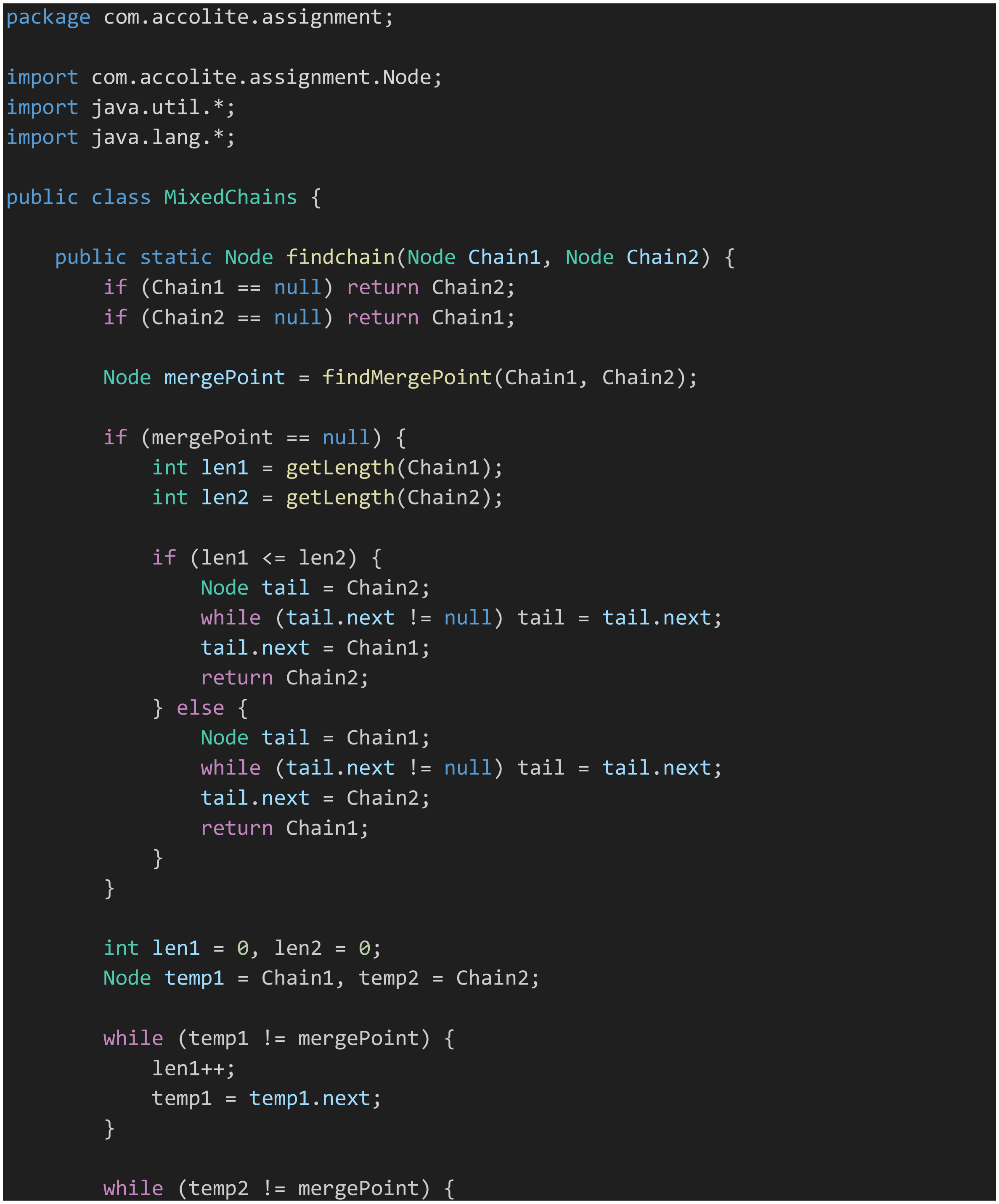
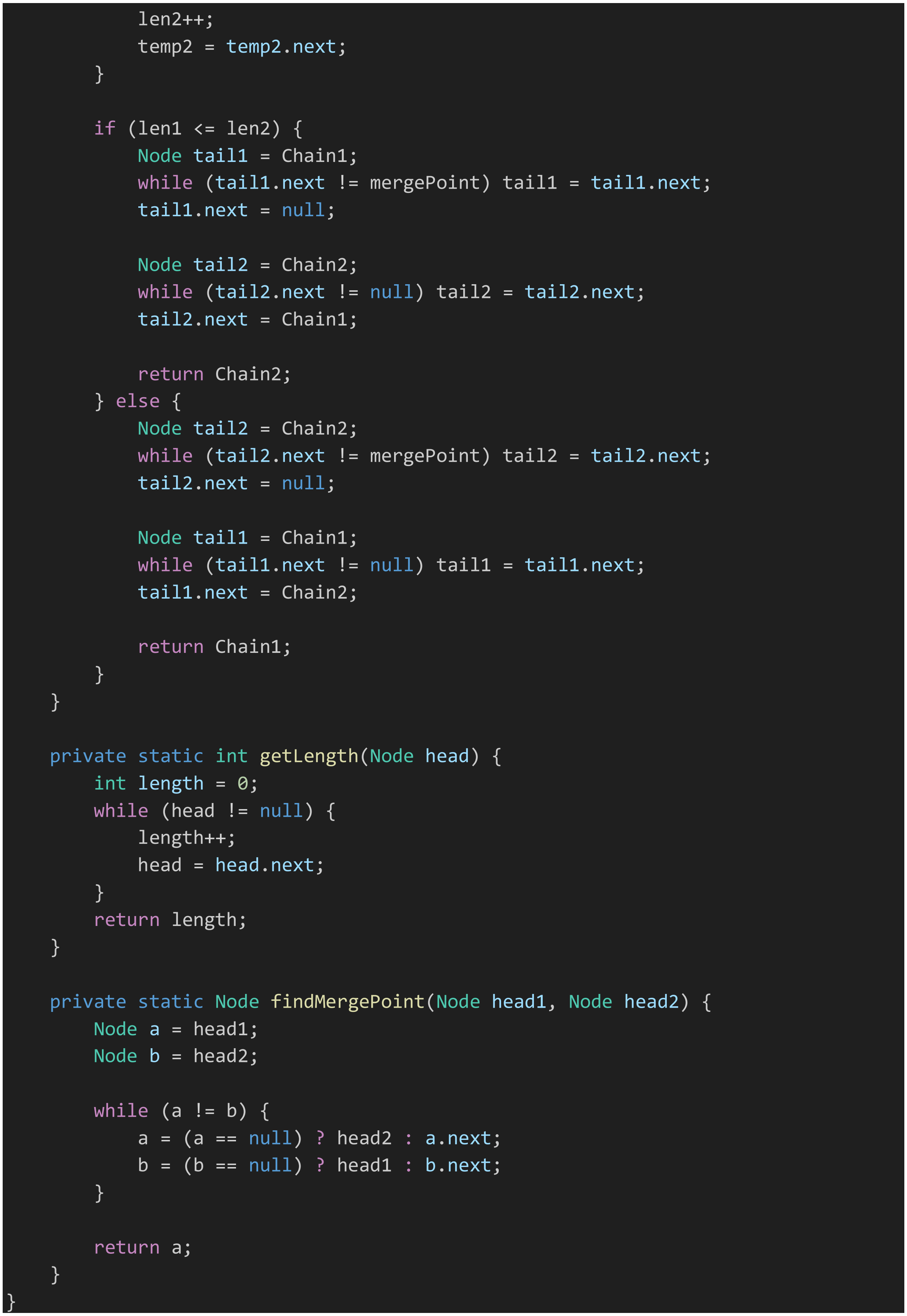
1. **Go to Explorer**

   Src- main - java - com - accolite – assignment – Mixed Chain . Java

2. **Remove line 10 & 11**
3. **Write this code**



```java
package com.accolite.assignment;

import com.accolite.assignment.Node;
import java.util.*;
import java.lang.*;

public class MixedChains {

    public static Node findchain(Node Chain1, Node Chain2) {
        if (Chain1 == null) return Chain2;
        if (Chain2 == null) return Chain1;

        Node mergePoint = findMergePoint(Chain1, Chain2);

        if (mergePoint == null) {
            int len1 = getLength(Chain1);
            int len2 = getLength(Chain2);

            if (len1 <= len2) {
                Node tail = Chain2;
                while (tail.next != null) tail = tail.next;
                tail.next = Chain1;
                return Chain2;
            } else {
                Node tail = Chain1;
                while (tail.next != null) tail = tail.next;
                tail.next = Chain2;
                return Chain1;
            }
        }

        int len1 = 0, len2 = 0;
        Node temp1 = Chain1, temp2 = Chain2;

        while (temp1 != mergePoint) {
            len1++;
            temp1 = temp1.next;
        }

        while (temp2 != mergePoint) {
```



```
            len2++;
            temp2 = temp2.next;
        }

        if (len1 <= len2) {
            Node tail1 = Chain1;
            while (tail1.next != mergePoint) tail1 = tail1.next;
            tail1.next = null;

            Node tail2 = Chain2;
            while (tail2.next != null) tail2 = tail2.next;
            tail2.next = Chain1;

            return Chain2;
        } else {
            Node tail2 = Chain2;
            while (tail2.next != mergePoint) tail2 = tail2.next;
            tail2.next = null;

            Node tail1 = Chain1;
            while (tail1.next != null) tail1 = tail1.next;
            tail1.next = Chain2;

            return Chain1;
        }
    }

    private static int getLength(Node head) {
        int length = 0;
        while (head != null) {
            length++;
            head = head.next;
        }
        return length;
    }

    private static Node findMergePoint(Node head1, Node head2) {
        Node a = head1;
        Node b = head2;

        while (a != b) {
            a = (a == null) ? head2 : a.next;
            b = (b == null) ? head1 : b.next;
        }

        return a;
    }
}
```