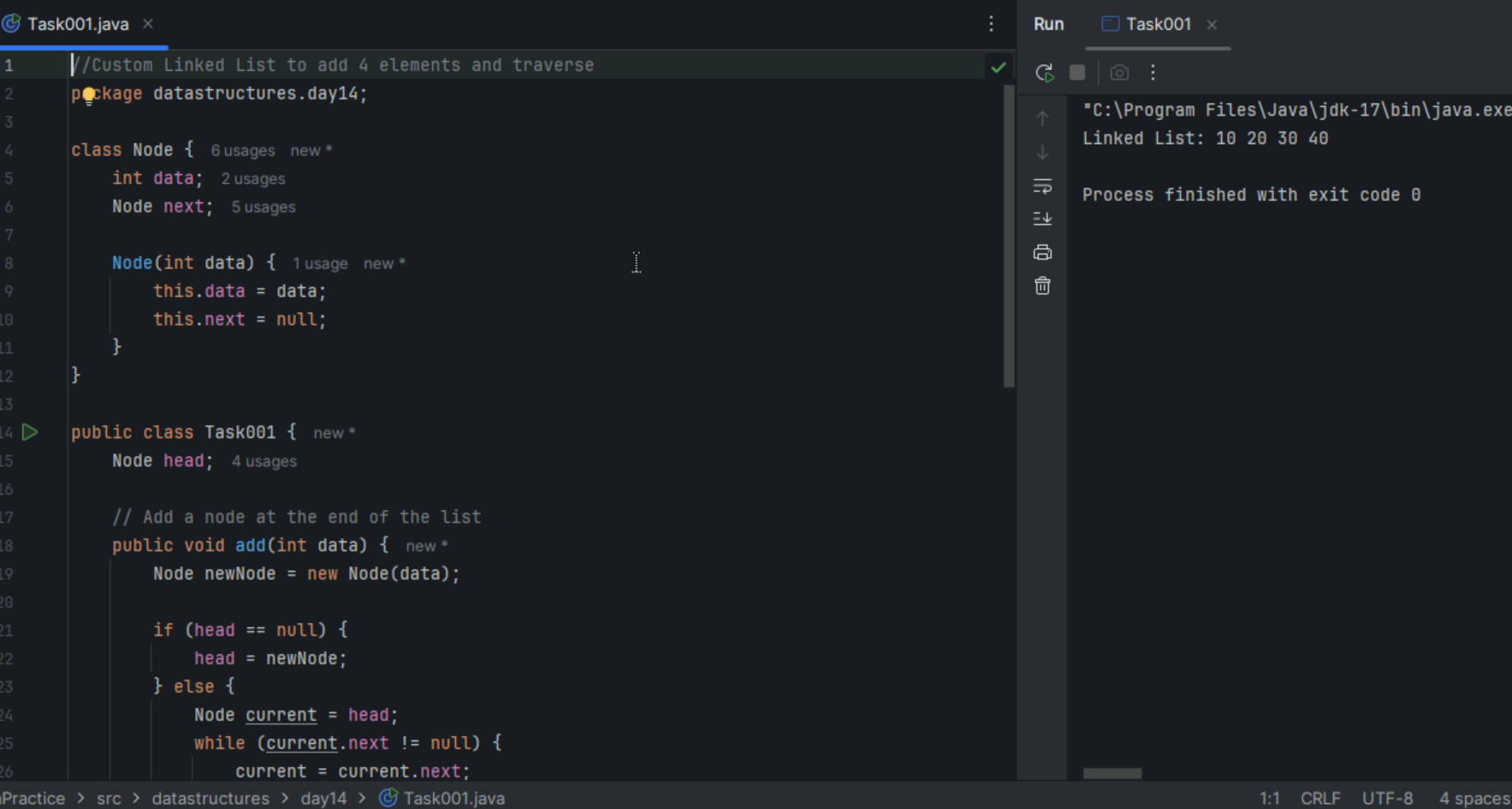
Day 14 - 104608492 - Shirisha Perapagu

Stacks and Queues and Recursion

Task001

Create  a custom node, add elements to it and traverse it.



Task002

What do you understand by traversing a linked list?

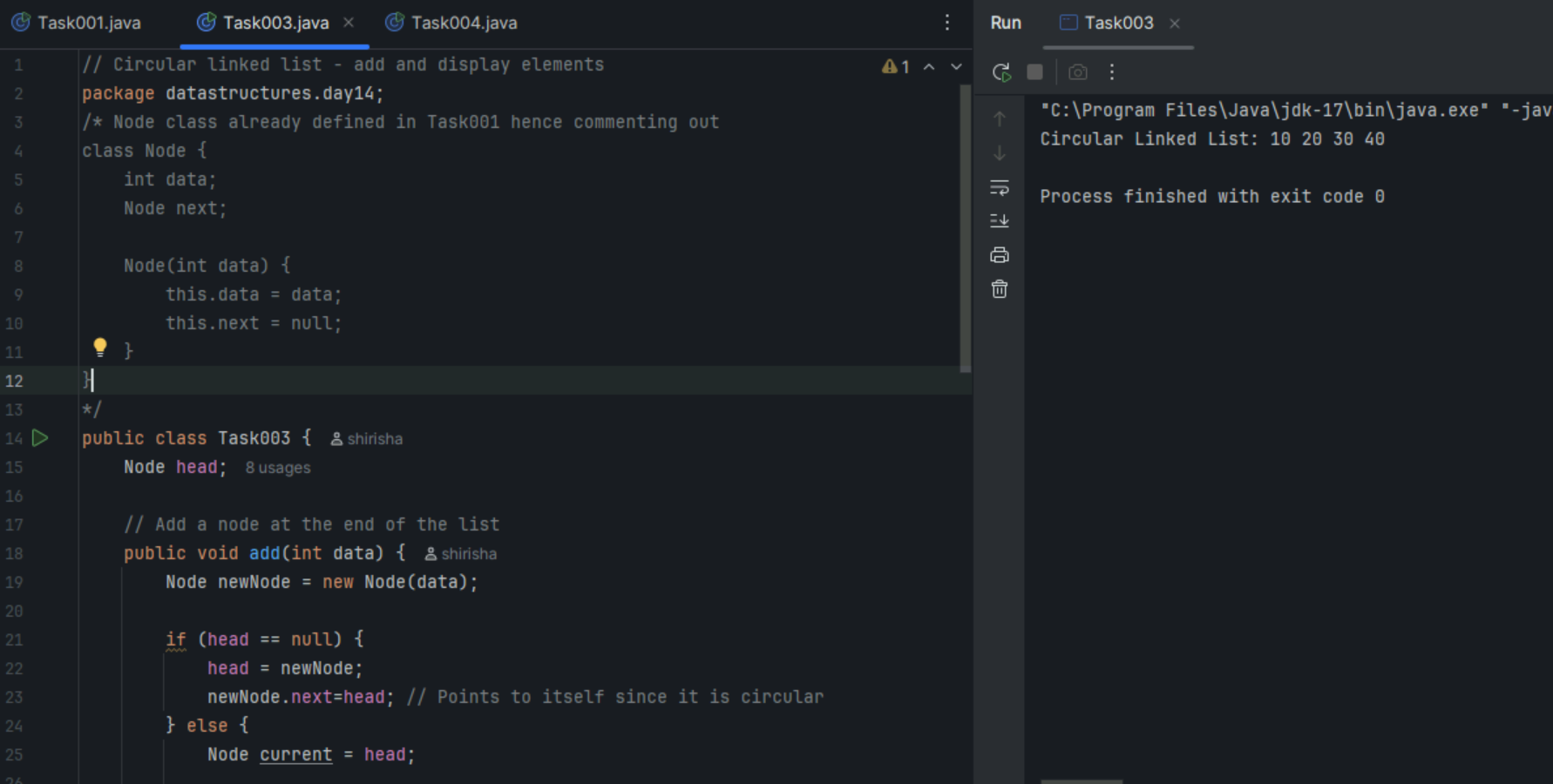
It is going through the elements one by one from starting to ending of the list.

It doesn’t mean we traverse only to display elements.

We usually do this when we want to print all elements, searching for a specific value etc.

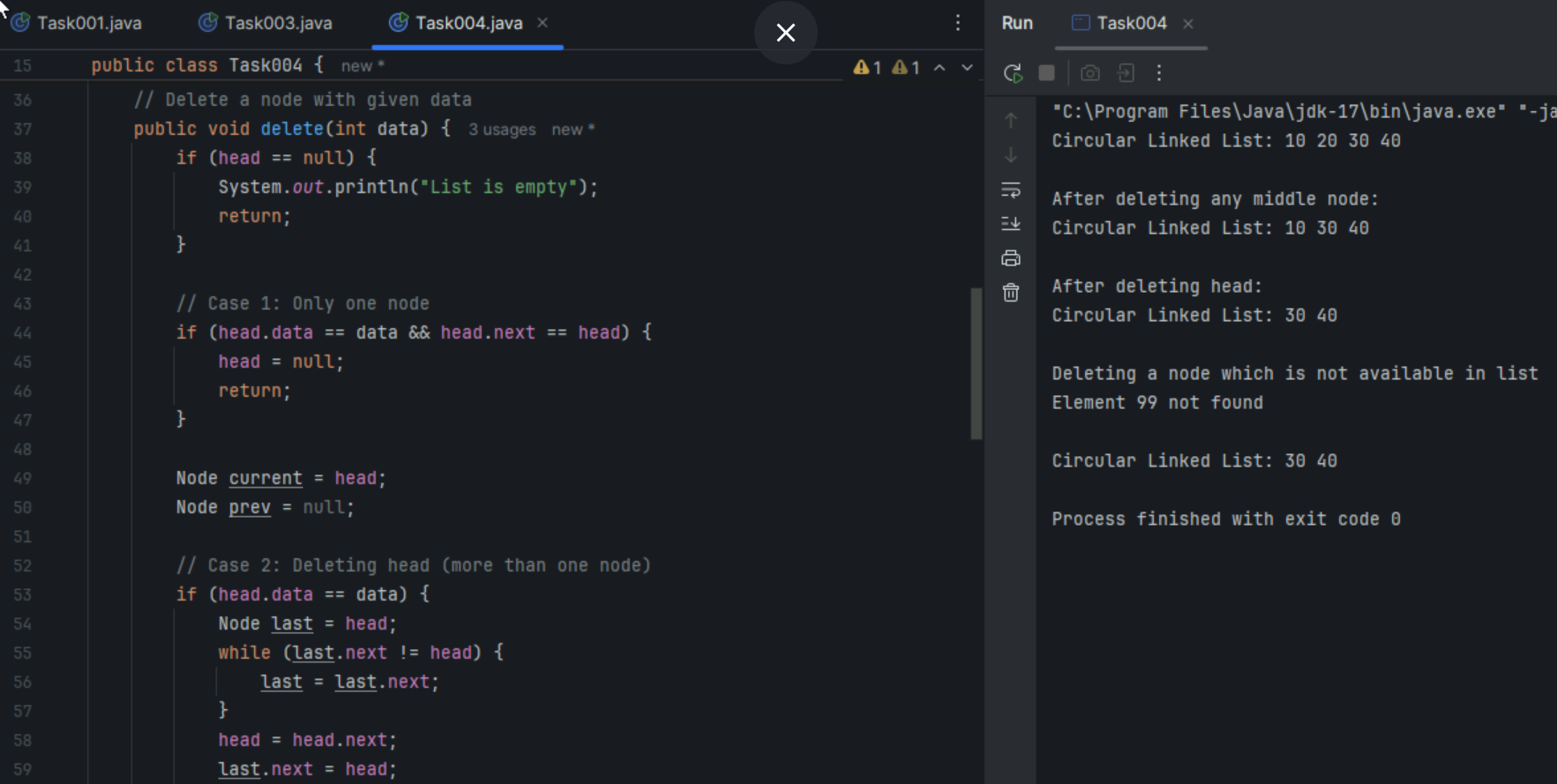
Task003

Create a Circular Linked list using Task001 Singly linked list.



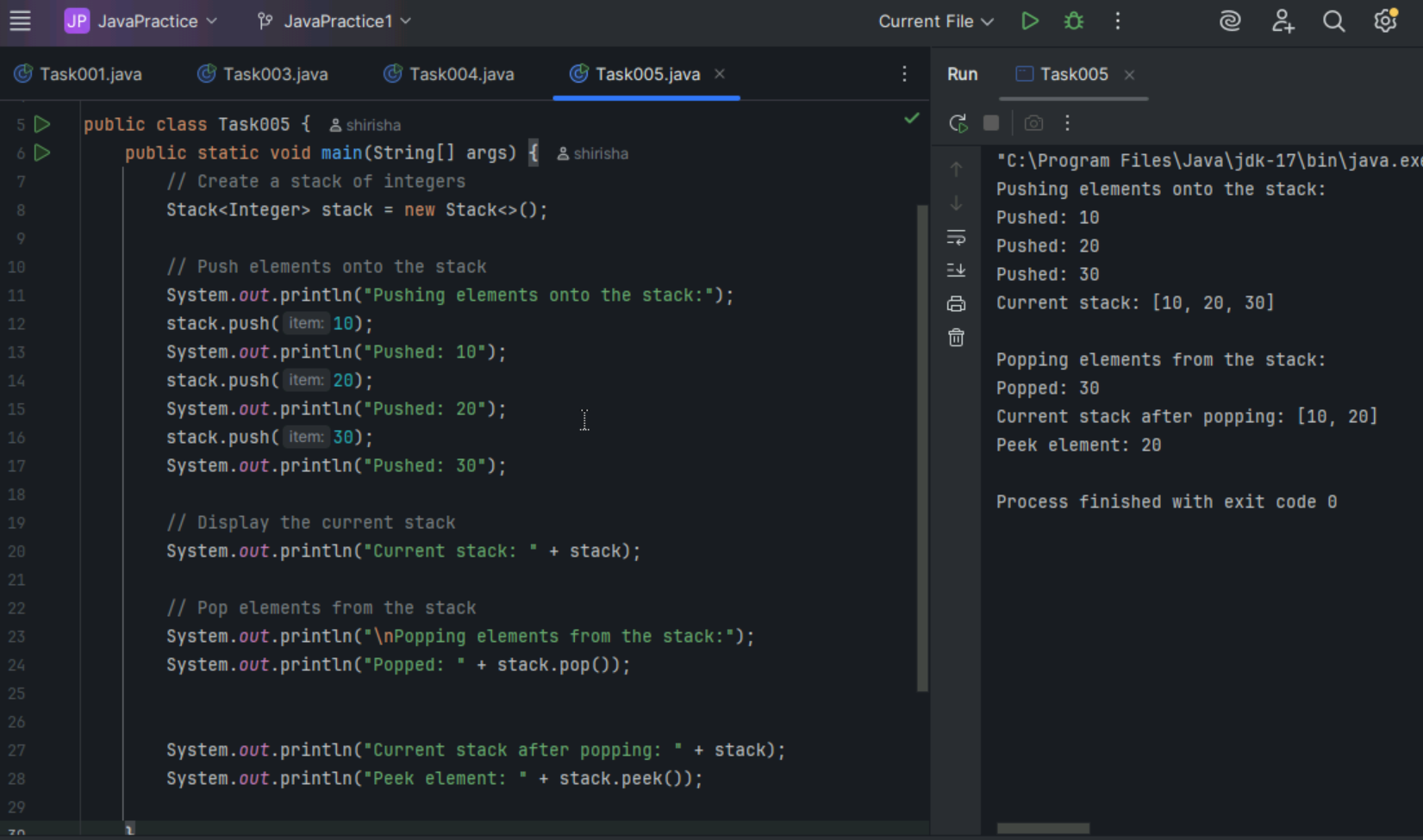
Task004

Now delete a node in Task003 Circular Linked List.



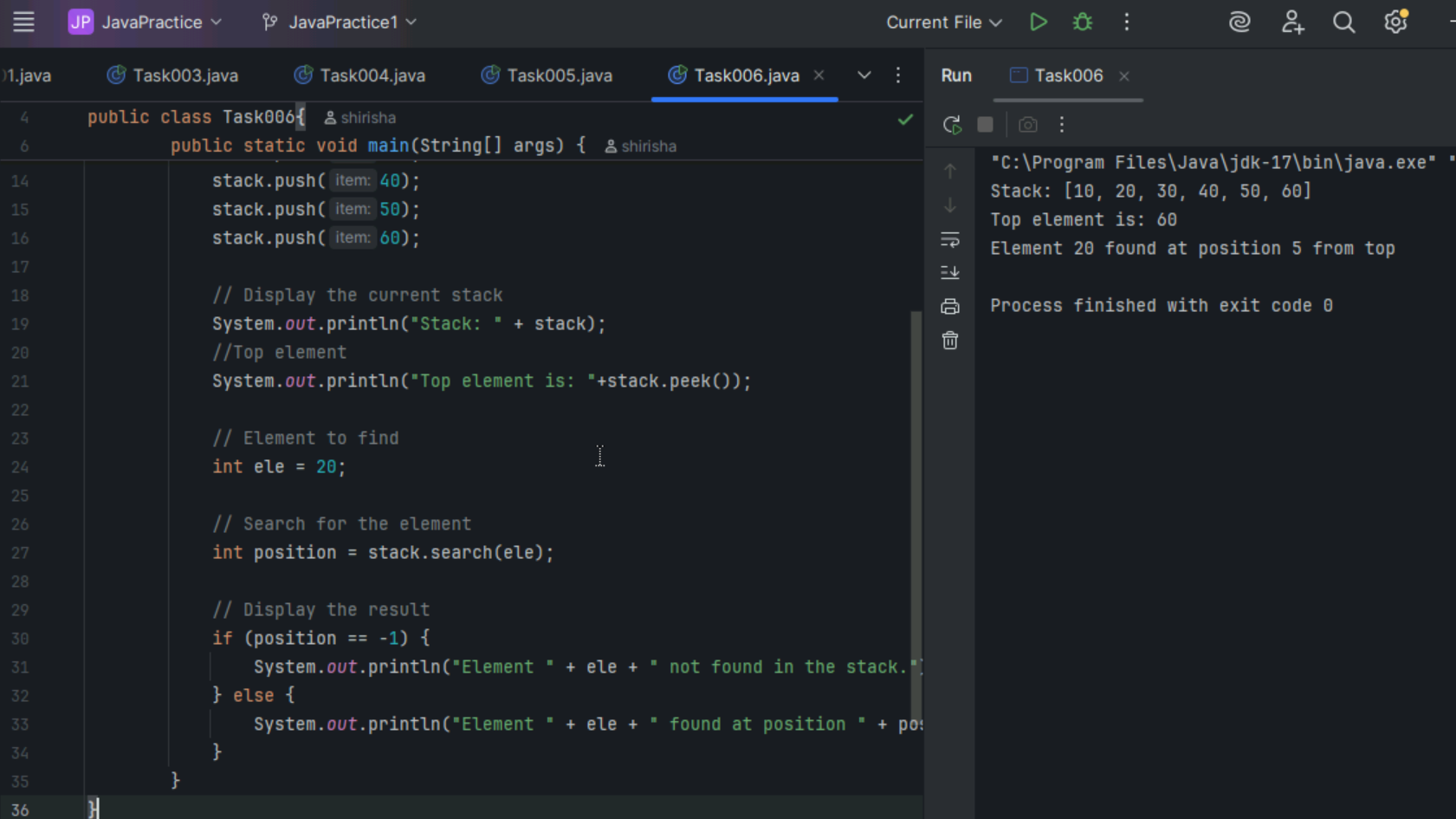
Task005

Create a stack and pop the element also print the popped element



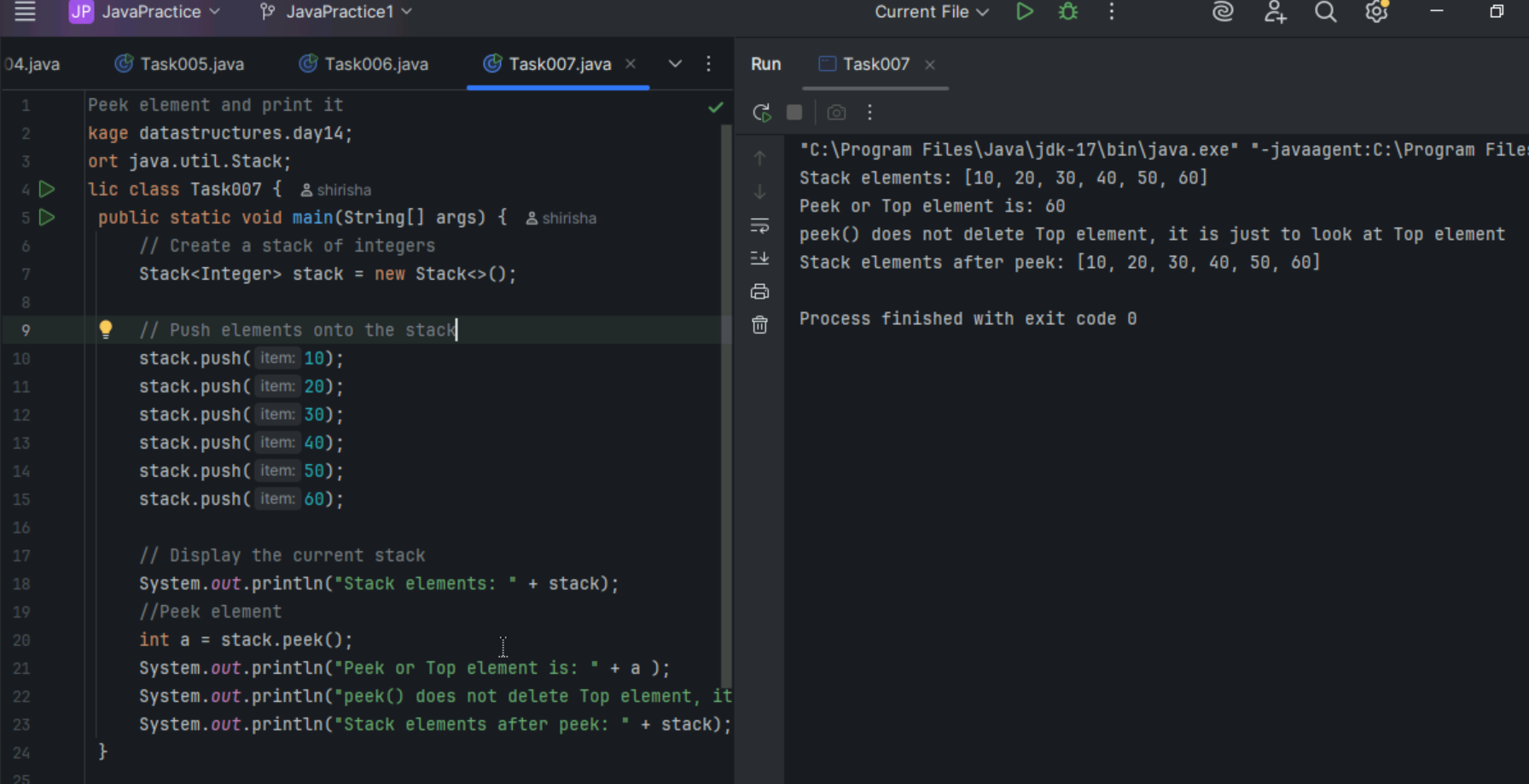
Task006

Find an element in the stack and display the position



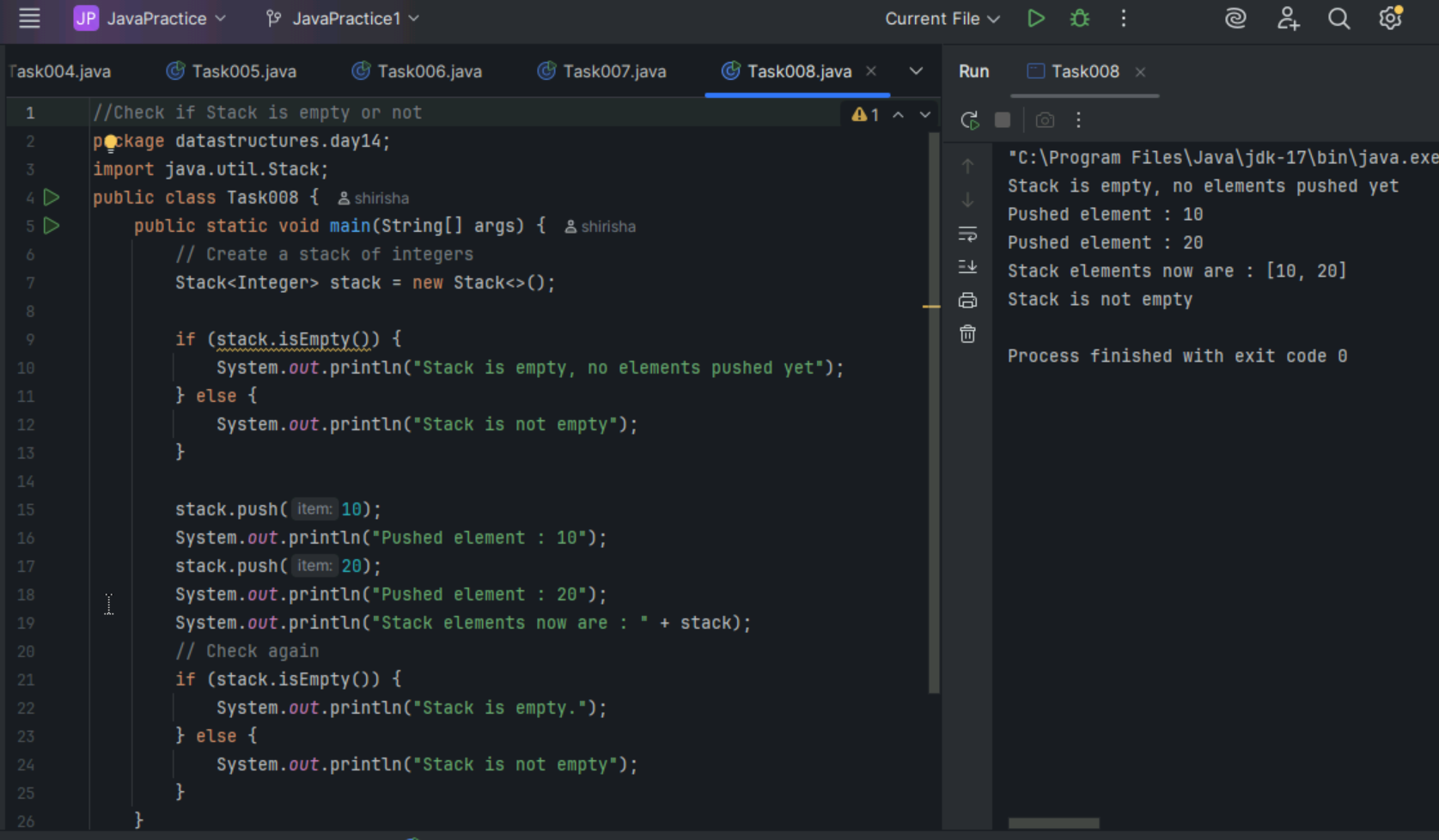
Task007

Peek the element and print it



Task008

Check if the stack is empty or not?



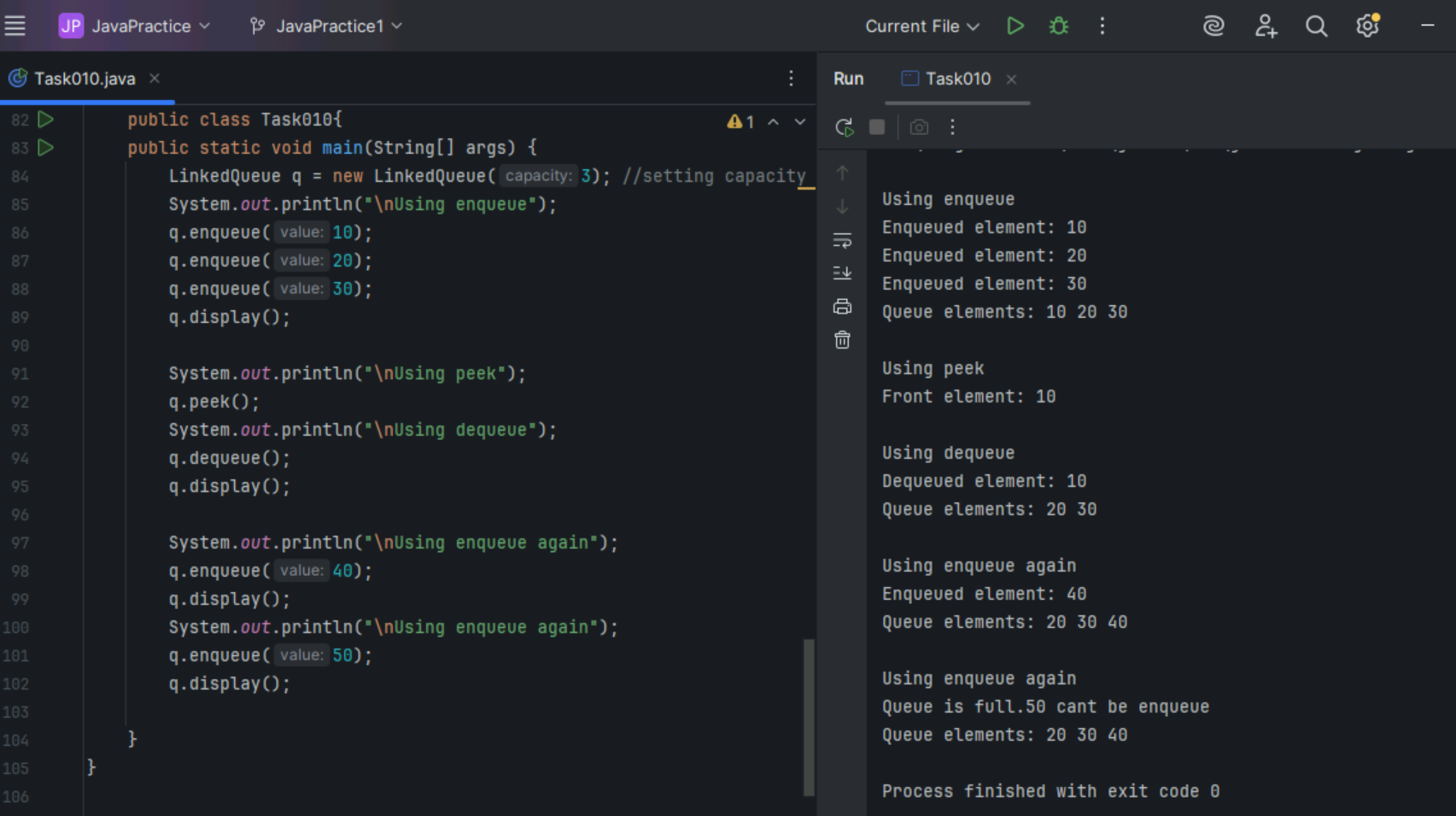
Task009

List methods of Stack class and write one line about it

|  |  |
| --- | --- |
| Stack method | Description |
| push(Object o) | Pushes an element onto the top of stack. |
| pop() | Pops or removes and returns the top element of stack. |
| peek() | Returns the element on the top of the stack but does not remove it. |
| search(Object o) | Searches for an element in stack. If the element is found, it returns the position of the element from the top of the stack else it returns -1. |
| empty() | Returns true if nothing is on the top of the stack else returns false. |

Task010

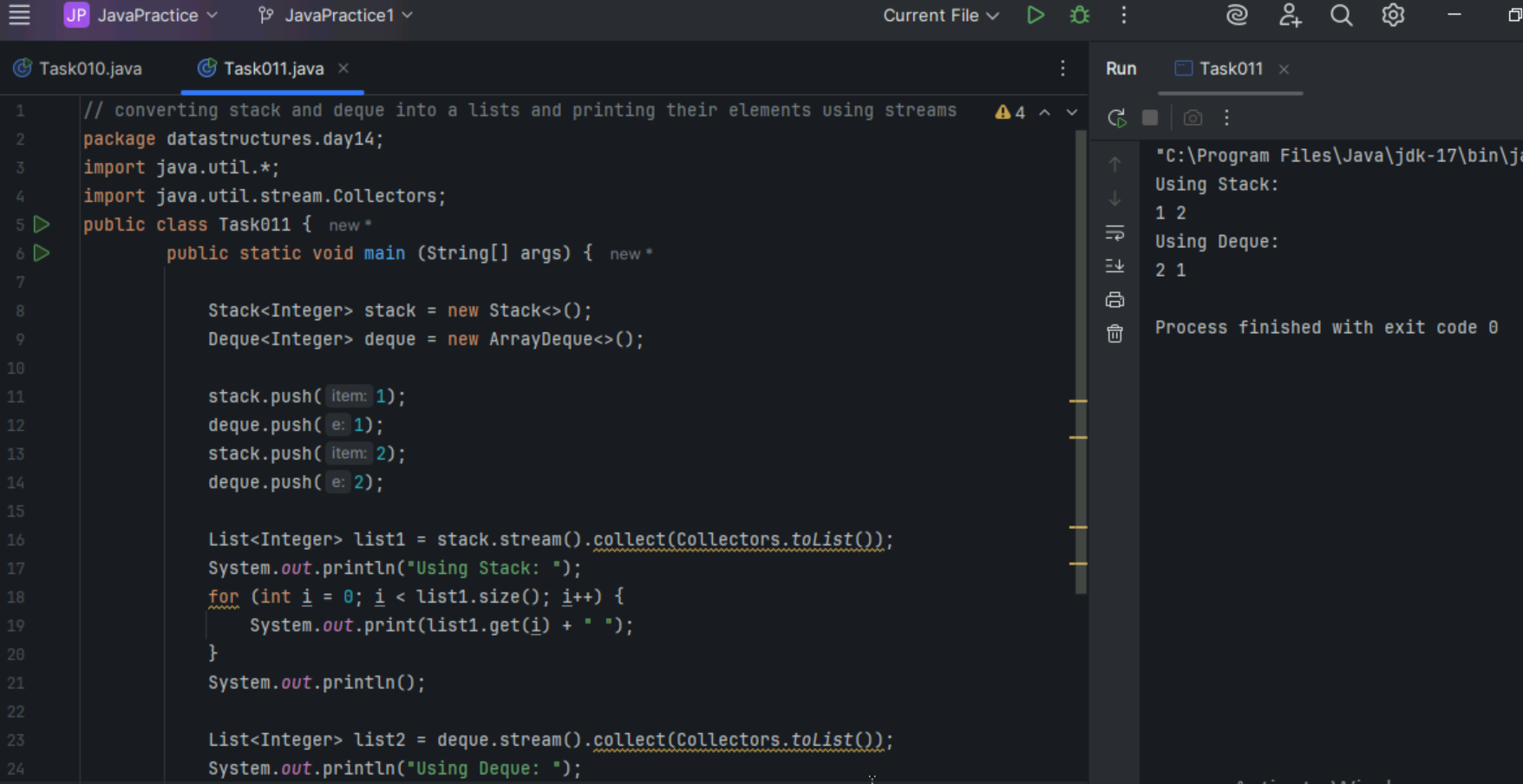
Queue operations using LinkedList.



Home Tasks

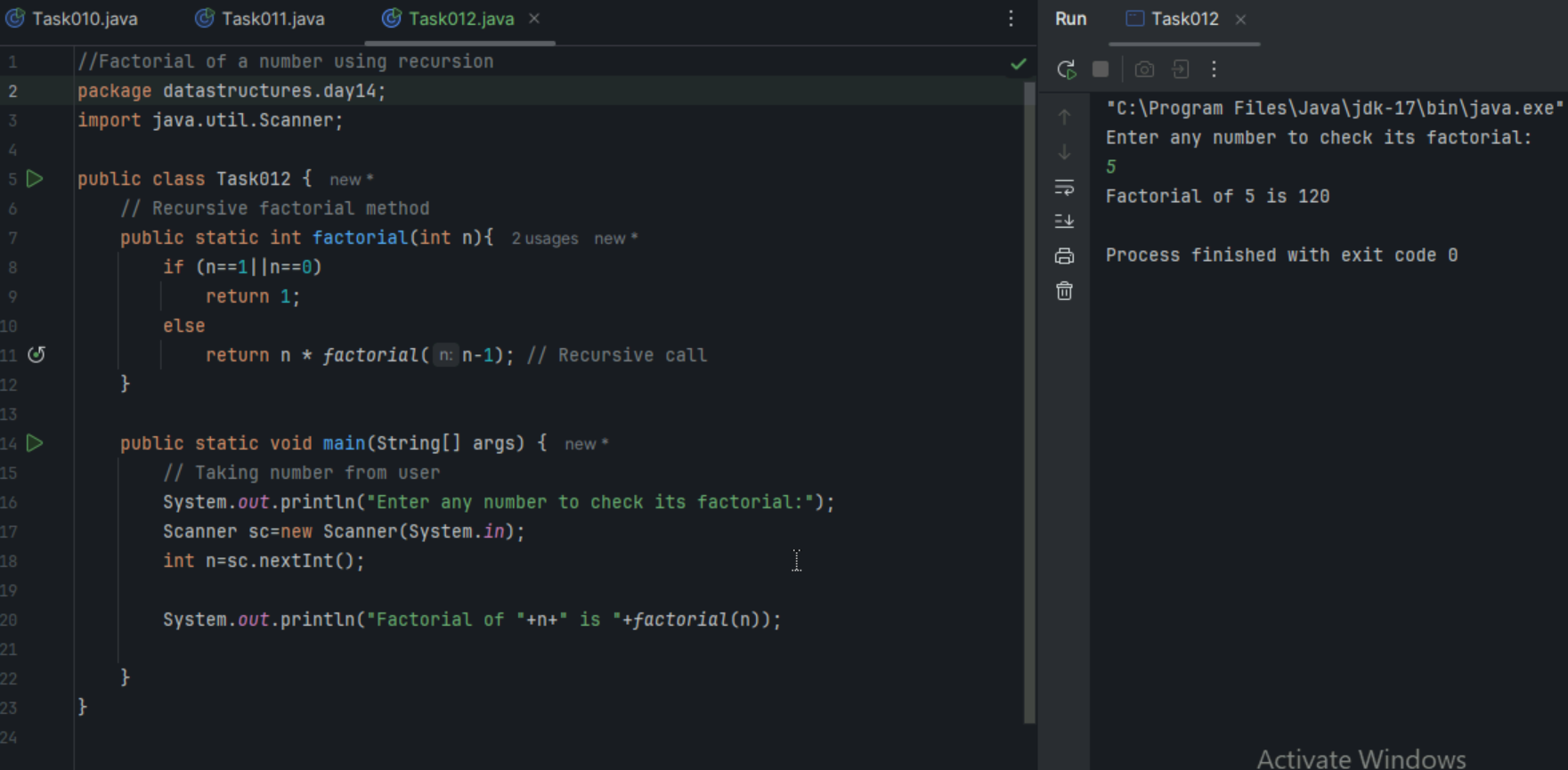
Task011

Convert stack and deque into a lists and print their elements in java using streams.



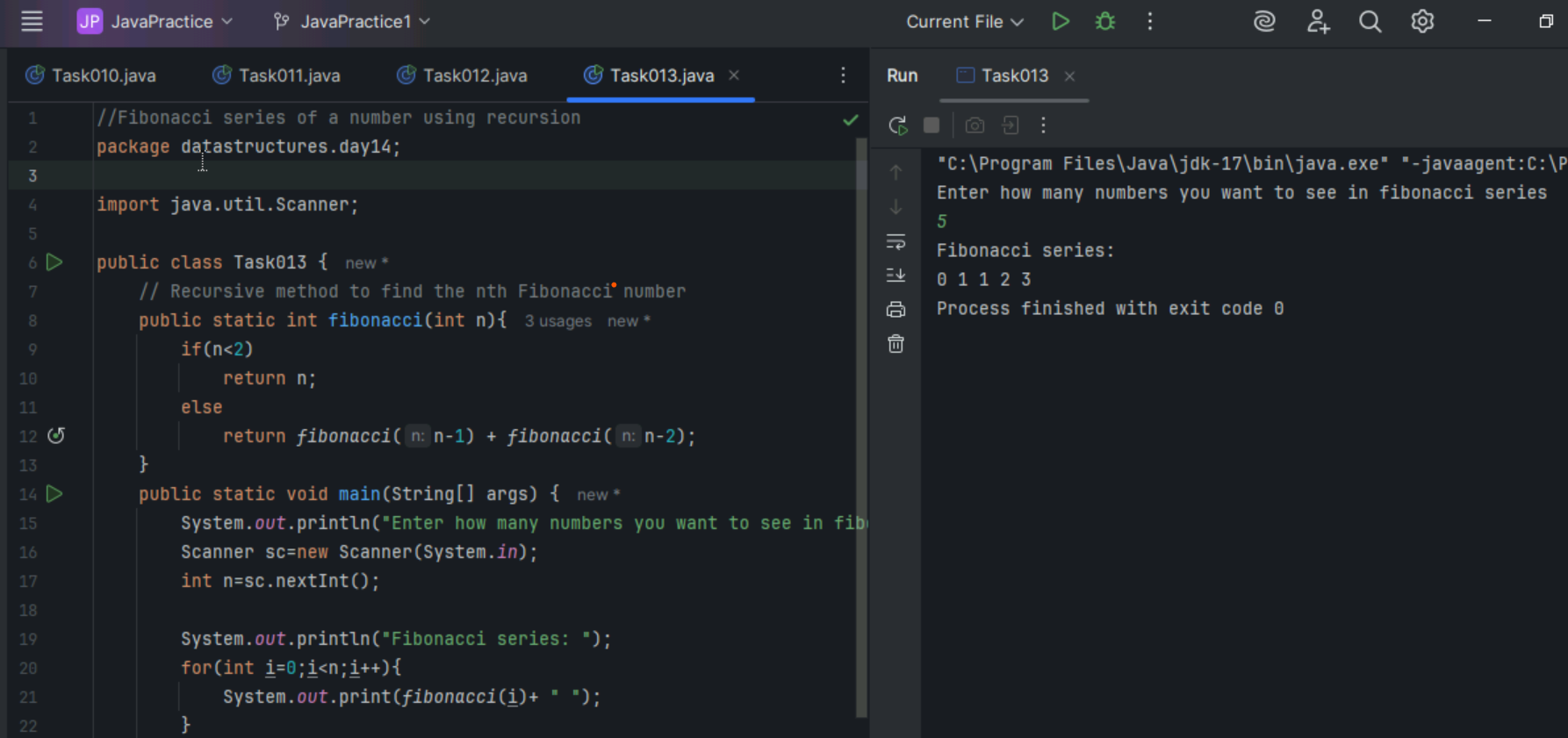
Task012

Wap to print factorial of a number



Task013

Wap to find the Fibonacci series of a number



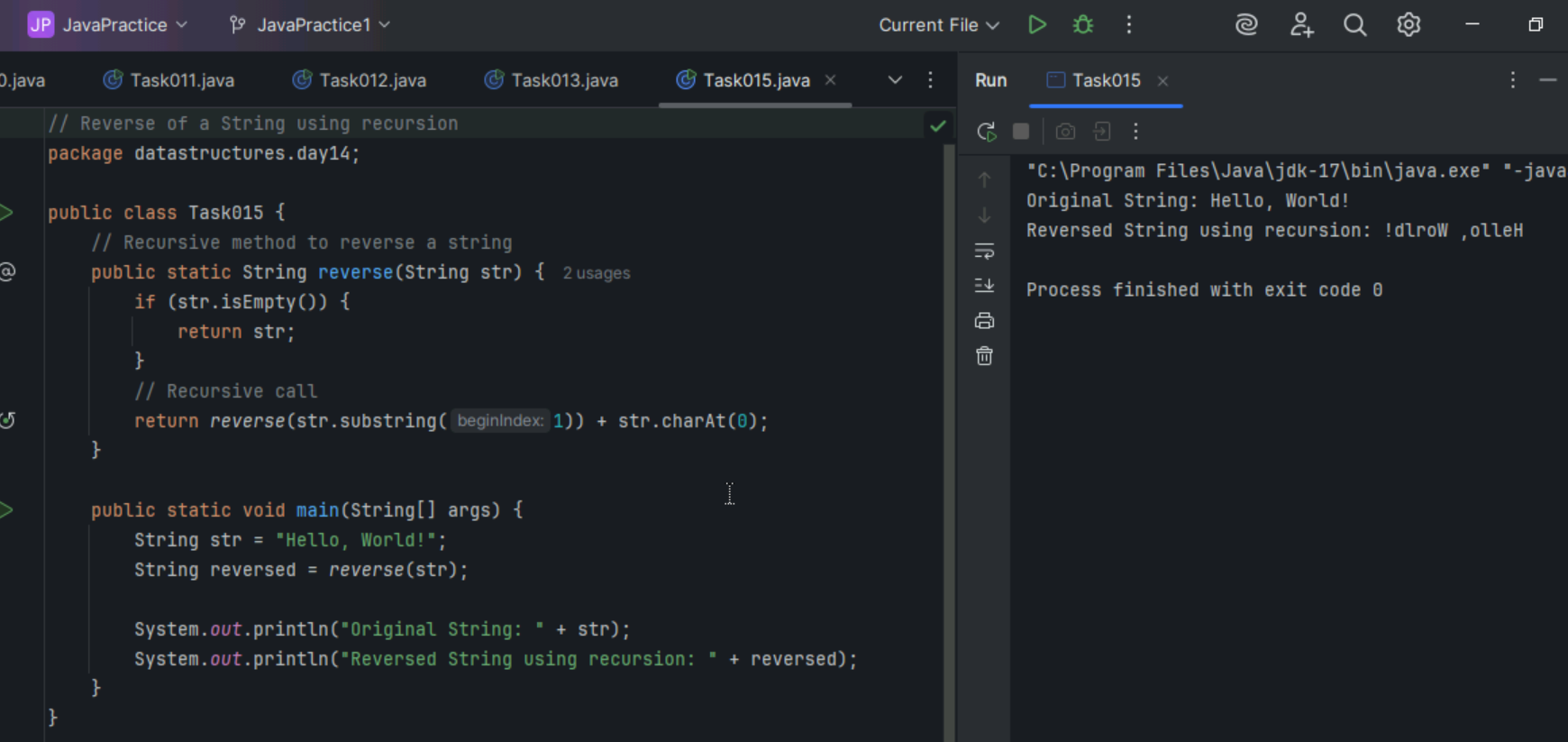
Task 014

What is the difference between iteration and recursion

|  |  |  |
| --- | --- | --- |
| Feature | Iteration | Recursion |
| What is it | Loop repeats steps | Method calls itself |
| Memory usage | Low | High as it uses stack memory |
| Speed | Faster | Can be slower |
| Used for | Simple problems like printing numbers, summing an array | Simple problems like factorial, Fibonacci or tree traversal |

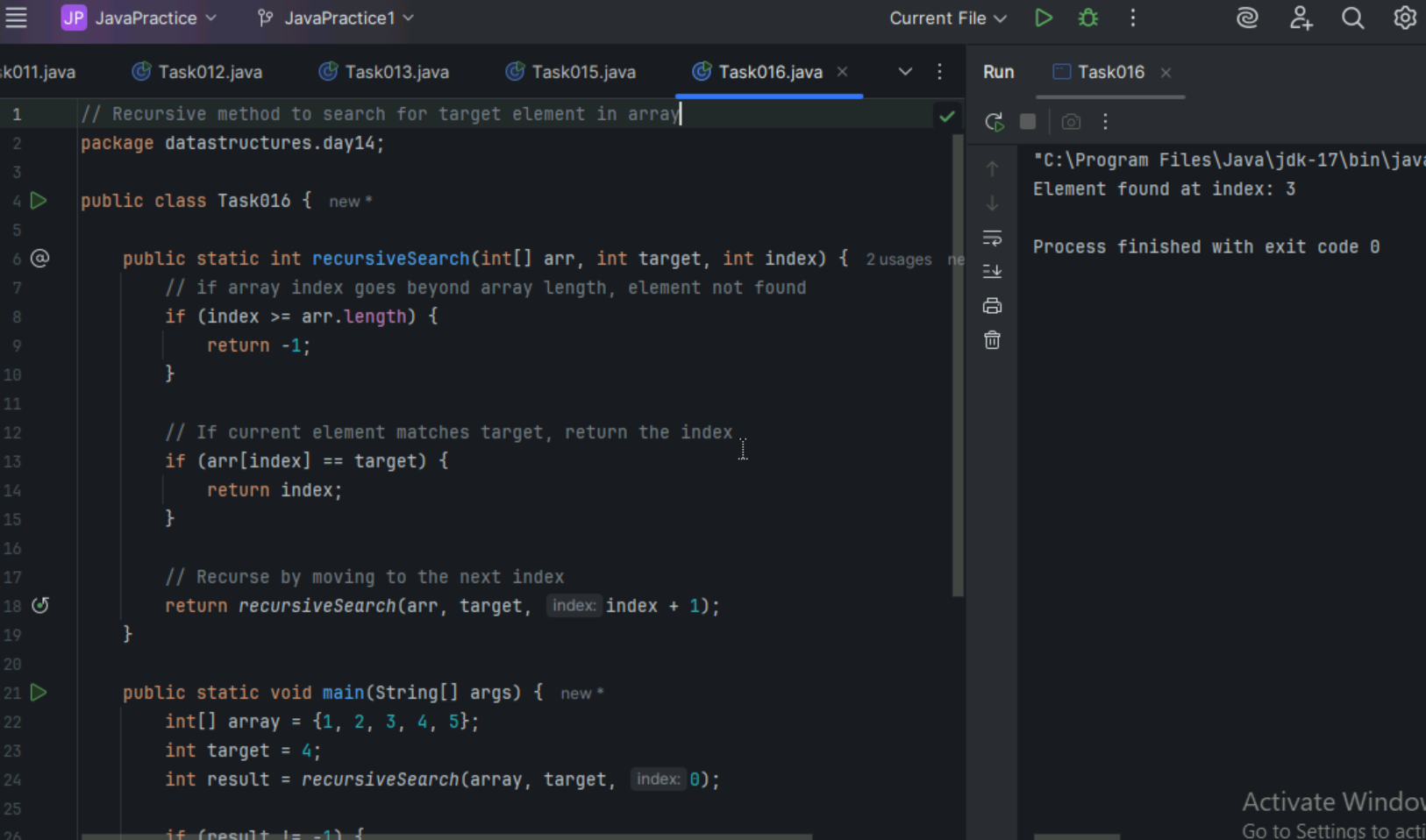
Task015

Wap to reverse a string using recursion



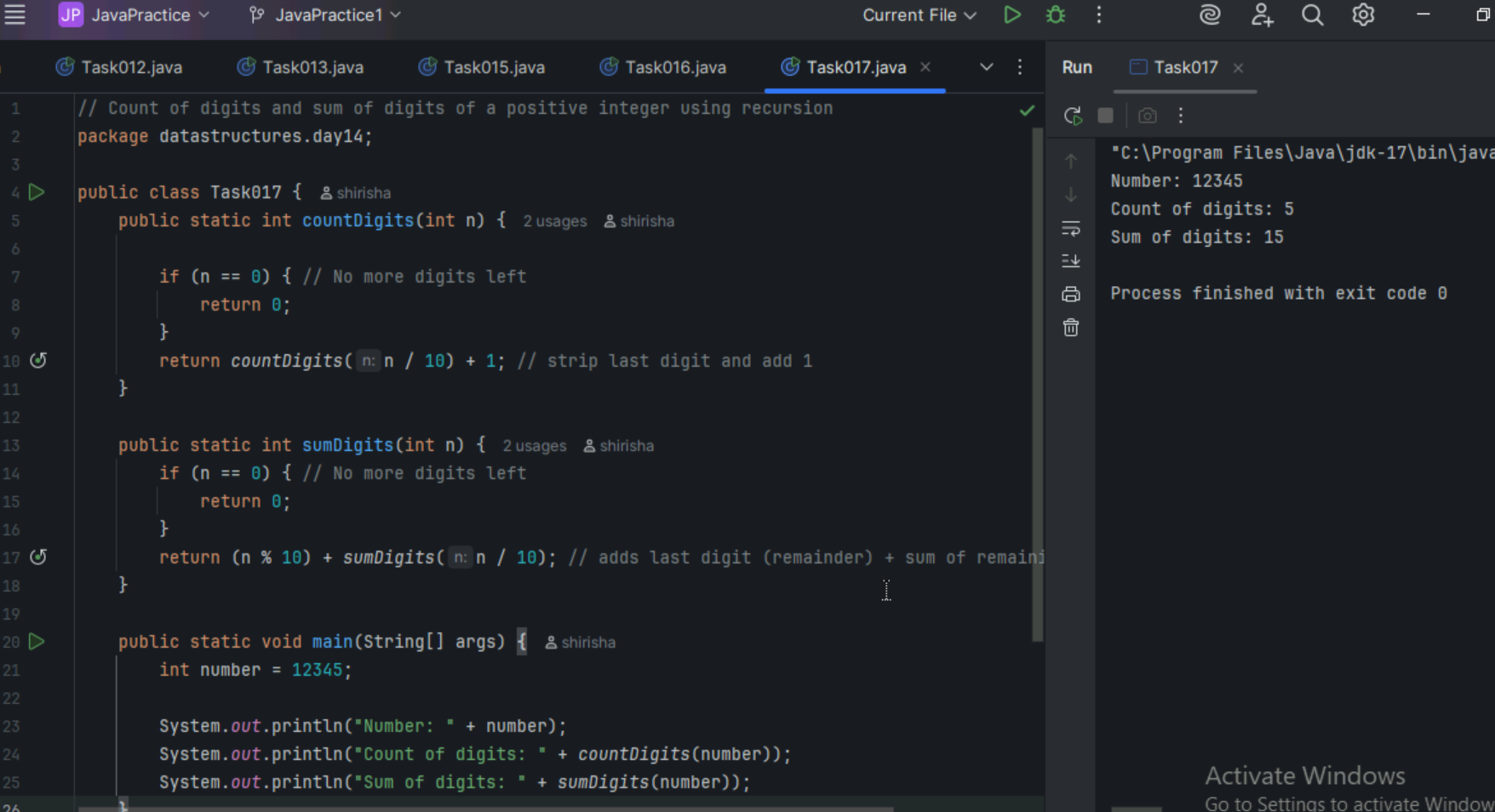
Task 016

Write a recursive method to search for an element in an array



Task017

Write a recursive method to count the digits of a positive integer (do also for sum of digits)



Task 018

Convert a decimal number to binary using recursion

