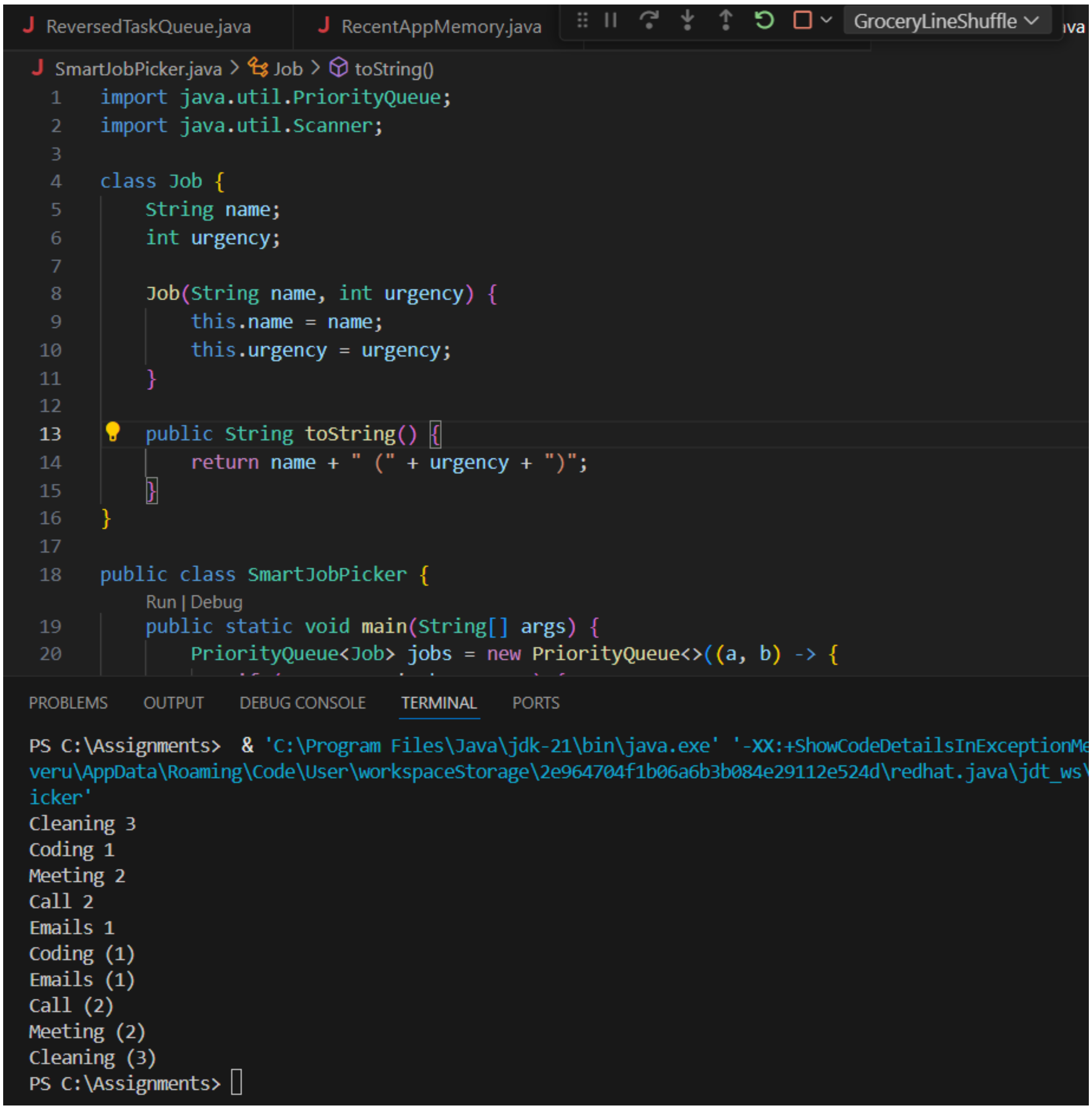
Figure 8RecentAppMemory



```java
import java.util.LinkedList;
import java.util.Scanner;

public class ReversedTaskQueue {
    public static void main(String[] args) {
        LinkedList<String> tasks = new LinkedList<>();
        Scanner scanner = new Scanner(System.in);

        System.out.println(x:"Enter 4 tasks:");

        for (int i = 0; i < 4; i++) {
            String task = scanner.nextLine();
            if (task.endsWith(suffix:"!")) {
                tasks.addFirst(task);
            } else {
                tasks.addLast(task);
            }
        }

        System.out.println(x:"Reversed Task List:");
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\jdt_ws\Assignments_5c0795
askQueue'
Enter 4 tasks:
Reading
Writing
Cooking
Swimming
Reversed Task List:
Swimming
Cooking
Writing
Reading
PS C:\Assignments> 
```

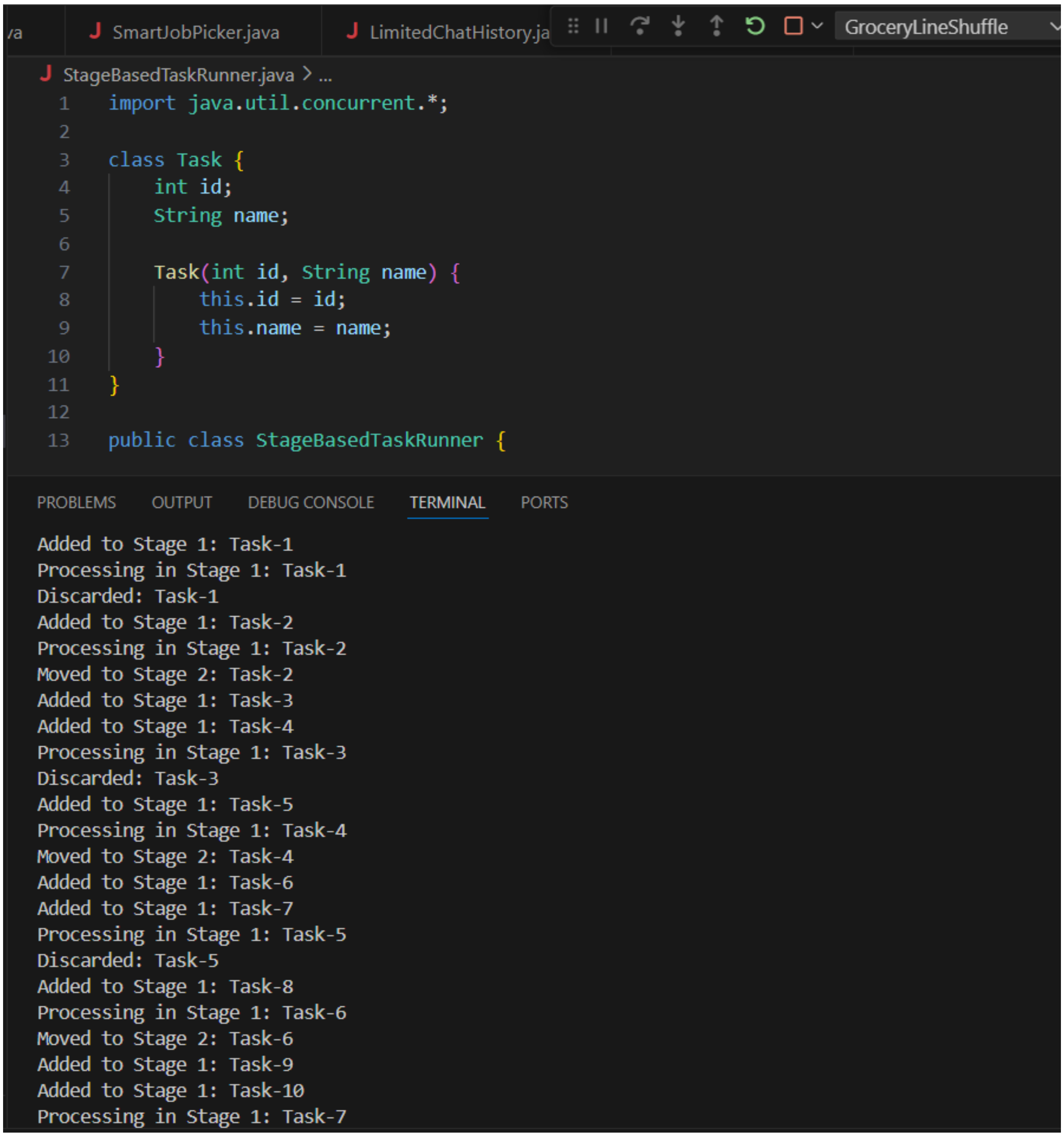
Figure 9ReversedTaskQueue



```java
import java.util.PriorityQueue;
import java.util.Scanner;

class Job {
    String name;
    int urgency;

    Job(String name, int urgency) {
        this.name = name;
        this.urgency = urgency;
    }

    public String toString() {
        return name + " (" + urgency + ")";
    }
}

public class SmartJobPicker {
    public static void main(String[] args) {
        PriorityQueue<Job> jobs = new PriorityQueue<>((a, b) -> {
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMe
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.java\jdt_ws\
icker'
Cleaning 3
Coding 1
Meeting 2
Call 2
Emails 1
Coding (1)
Emails (1)
Call (2)
Meeting (2)
Cleaning (3)
PS C:\Assignments> 
```

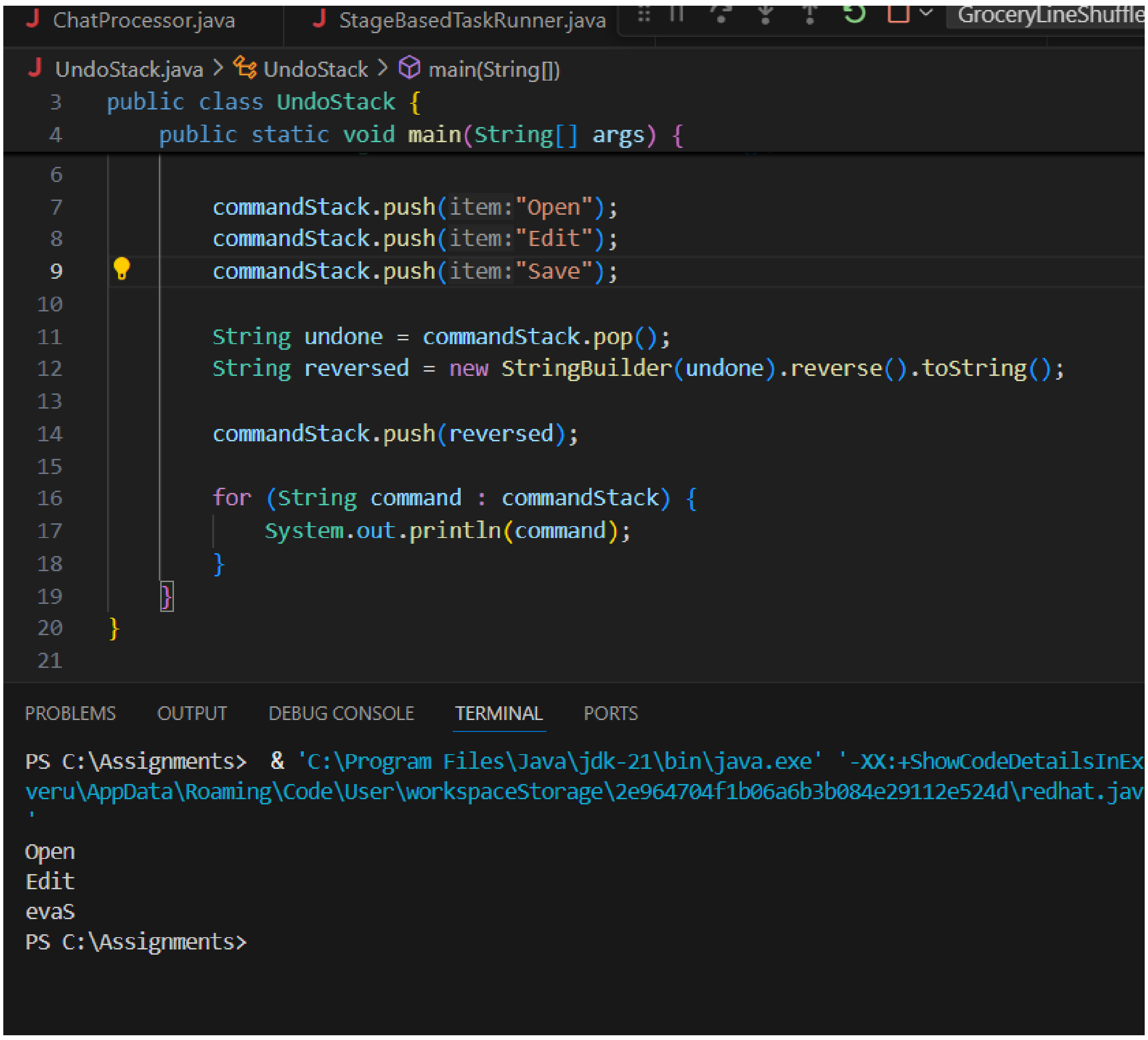
Figure 10SmartJobPicker



```java
import java.util.concurrent.*;

class Task {
    int id;
    String name;

    Task(int id, String name) {
        this.id = id;
        this.name = name;
    }
}

public class StageBasedTaskRunner {
```

```
Added to Stage 1: Task-1
Processing in Stage 1: Task-1
Discarded: Task-1
Added to Stage 1: Task-2
Processing in Stage 1: Task-2
Moved to Stage 2: Task-2
Added to Stage 1: Task-3
Added to Stage 1: Task-4
Processing in Stage 1: Task-3
Discarded: Task-3
Added to Stage 1: Task-5
Processing in Stage 1: Task-4
Moved to Stage 2: Task-4
Added to Stage 1: Task-6
Added to Stage 1: Task-7
Processing in Stage 1: Task-5
Discarded: Task-5
Added to Stage 1: Task-8
Processing in Stage 1: Task-6
Moved to Stage 2: Task-6
Added to Stage 1: Task-9
Added to Stage 1: Task-10
Processing in Stage 1: Task-7
```

Figure 11StageBasedTaskRunner



```java
public class UndoStack {
    public static void main(String[] args) {

        commandStack.push(item:"Open");
        commandStack.push(item:"Edit");
        commandStack.push(item:"Save");

        String undone = commandStack.pop();
        String reversed = new StringBuilder(undone).reverse().toString();

        commandStack.push(reversed);

        for (String command : commandStack) {
            System.out.println(command);
        }
    }
}
```

```
PS C:\Assignments>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInEx
veru\AppData\Roaming\Code\User\workspaceStorage\2e964704f1b06a6b3b084e29112e524d\redhat.jav
'
Open
Edit
evaS
PS C:\Assignments>
```

Figure 12UndoStack