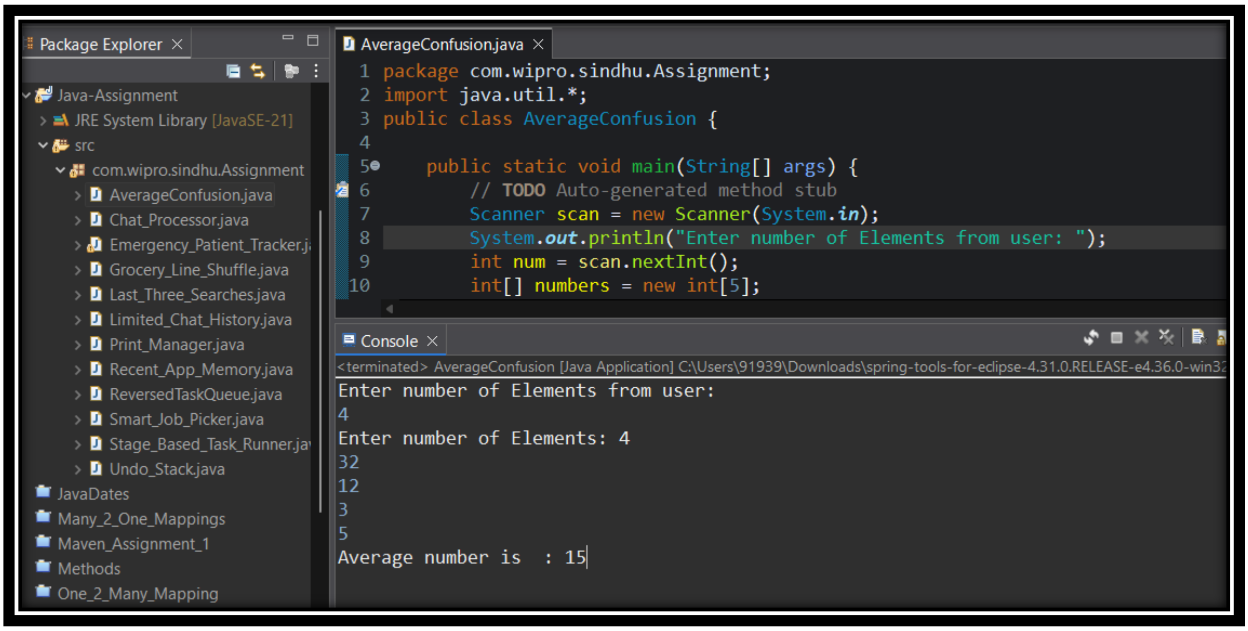
Java_Assignment_Outputs

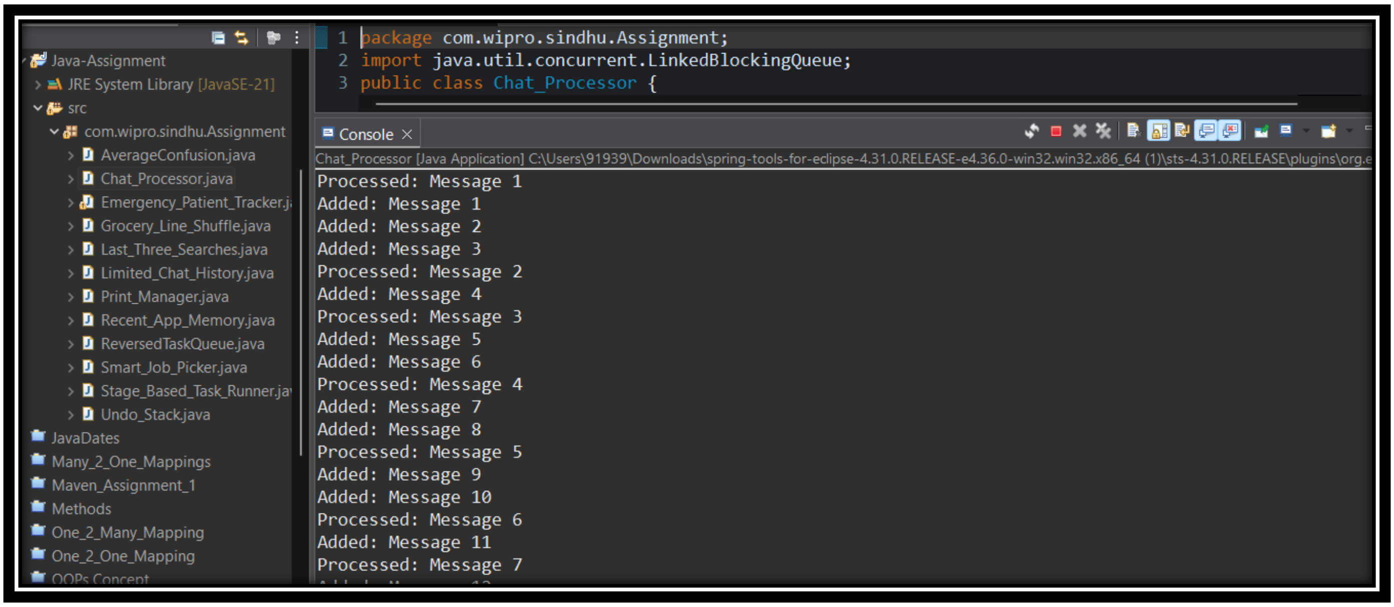
<u>AverageConfusion</u> :-This program reads a list of numbers from the user, applies a transformation to numbers less than 10, then calculates and displays the average of the modified numbers.



```java
package com.wipro.sindhu.Assignment;
import java.util.*;
public class AverageConfusion {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		Scanner scan = new Scanner(System.in);
		System.out.println("Enter number of Elements from user: ");
		int num = scan.nextInt();
		int[] numbers = new int[5];
```

```
<terminated> AverageConfusion [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32
Enter number of Elements from user:
4
Enter number of Elements: 4
32
12
3
5
Average number is  : 15
```

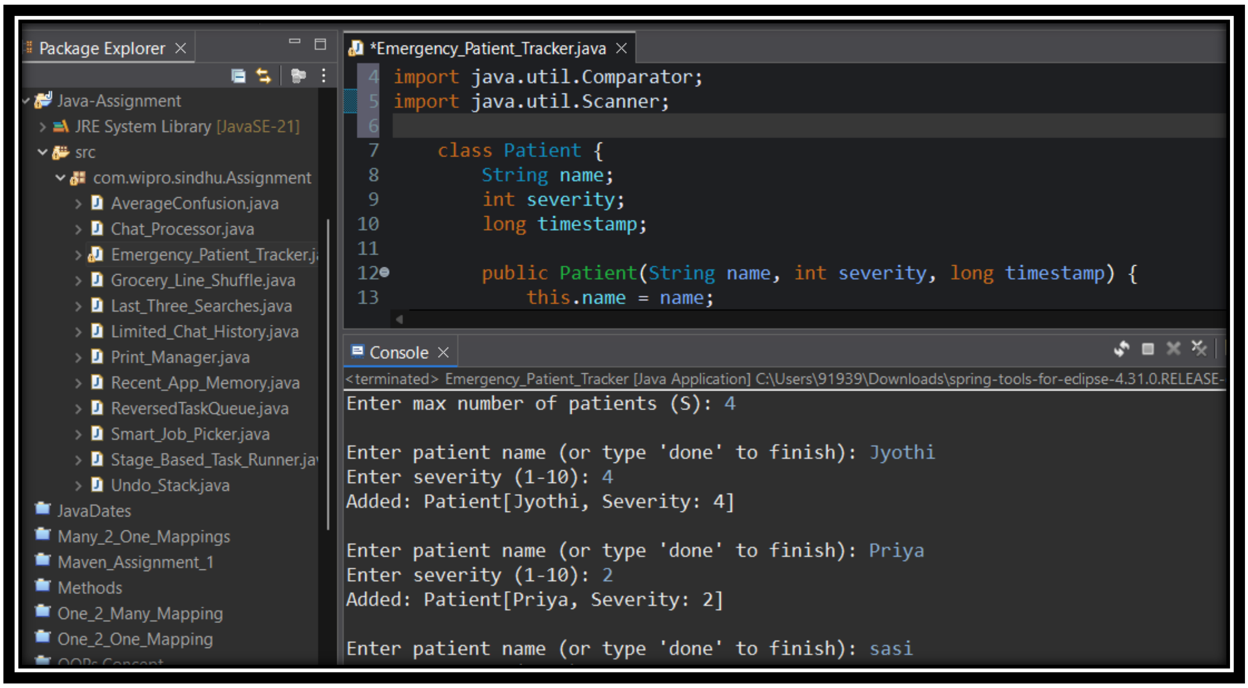
<u>Chat_Processor</u> :- It is a simple chat message processing system that means a Producer thread generates messages and adds them to a queue, a Consumer thread takes messages from the queue and processes them.



```java
package com.wipro.sindhu.Assignment;
import java.util.concurrent.LinkedBlockingQueue;
public class Chat_Processor {
```

```
Chat_Processor [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win32.x86_64 (1)\sts-4.31.0.RELEASE\plugins\org.e
Processed: Message 1
Added: Message 1
Added: Message 2
Added: Message 3
Processed: Message 2
Added: Message 4
Processed: Message 3
Added: Message 5
Added: Message 6
Processed: Message 4
Added: Message 7
Added: Message 8
Processed: Message 5
Added: Message 9
Added: Message 10
Processed: Message 6
Added: Message 11
Processed: Message 7
```

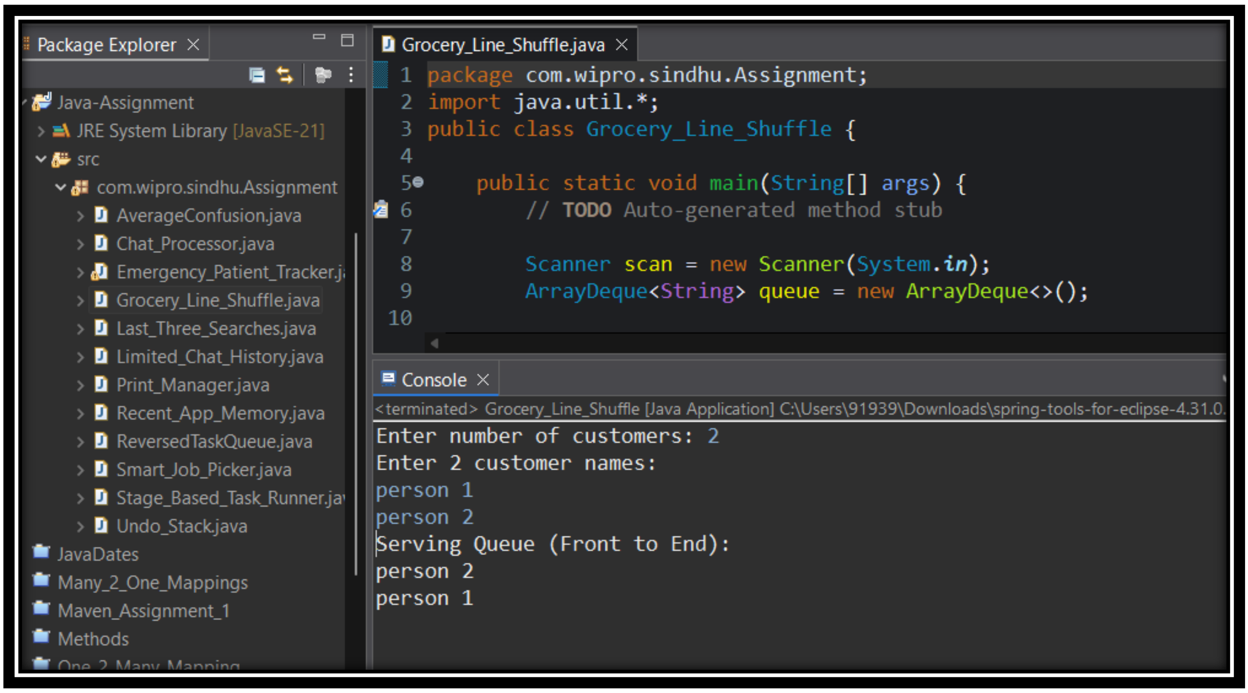
Emergency_Patient_Tracker :- Keeps track of emergency patients based on severity and arrival order, and treats them in the correct priority.



```
import java.util.Comparator;
import java.util.Scanner;

class Patient {
    String name;
    int severity;
    long timestamp;

    public Patient(String name, int severity, long timestamp) {
        this.name = name;
```

```
<terminated> Emergency_Patient_Tracker [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-
Enter max number of patients (S): 4

Enter patient name (or type 'done' to finish): Jyothi
Enter severity (1-10): 4
Added: Patient[Jyothi, Severity: 4]

Enter patient name (or type 'done' to finish): Priya
Enter severity (1-10): 2
Added: Patient[Priya, Severity: 2]

Enter patient name (or type 'done' to finish): sasi
```

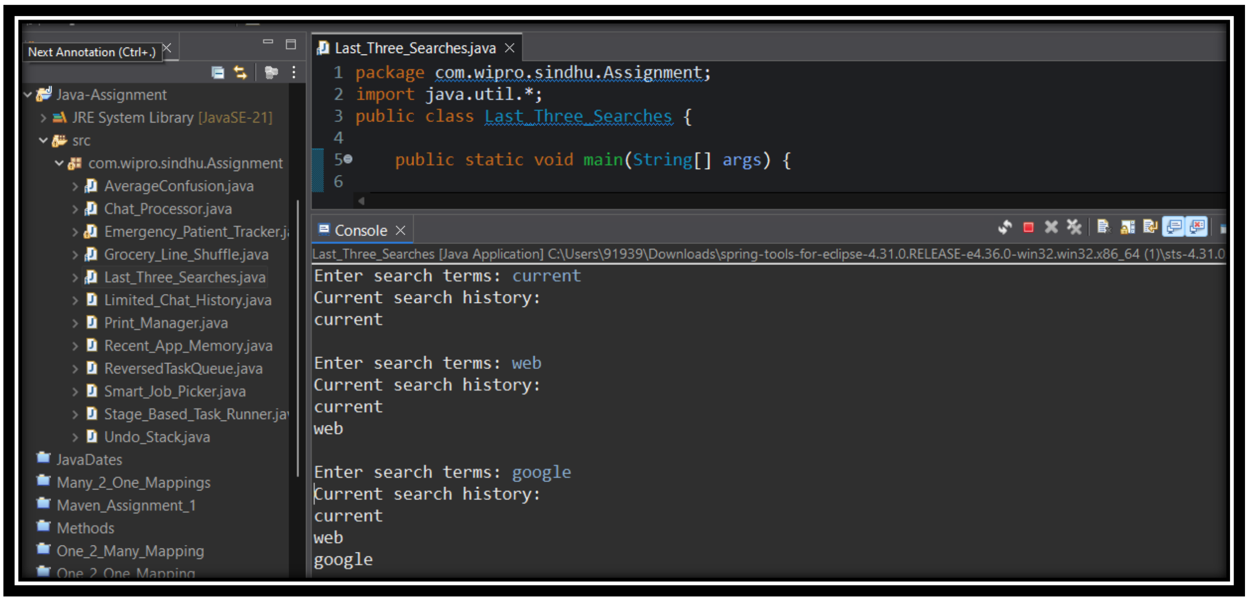
Grocery_Line_Shuffle :- A grocery store queue where customers are arranged differently based on whether their name length is even or odd.



```
package com.wipro.sindhu.Assignment;
import java.util.*;
public class Grocery_Line_Shuffle {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        Scanner scan = new Scanner(System.in);
        ArrayDeque<String> queue = new ArrayDeque<>();

```

```
<terminated> Grocery_Line_Shuffle [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.
Enter number of customers: 2
Enter 2 customer names:
person 1
person 2
Serving Queue (Front to End):
person 2
person 1
```

Last_Three_Searches :- It Keeps track of the last three search terms entered by the user and displays them after each entry.

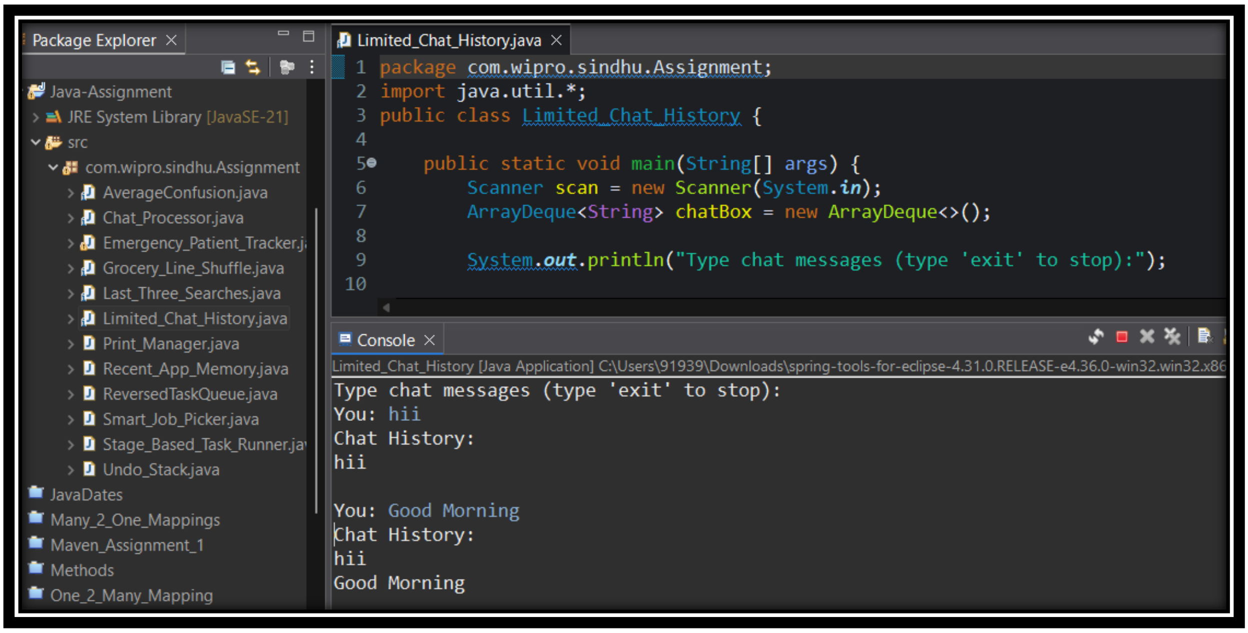


```
package com.wipro.sindhu.Assignment;
import java.util.*;
public class Last_Three_Searches {

    public static void main(String[] args) {

```

Last_Three_Searches [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win32.x86_64 (1)\sts-4.31.0

```
Enter search terms: current
Current search history:
current

Enter search terms: web
Current search history:
current
web

Enter search terms: google
Current search history:
current
web
google
```

Limited_Chat_History :- It Stores and displays only the last 4 chat messages entered by the user.

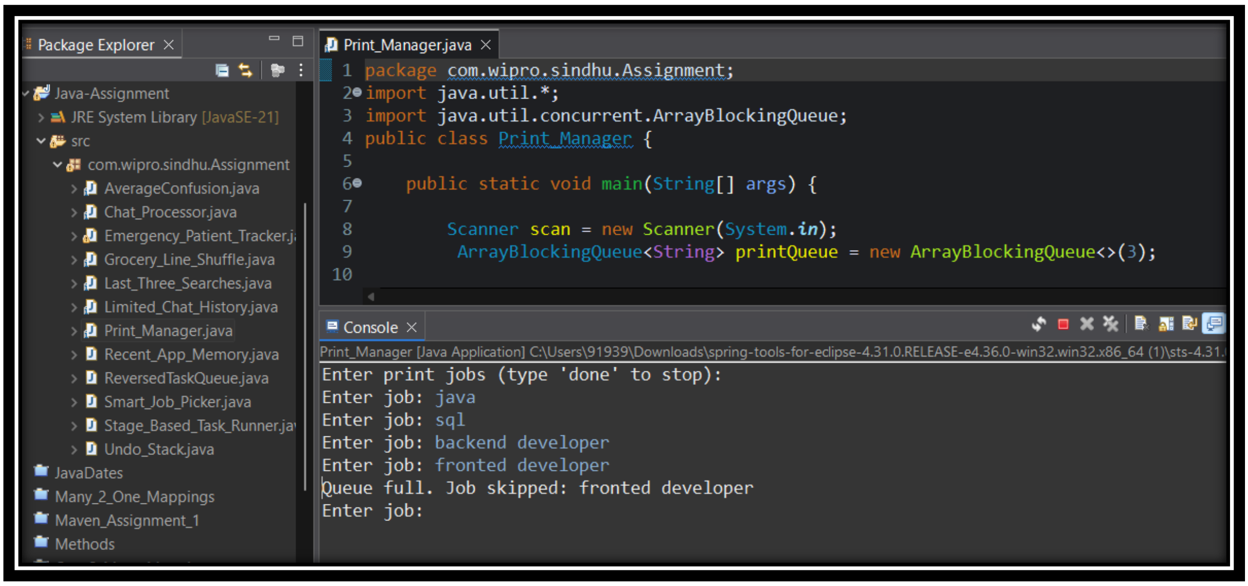


```
package com.wipro.sindhu.Assignment;
import java.util.*;
public class Limited_Chat_History {

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        ArrayDeque<String> chatBox = new ArrayDeque<>();

        System.out.println("Type chat messages (type 'exit' to stop):");

```

Limited_Chat_History [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win32.x86

```
Type chat messages (type 'exit' to stop):
You: hii
Chat History:
hii

You: Good Morning
Chat History:
hii
Good Morning
```

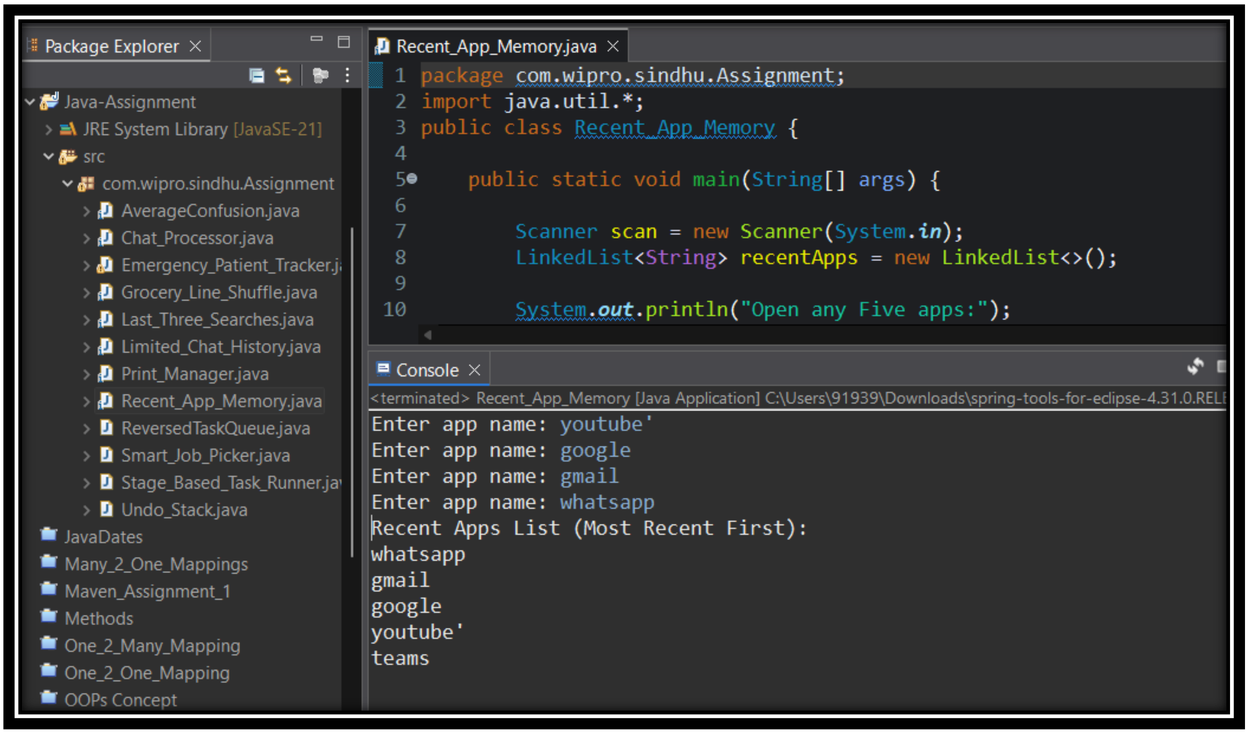
Print_Manager :- A printer job manager where print jobs are stored in a fixed-size queue.



```java
package com.wipro.sindhu.Assignment;
import java.util.*;
import java.util.concurrent.ArrayBlockingQueue;
public class Print_Manager {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
         ArrayBlockingQueue<String> printQueue = new ArrayBlockingQueue<>(3);

```

```
Print_Manager [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win32.x86_64 (1)\sts-4.31.
Enter print jobs (type 'done' to stop):
Enter job: java
Enter job: sql
Enter job: backend developer
Enter job: fronted developer
Queue full. Job skipped: fronted developer
Enter job:
```

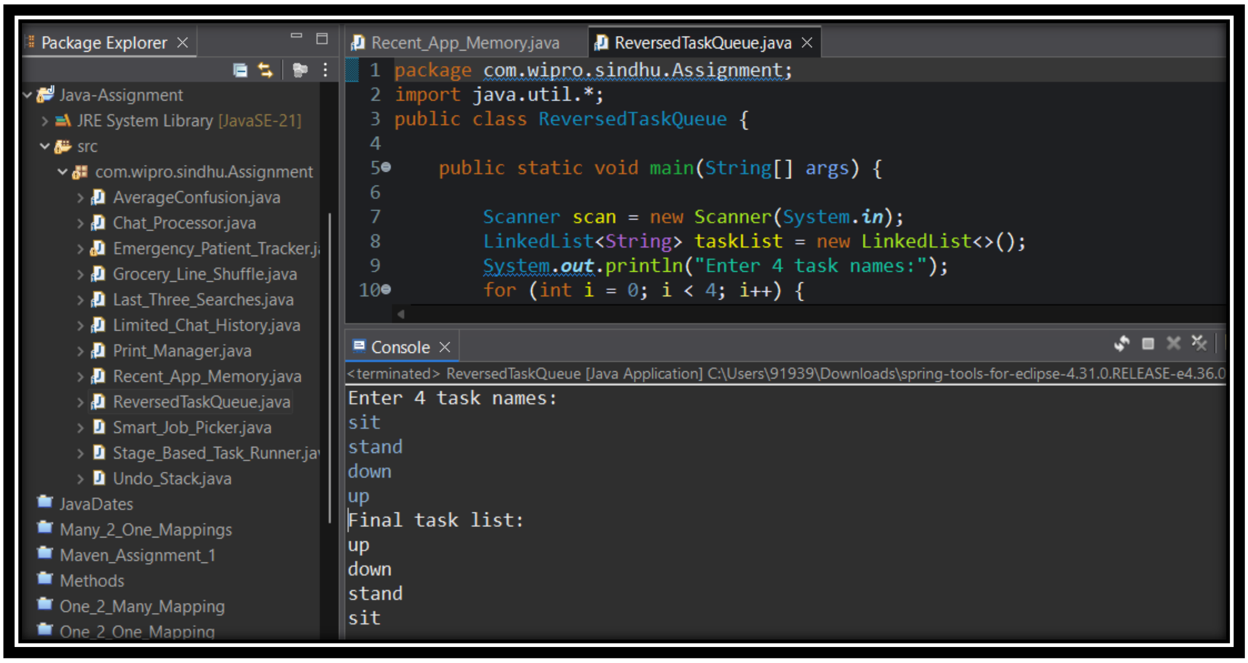
Recent_App_Memory :- The Recent Apps feature in mobile operating systems, where the most recently opened app appears at the top and duplicates are removed.



```java
package com.wipro.sindhu.Assignment;
import java.util.*;
public class Recent_App_Memory {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
        LinkedList<String> recentApps = new LinkedList<>();

        System.out.println("Open any Five apps:");
```

```
<terminated> Recent_App_Memory [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELE
Enter app name: youtube'
Enter app name: google
Enter app name: gmail
Enter app name: whatsapp
Recent Apps List (Most Recent First):
whatsapp
gmail
google
youtube'
teams
```

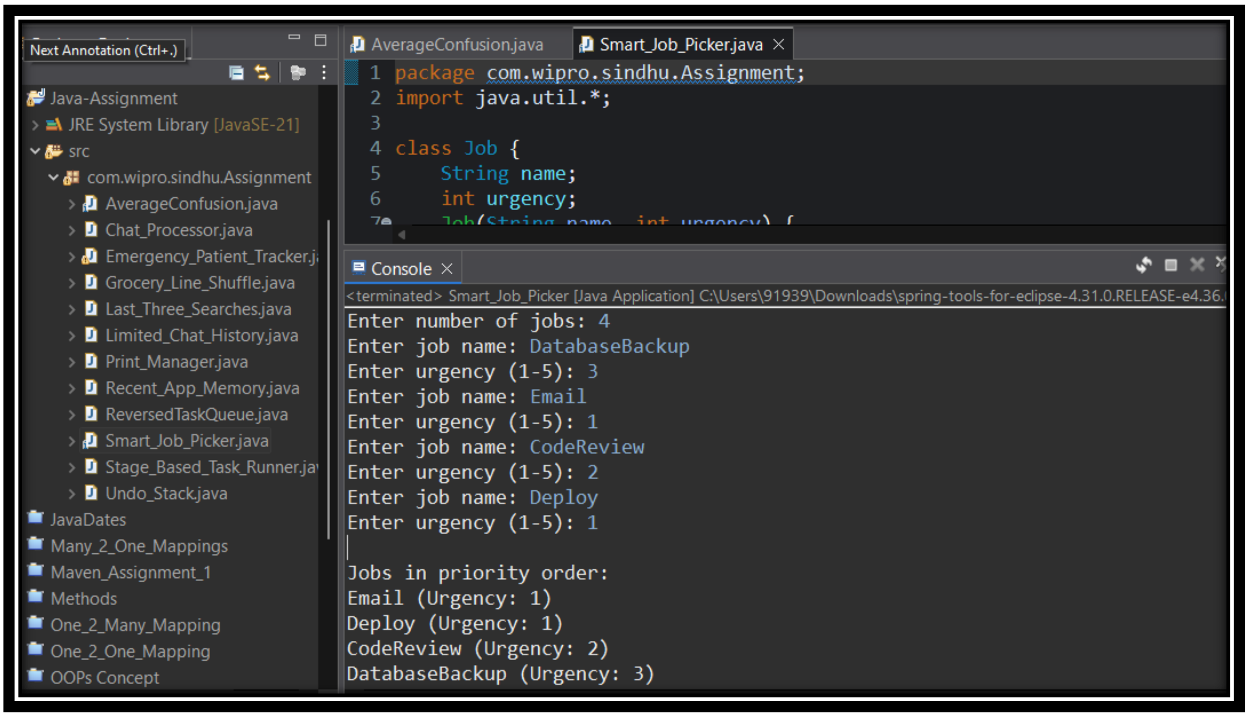
ReversedTaskQueue :- It Takes 4 task names from the user and stores them in a linked list, placing each task at the front or back depending on its ending.



```
package com.wipro.sindhu.Assignment;
import java.util.*;
public class ReversedTaskQueue {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
        LinkedList<String> taskList = new LinkedList<>();
        System.out.println("Enter 4 task names:");
        for (int i = 0; i < 4; i++) {
```

```
<terminated> ReversedTaskQueue [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0
Enter 4 task names:
sit
stand
down
up
Final task list:
up
down
stand
sit
```

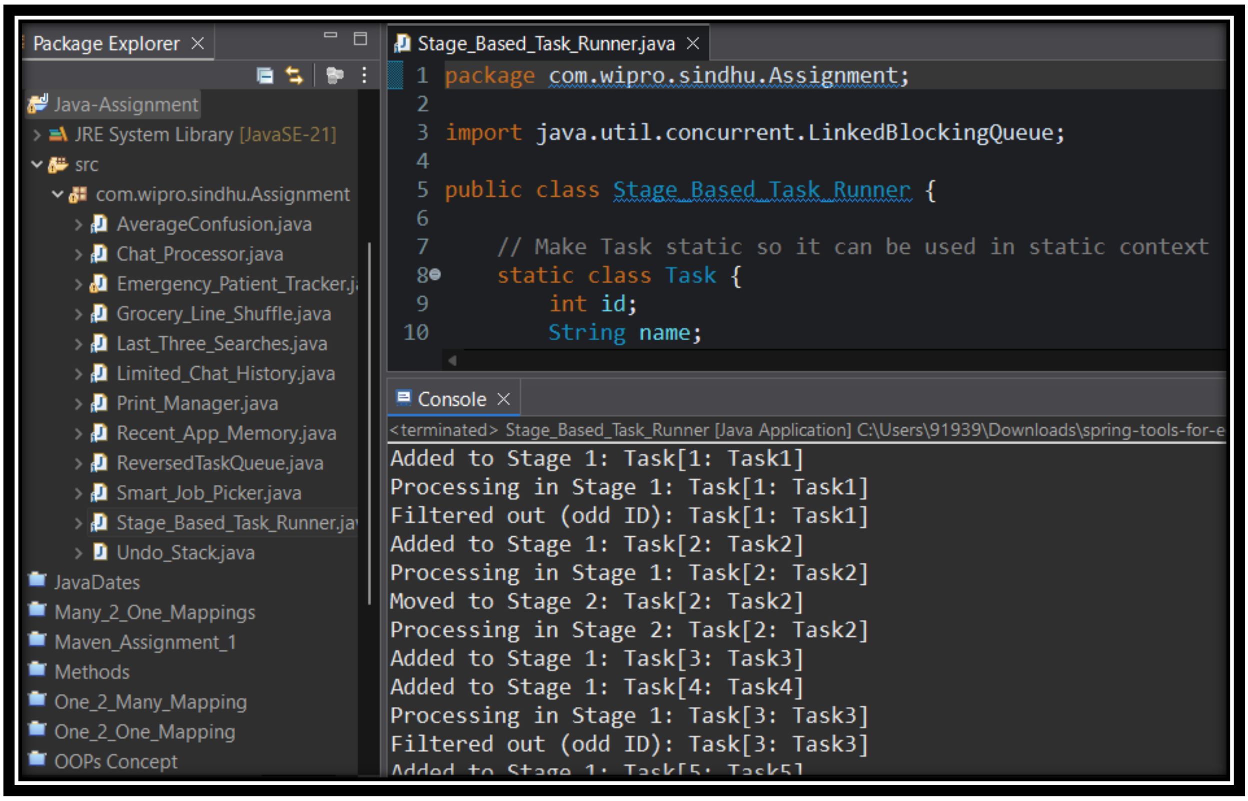
Smart_Job_Picker :- Takes a list of jobs from the user, each with a name and urgency level, then prints them in priority order.



```
package com.wipro.sindhu.Assignment;
import java.util.*;

class Job {
    String name;
    int urgency;
    Job(String name, int urgency) {
```

```
<terminated> Smart_Job_Picker [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.
Enter number of jobs: 4
Enter job name: DatabaseBackup
Enter urgency (1-5): 3
Enter job name: Email
Enter urgency (1-5): 1
Enter job name: CodeReview
Enter urgency (1-5): 2
Enter job name: Deploy
Enter urgency (1-5): 1

Jobs in priority order:
Email (Urgency: 1)
Deploy (Urgency: 1)
CodeReview (Urgency: 2)
DatabaseBackup (Urgency: 3)
```

Stage_Based_Task_Runner :- Tasks are produced. Certain tasks are filtered and moved to another queue.

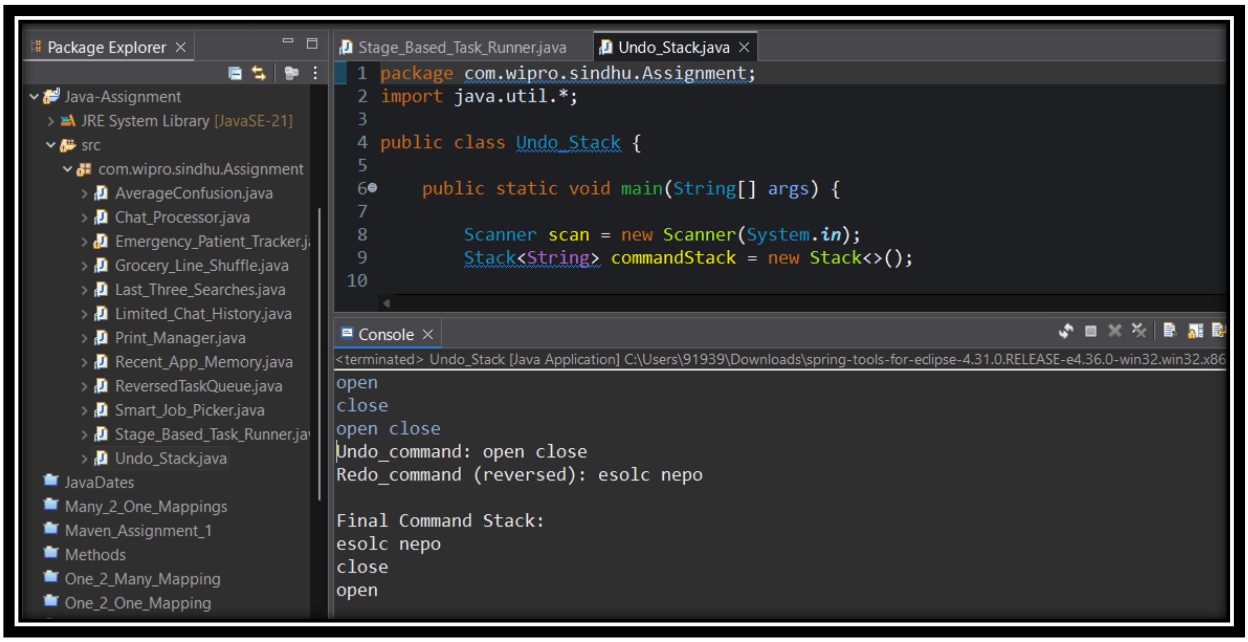


```java
package com.wipro.sindhu.Assignment;

import java.util.concurrent.LinkedBlockingQueue;

public class Stage_Based_Task_Runner {

    // Make Task static so it can be used in static context
    static class Task {
        int id;
        String name;
```

<terminated> Stage_Based_Task_Runner [Java Application] C:\Users\91939\Downloads\spring-tools-for-e

```
Added to Stage 1: Task[1: Task1]
Processing in Stage 1: Task[1: Task1]
Filtered out (odd ID): Task[1: Task1]
Added to Stage 1: Task[2: Task2]
Processing in Stage 1: Task[2: Task2]
Moved to Stage 2: Task[2: Task2]
Processing in Stage 2: Task[2: Task2]
Added to Stage 1: Task[3: Task3]
Added to Stage 1: Task[4: Task4]
Processing in Stage 1: Task[3: Task3]
Filtered out (odd ID): Task[3: Task3]
Added to Stage 1: Task[5: Task5]
```

Undo_Stack :- To store commands in a stack. Perform Undo by removing the last command. Perform Redo by reversing that undone command and adding it back.



```java
package com.wipro.sindhu.Assignment;
import java.util.*;

public class Undo_Stack {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
        Stack<String> commandStack = new Stack<>();

```

<terminated> Undo_Stack [Java Application] C:\Users\91939\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win32.x86

```
open
close
open close
Undo_command: open close
Redo_command (reversed): esolc nepo

Final Command Stack:
esolc nepo
close
open
```