Nama : Deanissa Sherly Sabilla

Kelas / Absen : 1B SIB / 06

JOBSHEET IX

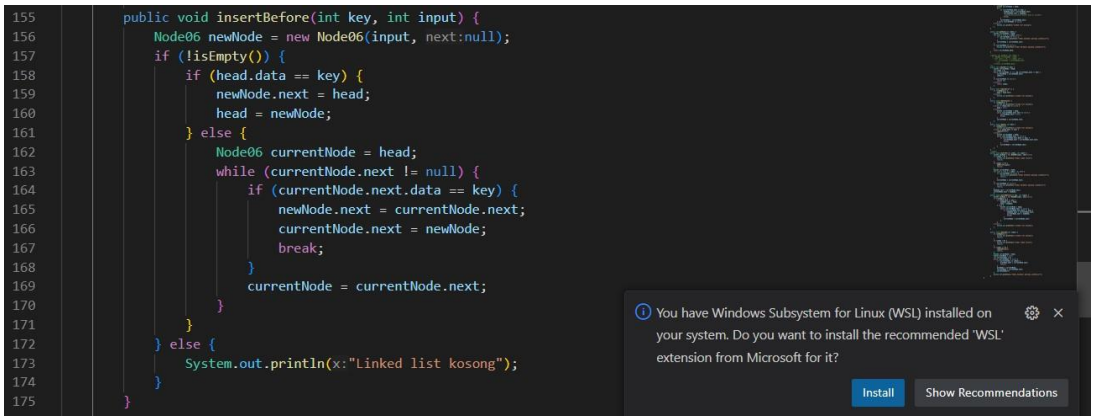
LINKED LIST

3. Tugas

Waktu pengerjaan: 50 menit

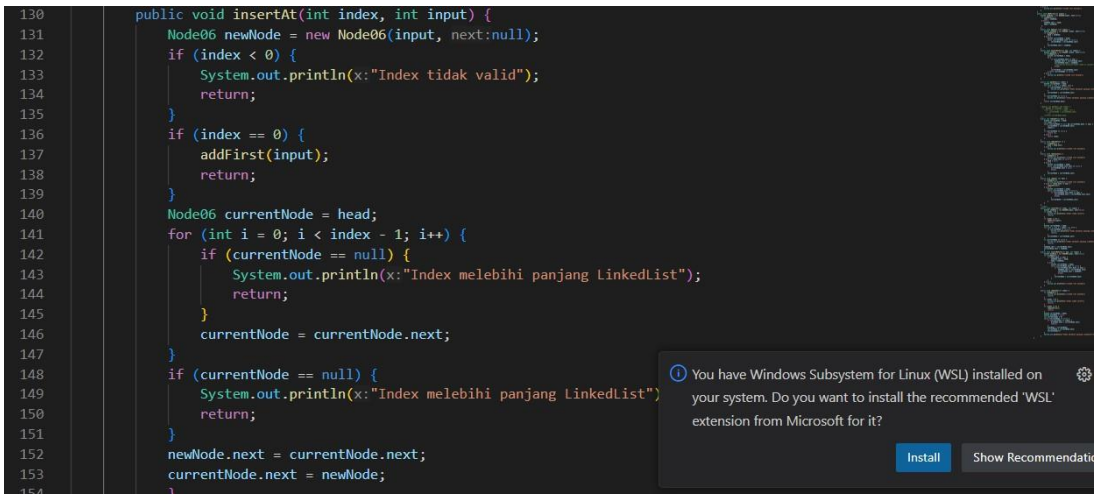
1. Implementasikan method-method berikut pada class LinkedList:

   a. insertBefore() untuk menambahkan node sebelum keyword yang diinginkan



```
public void insertBefore(int key, int input) {
    Node06 newNode = new Node06(input, next:null);
    if (!isEmpty()) {
        if (head.data == key) {
            newNode.next = head;
            head = newNode;
        } else {
            Node06 currentNode = head;
            while (currentNode.next != null) {
                if (currentNode.next.data == key) {
                    newNode.next = currentNode.next;
                    currentNode.next = newNode;
                    break;
                }
                currentNode = currentNode.next;
            }
        }
    } else {
        System.out.println(x:"Linked list kosong");
    }
}
```


   b. insertAt(int index, int key) untuk menambahkan node pada index tertentu



```
public void insertAt(int index, int input) {
    Node06 newNode = new Node06(input, next:null);
    if (index < 0) {
        System.out.println(x:"Index tidak valid");
        return;
    }
    if (index == 0) {
        addFirst(input);
        return;
    }
    Node06 currentNode = head;
    for (int i = 0; i < index - 1; i++) {
        if (currentNode == null) {
            System.out.println(x:"Index melebihi panjang LinkedList");
            return;
        }
        currentNode = currentNode.next;
    }
    if (currentNode == null) {
        System.out.println(x:"Index melebihi panjang LinkedList");
        return;
    }
    newNode.next = currentNode.next;
    currentNode.next = newNode;
```

c. removeAt(int index) untuk menghapus node pada index tertentu

```
177     public void removeAt(int index) {
178         if (isEmpty()) {
179             System.out.println(x:"Linked list kosong");
180             return;
181         } if (index < 0) {
182             System.out.println(x:"Index tidak valid");
183             return;
184         } if (index == 0) {
185             removeFirst();
186             return;
187         }
188         Node06 currentNode = head;
189         Node06 prevNode = null;
190         int currentIndex = 0;
191         while (currentNode != null) {
192             if (currentIndex == index) {
193                 prevNode.next = currentNode.next;
194                 return;
195             }
196             prevNode = currentNode;
197             currentNode = currentNode.next;
198             currentIndex++;
199         }
200         System.out.println(x:"Index melebihi panjang LinkedList");
201     }
```


Class Main :

```
SLLMain06.java > SLLMain06 > main(String[])
1   public class SLLMain06 {
2     public static void main(String[] args) {
3       LinkedList06 myLinkedList = new LinkedList06();
4       myLinkedList.print();
5       myLinkedList.addFirst(input:800);
6       myLinkedList.print();
7       myLinkedList.addFirst(input:700);
8       myLinkedList.print();
9       myLinkedList.addLast(input:500);
10      myLinkedList.print();
11      myLinkedList.insertAfter(key:700, input:300);
12      myLinkedList.print();
13      System.out.println("Data pada index ke-1 : " +myLinkedList.getData(index:1));
14      System.out.println("Data 300 berada pada index ke : " +myLinkedList.indexOf(key:300));
15      myLinkedList.remove(key:300);
16      myLinkedList.print();
17      myLinkedList.removeFirst();
18      myLinkedList.print();
19      myLinkedList.removeLast();
20      myLinkedList.print();
21      myLinkedList.insertAt(index:1, input:400);
22      myLinkedList.print();
23      myLinkedList.insertBefore(key:500, input:600);
24      myLinkedList.print();
25      myLinkedList.removeAt(index:2);
26      myLinkedList.print();
```

Tambahan :

```
21      myLinkedList.insertAt(index:1, input:400);
22      myLinkedList.print();
23      myLinkedList.insertBefore(key:500, input:600);
24      myLinkedList.print();
25      myLinkedList.removeAt(index:2);
26      myLinkedList.print();
```

Hasil Output :

```
Linked list kosong
Isi linked list : 800
Isi linked list : 700   800
Isi linked list : 700   800     500
Isi linked list : 700   300     800     500
Data pada index ke-1 : 300
Data 300 berada pada index ke : 1
Isi linked list : 700   800     500
Isi linked list : 800   500
Isi linked list : 800
Isi linked list : 800   400
Isi linked list : 800   400
Index melebihi panjang LinkedList
Isi linked list : 800   400
```

2. Dalam suatu game scavenger hunt, terdapat beberapa point yang harus dilalui peserta untuk menemukan harta karun. Setiap point memiliki soal yang harus dijawab, kunci jawaban, dan pointer ke point selanjutnya. Buatlah implementasi game tersebut dengan linked list.

Class GameScavenger :

GameScavenger.java > ...

```java
public class GameScavenger {
        String question;
        String answer;
        GameScavenger next;

        public GameScavenger(String question, String answer) {
            this.question = question;
            this.answer = answer;
            this.next = null;
        }
}
```

Class Linked List :

LinkedList.java > LinkedList > playGame()

```java
import java.util.Scanner;
public class LinkedList {
    private GameScavenger head;

    public LinkedList() {
        head = null;
    }
    public void addPoint(String question, String answer) {
        GameScavenger newPoint = new GameScavenger(question, answer);
        if (head == null) {
            head = newPoint;
        } else {
            GameScavenger current = head;
            while (current.next != null) {
                current = current.next;
            }
            current.next = newPoint;
        }
    }
```

```java
    public String playGame() {
        StringBuilder result = new StringBuilder();
        GameScavenger current = head;
        Scanner scanner = new Scanner(System.in);

        while (current != null) {
            System.out.println(current.question);
            String userAnswer = scanner.nextLine();
            if (userAnswer.equalsIgnoreCase(current.answer)) {
                System.out.println(x:"Jawaban Anda benar! ");
                current = current.next;
            } else {
                System.out.println(x:"Jawaban Anda salah. Silakan coba lagi. ");
            }
        }
        scanner.close();
        return result.toString();
}
}
```

Class Main :

```java
public class GameScavengerMain {
    Run | Debug
    public static void main(String[] args) {
        LinkedList game = new LinkedList();

        game.addPoint(question:"Hasil dari perkalian 2 x 5 ?", answer:"10");
        game.addPoint(question:"Apakah sapi memiliki kak 4? (IYA / TIDAK)", answer:"IYA");
        game.addPoint(question:"Berapakah jumlah provinsi di Indonesia?", answer:"38");

        // Mulai permainan
        System.out.println(x:"+++++++++++++++++++++++++++++++++++++");
        System.out.println(x:"Selamat datang di Game Scavenger Hunt!");
        System.out.println(x:"+++++++++++++++++++++++++++++++++++++");
        String result = game.playGame();
        System.out.println(result);
    }
}
```

Hasil Output :

- **Hasil Jawaban jika sesuai dengan Kunci Jawaban**

```
+++++++++++++++++++++++++++++++++++++
Selamat datang di Game Scavenger Hunt!
+++++++++++++++++++++++++++++++++++++
Hasil dari perkalian 2 x 5 ?
10
Jawaban Anda benar!
Apakah sapi memiliki kak 4? (IYA / TIDAK)
IYA
Jawaban Anda benar!
Berapakah jumlah provinsi di Indonesia?
38
Jawaban Anda benar!
```

- **Hasil Jawaban jika tidak sesuai dengan Kunci Jawaban**

```
+++++++++++++++++++++++++++++++++++++
Selamat datang di Game Scavenger Hunt!
+++++++++++++++++++++++++++++++++++++
Hasil dari perkalian 2 x 5 ?
9
Jawaban Anda salah. Silakan coba lagi.
Hasil dari perkalian 2 x 5 ?
10
Jawaban Anda benar!
Apakah sapi memiliki kak 4? (IYA / TIDAK)
IYA
Jawaban Anda benar!
Berapakah jumlah provinsi di Indonesia?
34
Jawaban Anda salah. Silakan coba lagi.
Berapakah jumlah provinsi di Indonesia?
38
Jawaban Anda benar!
```