

Velkommen!



Johanna
johannph på mattermost
johannph@uio.no på mail!

Kort: Praktisk informasjon

- Undervisningstilbud
 - <https://www.uio.no/studier/emner/matnat/ifi/IN1010/v21/undervisningstilbud/>
 - Jeg har konkrete spørsmål/problemer med min kode -> Labtime!
 - Jeg vil ha mer liveprogrammering -> Plenumstime!
 - Jeg vil jobbe med andre (og kanskje en kjapp recap av forelesning) -> Gruppetime!
 - Jeg vil ha en recap av de vanskeligste konseptene fra forelesning -> Repetisjonsgruppe!
- Oblig 3, frist Mandag 1. mars kl 23.59
- Skriv alt dere lurer på i chatten enten til everybody eller bare til meg 😊

Repetisjon forrige uke

Forrige uke

Interface

HashMap

Send meg direkte melding

1. Hva er interface
2. Hva er forskjellen på interface og superklasse?
3. Hva er HashMap?

Superklasse

Superklasse er lurt hvis flere klasser delvis deler samme funksjonalitet.

Feks. at vi slipper å skrive `hentAlder()`, `hentNavn()`, lik konstruktør og lik `toString` for alle dyreklasser, vi trenger bare gjøre det én gang.

```
1 class Katt extends Dyr{
2     ↵
3     public Katt(String navn, int alder){
4         super(navn, alder);
5     }
6     @Override
7     public String toString(){
8         return "Jeg er en katt!" + super.toString();
9     }
10    ↵
11 }
```

```
1 abstract class Dyr{
2     protected int alder;
3     protected String navn;
4     ↵
5     public Dyr(String navn, int alder){
6         this.navn = navn;
7         this.alder = alder;
8     }
9     public int hentAlder(){
10        return alder;
11    }
12    public String hentNavn(){
13        return navn;
14    }
15    @Override
16    public String toString(){
17        return "Navn: " + navn + "\n" + "Alder: " + alder;
18    }
19 }
```

Interface

Beskriver en gruppe objekter med de samme egenskapene.

Et interface har definert signaturen til noen metoder. Alle klasser som implementerer et interface må ha disse metodene.

Med interface må vi skrive all koden inne i metodene for hver klasse som implementerer interfacet.

```
1 interface HarChip{  
2     · public Person hentEier();  
3     · public String hentId();  
4 }  
5
```

```
1 class Katt extends Dyr implements HarChip{  
2     · private Person eier;  
3     · private String id;  
4     ·  
5     · public Katt(String navn, int alder, Person eier, String id){  
6         ··· super(navn, alder);  
7         ··· this.eier = eier;  
8         ··· this.id = id;  
9     }  
10    · public Person hentEier(){  
11        ··· return eier;  
12    }  
13    · public String hentId(){  
14        ··· return id;  
15    }  
16    · @Override  
17    · public String toString(){  
18        ··· return super.toString() + "\n" + eier.toString();  
19    }  
}
```

Interface og superklasse

Uansett om vi bruker Array, ArrayList eller HashMap må vi fortelle programmet hva vi vil putte oppi.

```
34 →  
35 ... NavnKlasseEllerInterface[] navnVariabel = new NavnKlasseEllerInterface[lengdeArray]; →  
36 ... ArrayList<NavnKlasseEllerInterface> navnVariabel = new ArrayList(); →  
37 ... HashMap<NavnKlasseEllerInterfaceNokkel, NavnKlasseEllerInterfaceVerdi> navnVariabel = new HashMap(); →  
38 →
```


Repetisjon denne uken

Beholder

Et objekt som representerer en gruppe objekter av samme type.

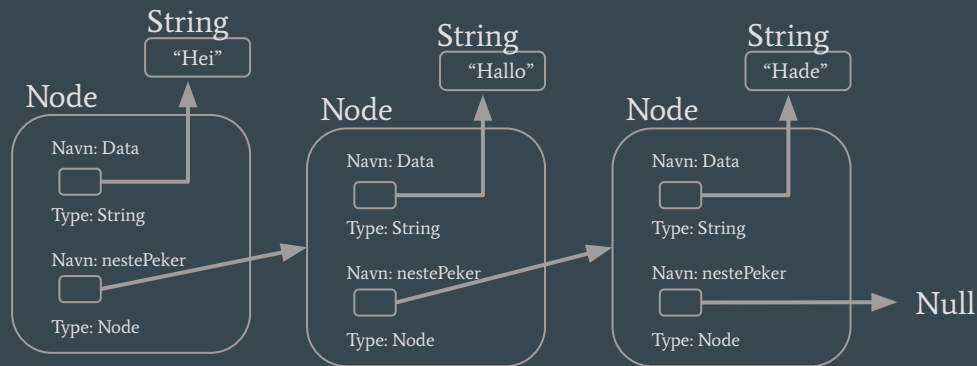
Ofte metoder for å legge til, hente ut og finne størrelse.

Eksempler på beholdere: array(har ingen metoder), ArrayList, HashMap, Lenkeliste

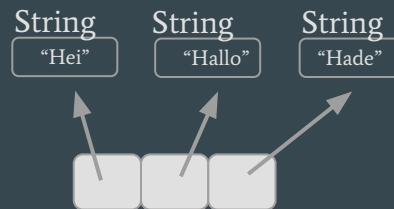
```
34 →  
35 ... NavnKlasseEllerInterface[] navnVariabel = new NavnKlasseEllerInterface[lengdeArray]; →  
36 ... ArrayList<NavnKlasseEllerInterface> navnVariabel = new ArrayList(); →  
37 ... HashMap<NavnKlasseEllerInterfaceNokkel, NavnKlasseEllerInterfaceVerdi> navnVariabel = new HashMap(); →  
38 →
```

Lenkeliste

Vi skal se på en spesiell type beholder: lenkeliste



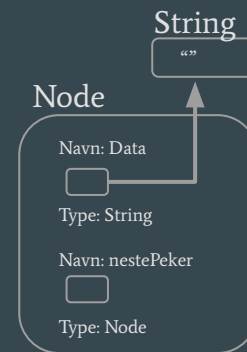
Lenkeliste



ArrayList eller Array f.eks.

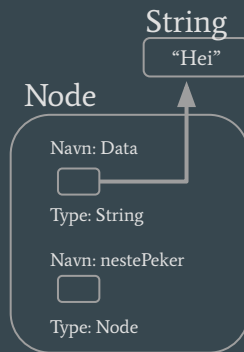
Node

```
2 //String versjon av Node:-  
3 class Node{-  
4   ··private String data;-  
5   ··private Node nestePeker;-  
6   -  
7   ··public Node(String data){-  
8     ····this.data = data;-  
9   ··}-  
10  ··public void settNestePeker(Node nestePeker){-  
11    ····this.nestePeker = nestePeker;-  
12  ··}-  
13  ··public Node hentNestePeker(){-  
14    ····return nestePeker;-  
15  ··}-  
16  ··public String hentData(){-  
17    ····return data;-  
18  ··}-  
19  }-
```



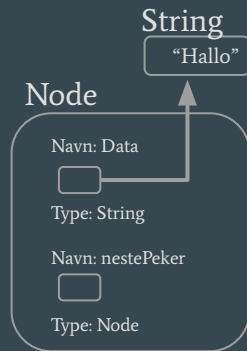
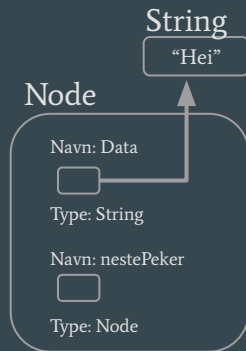
Lenkeliste

```
1 class Lenkeliste{  
2     ↵  
3     // Main for Node med data type String:↵  
4     public static void main(String[] args) {  
5     → Node node1 = new Node("Hei");  
6     Node node2 = new Node("Hallo");  
7     Node node3 = new Node("Hade");  
8     node1.settNestePeker(node2);  
9     node2.settNestePeker(node3);  
10    }  
↵
```



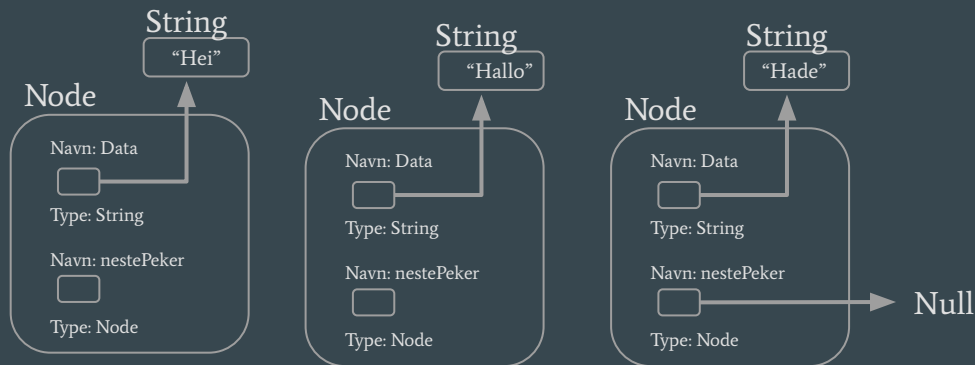
Lenkeliste

```
1 class Lenkeliste{
2
3   // Main for Node med data type String:
4   public static void main(String[] args) {
5       Node node1 = new Node("Hei");
6       Node node2 = new Node("Hallo");
7       Node node3 = new Node("Hade");
8       node1.settNestePeker(node2);
9       node2.settNestePeker(node3);
10  }
```



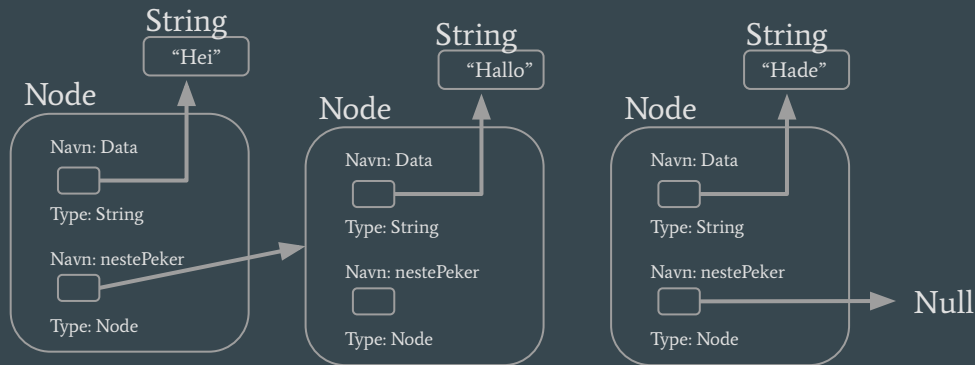
Lenkeliste

```
1 class Lenkeliste{
2
3     // Main for Node med data type String:
4     public static void main(String[] args) {
5         Node node1 = new Node("Hei");
6         Node node2 = new Node("Hallo");
7         Node node3 = new Node("Hade");
8         node1.settNestePeker(node2);
9         node2.settNestePeker(node3);
10    }
```



Lenkeliste

```
1 class Lenkeliste{  
2   ~  
3   // Main for Node med data type String:-  
4   public static void main(String[] args) {  
5     Node node1 = new Node("Hei");  
6     Node node2 = new Node("Hallo");  
7     Node node3 = new Node("Hade");  
8     node1.settNestePeker(node2);  
9     node2.settNestePeker(node3);  
10  }  
11 }  
12 }  
13 }  
14 }  
15 }  
16 }  
17 }  
18 }  
19 }
```

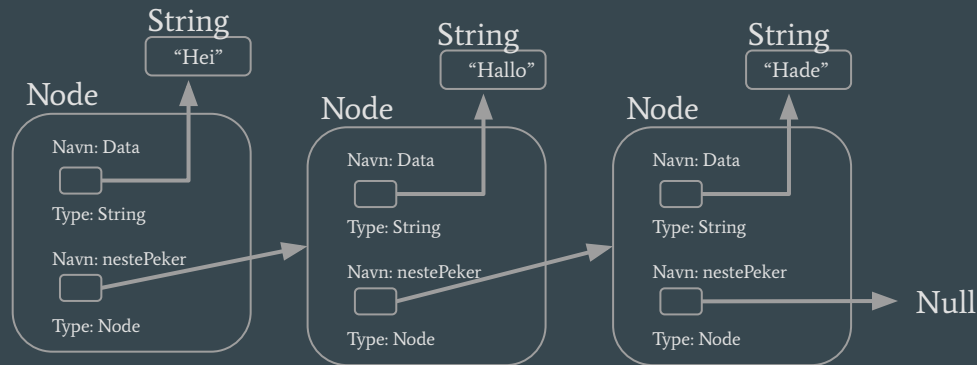


```
2 //String versjon av Node:-  
3 class Node{-  
4   private String data;  
5   private Node nestePeker;  
6   ~  
7   public Node(String data){  
8     this.data = data;  
9   }  
10  public void settNestePeker(Node nestePeker){  
11    this.nestePeker = nestePeker;  
12  }  
13  public Node hentNestePeker(){  
14    return nestePeker;  
15  }  
16  public String hentData(){  
17    return data;  
18  }  
19 }
```


Lenkeliste

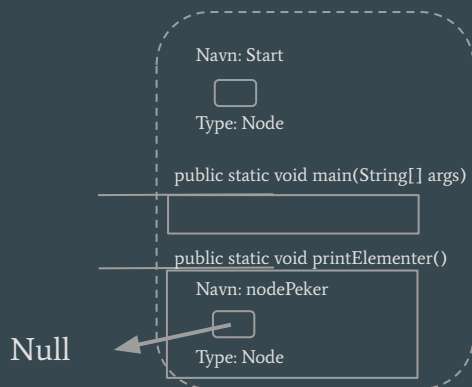
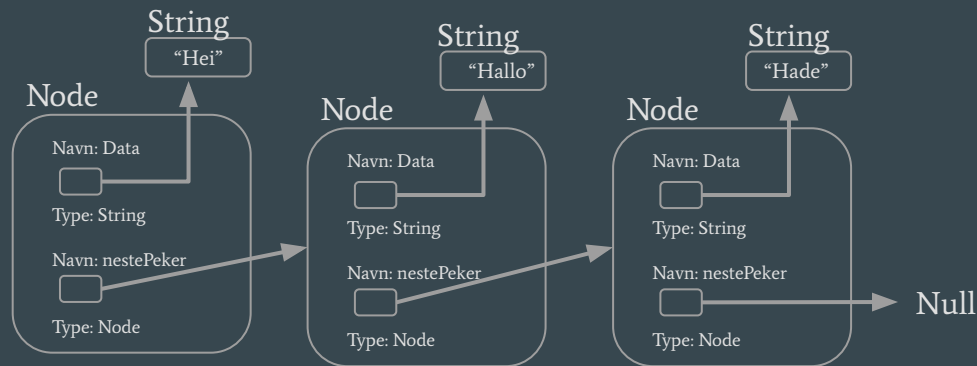
```
1 class Lenkeliste{
2
3 // Main for Node med data type String:-
4 public static void main(String[] args) {
5     Node node1 = new Node("Hei");
6     Node node2 = new Node("Hallo");
7     Node node3 = new Node("Hade");
8     node1.settNestePeker(node2);
9     node2.settNestePeker(node3);
10 }
```

```
2 //String versjon av Node:-
3 class Node{
4     private String data;
5     private Node nestePeker;
6
7     public Node(String data){
8         this.data = data;
9     }
10    public void settNestePeker(Node nestePeker){
11        this.nestePeker = nestePeker;
12    }
13    public Node hentNestePeker(){
14        return nestePeker;
15    }
16    public String hentData(){
17        return data;
18    }
19 }
```



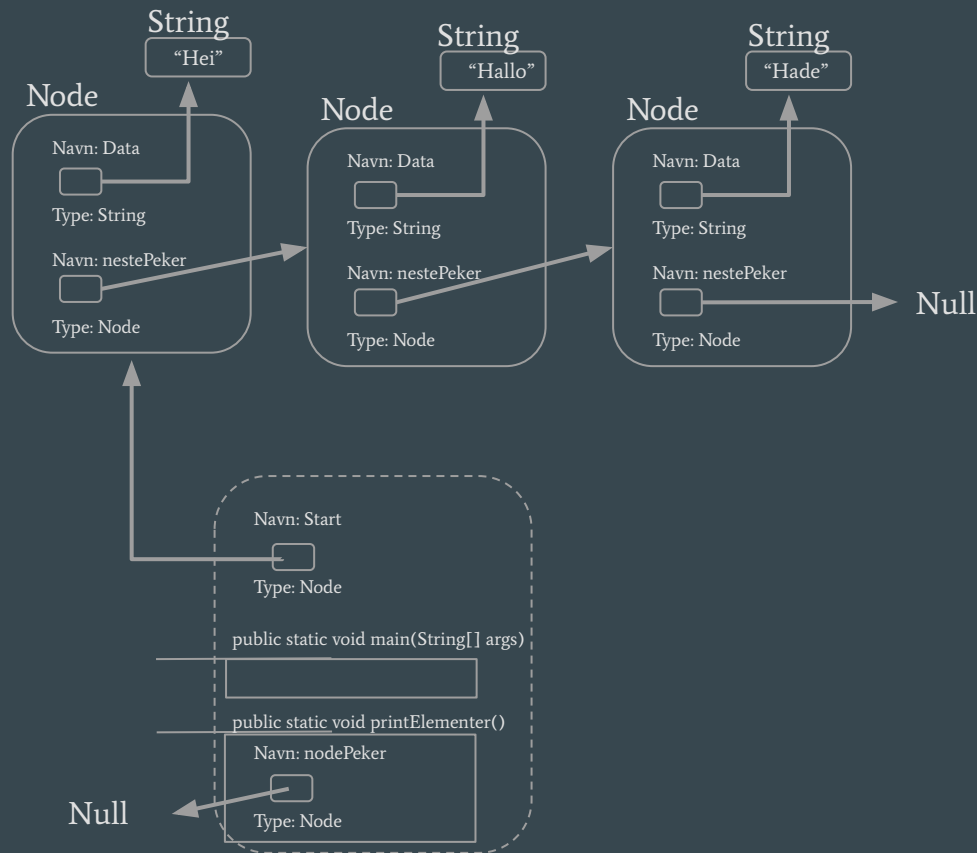
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        → node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



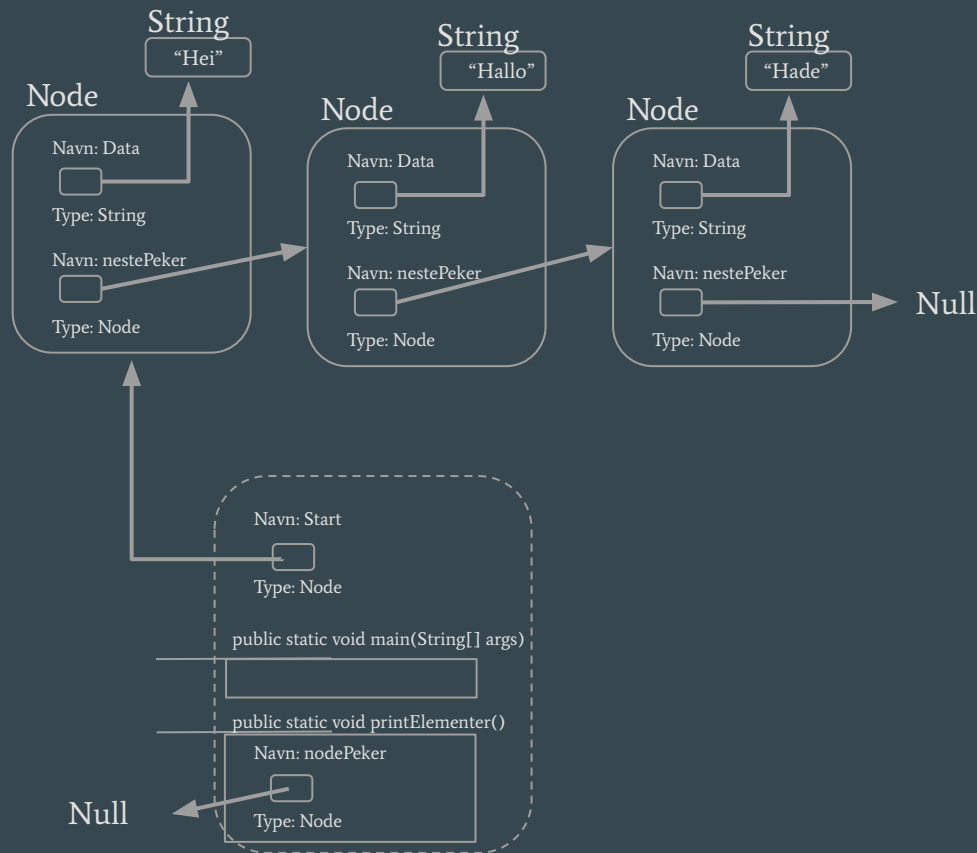
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2   public static Node start;
3
4   // Main for Node med data type String:
5   public static void main(String[] args) {
6     Node node1 = new Node("Hei");
7     Node node2 = new Node("Hallo");
8     Node node3 = new Node("Hade");
9     node1.settNestePeker(node2);
10    node2.settNestePeker(node3);
11    start = node1;
12
13    printElementer();
14  }
15
16  public static void printElementer(){
17    Node nodePeker = start;
18    while (nodePeker != null){
19      System.out.println(nodePeker.hentData());
20      nodePeker = nodePeker.hentNestePeker();
21    }
22  }
23 }
```



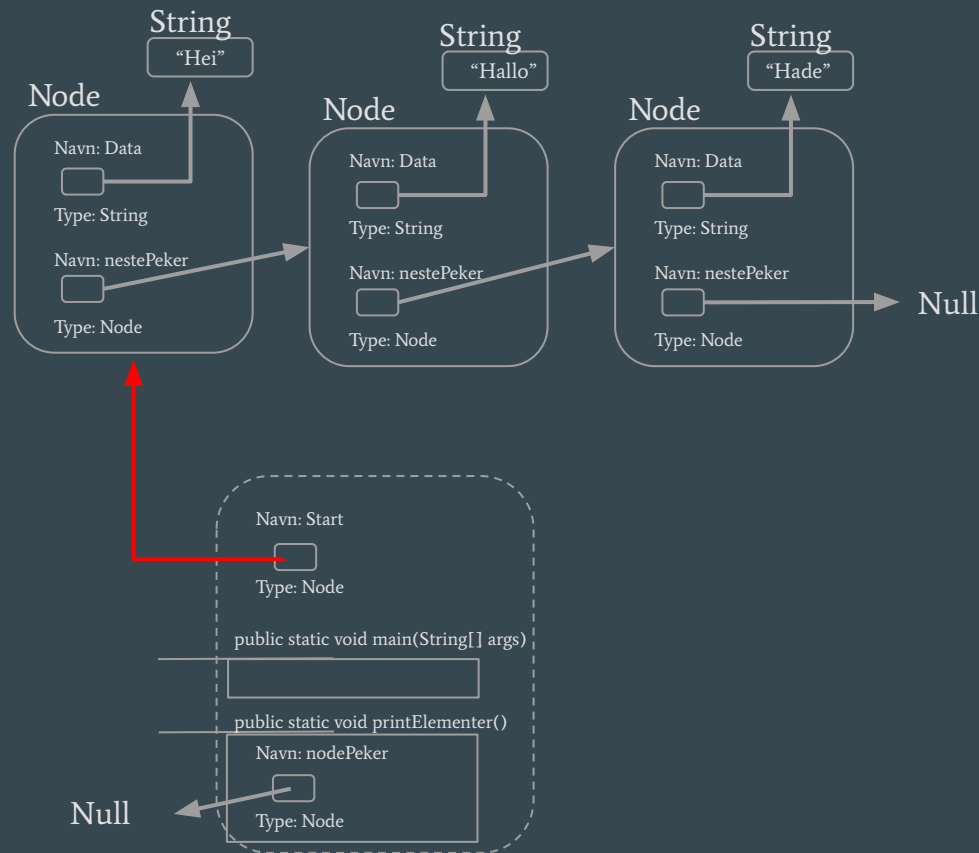
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



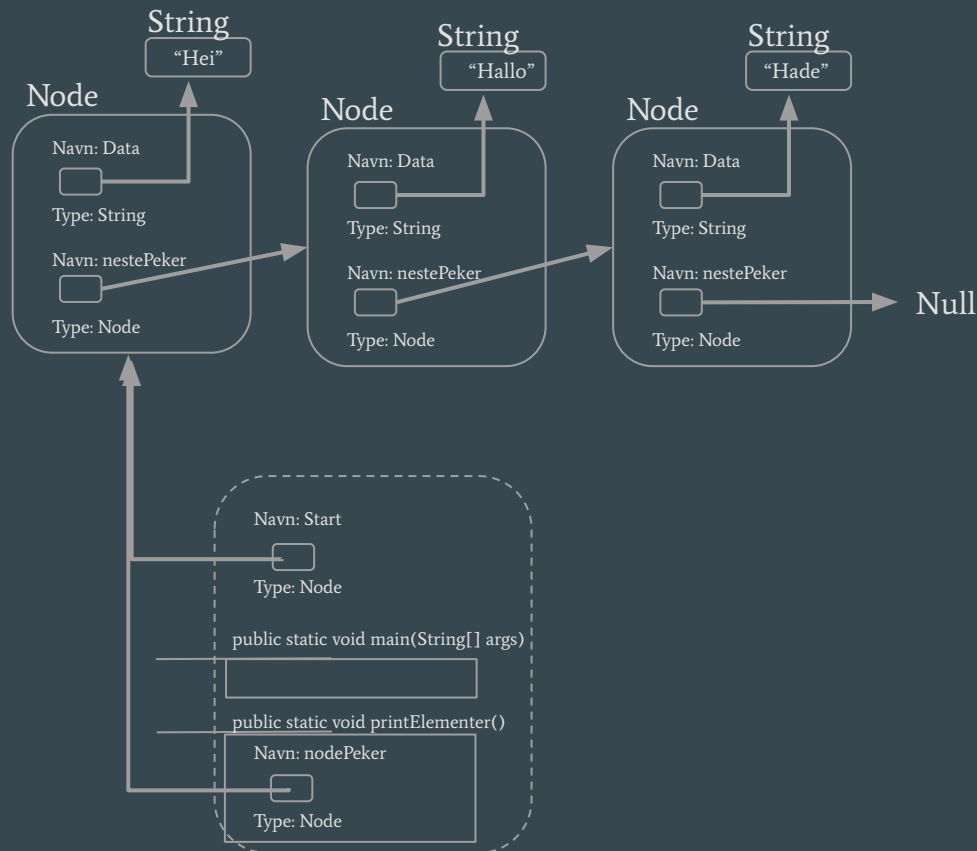
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



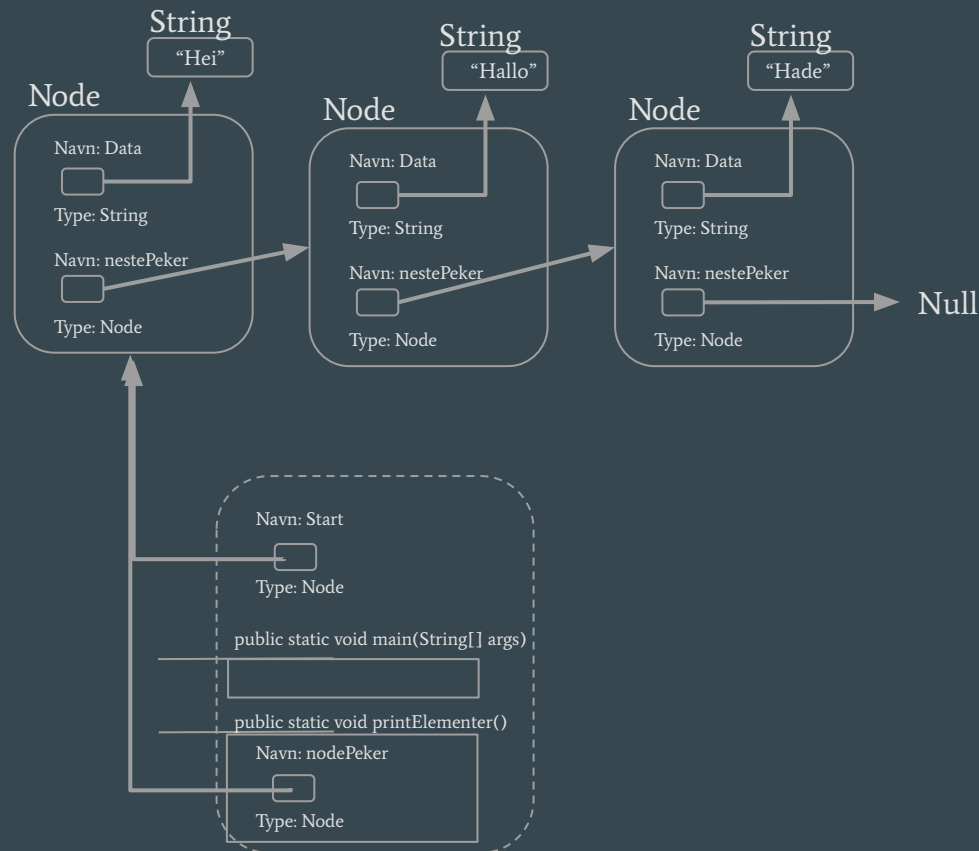
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



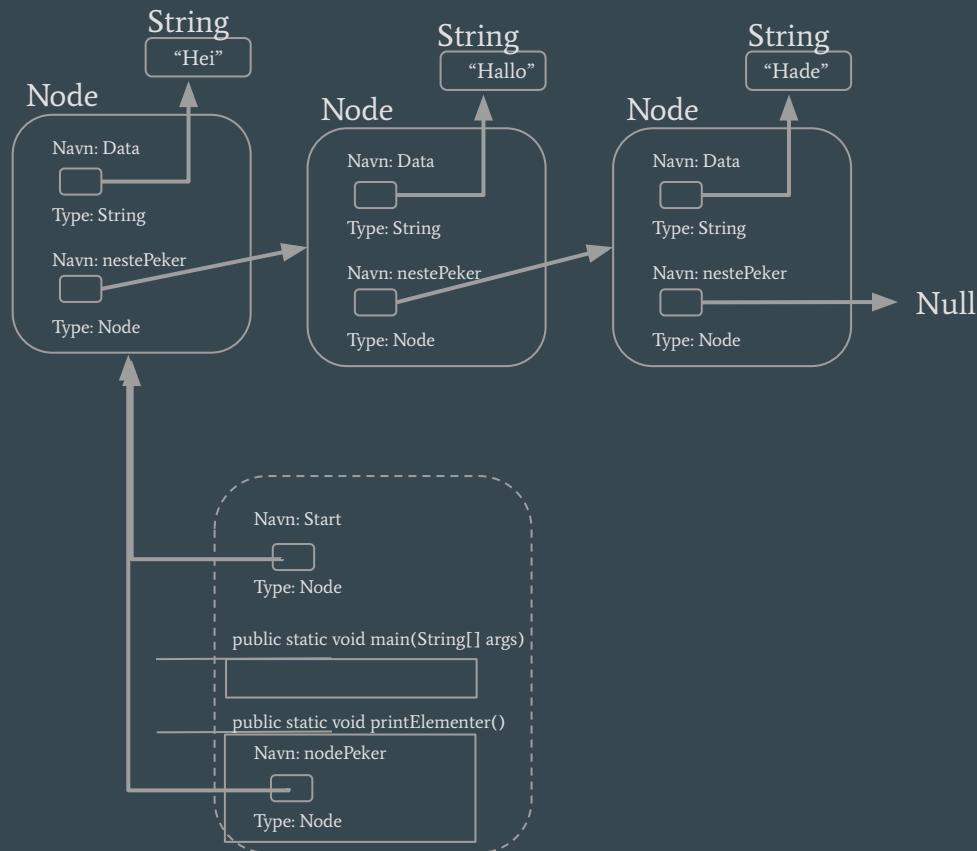
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



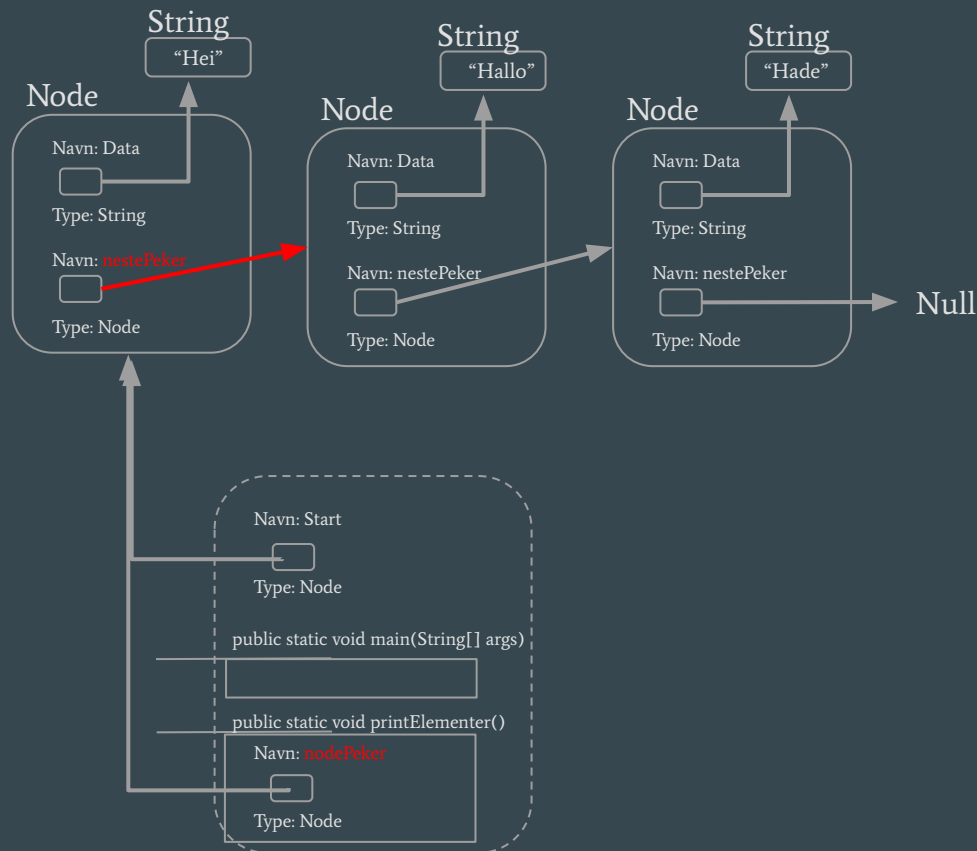
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



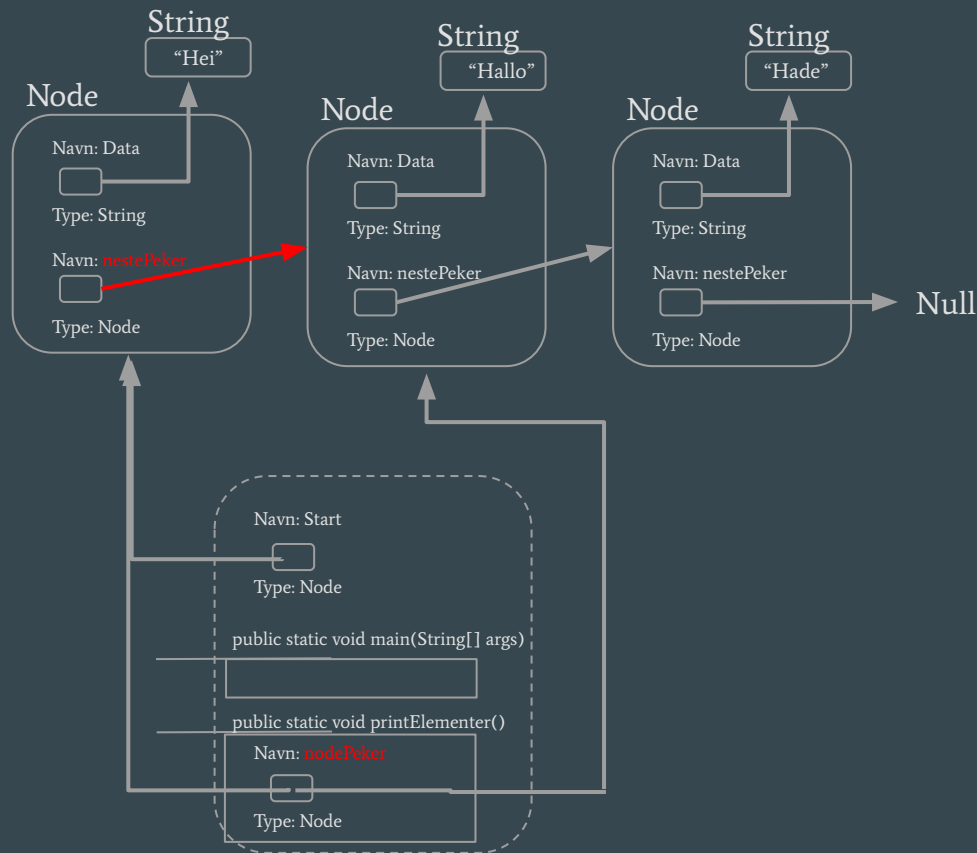
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



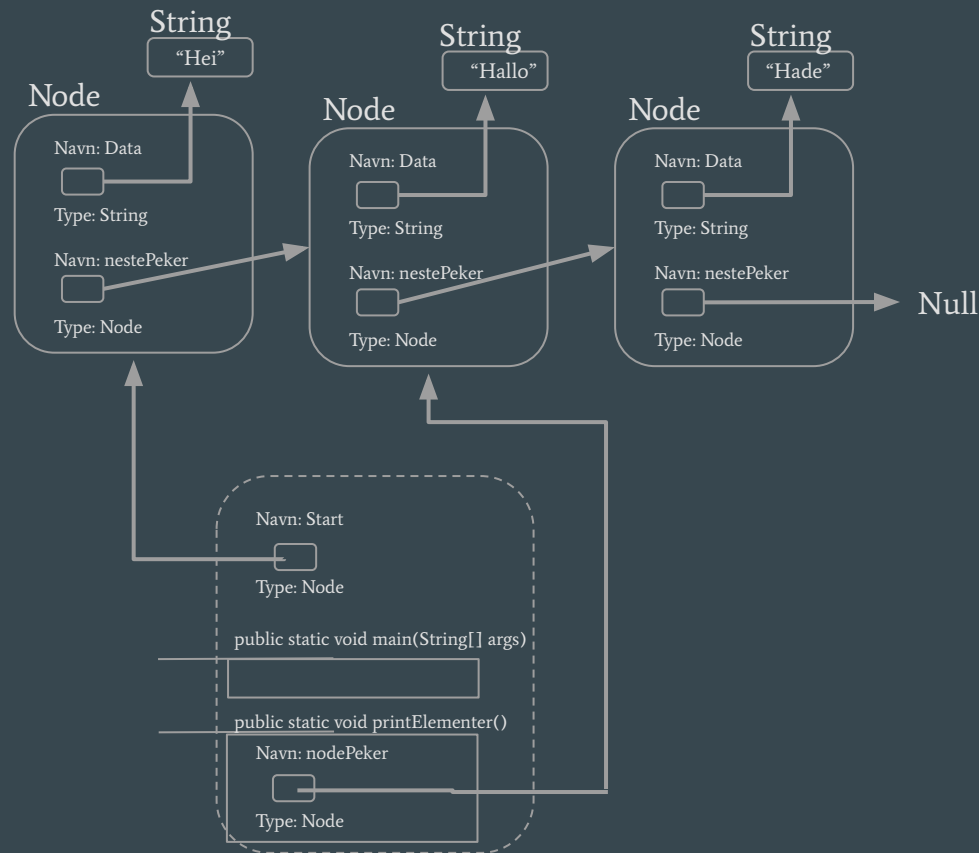
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



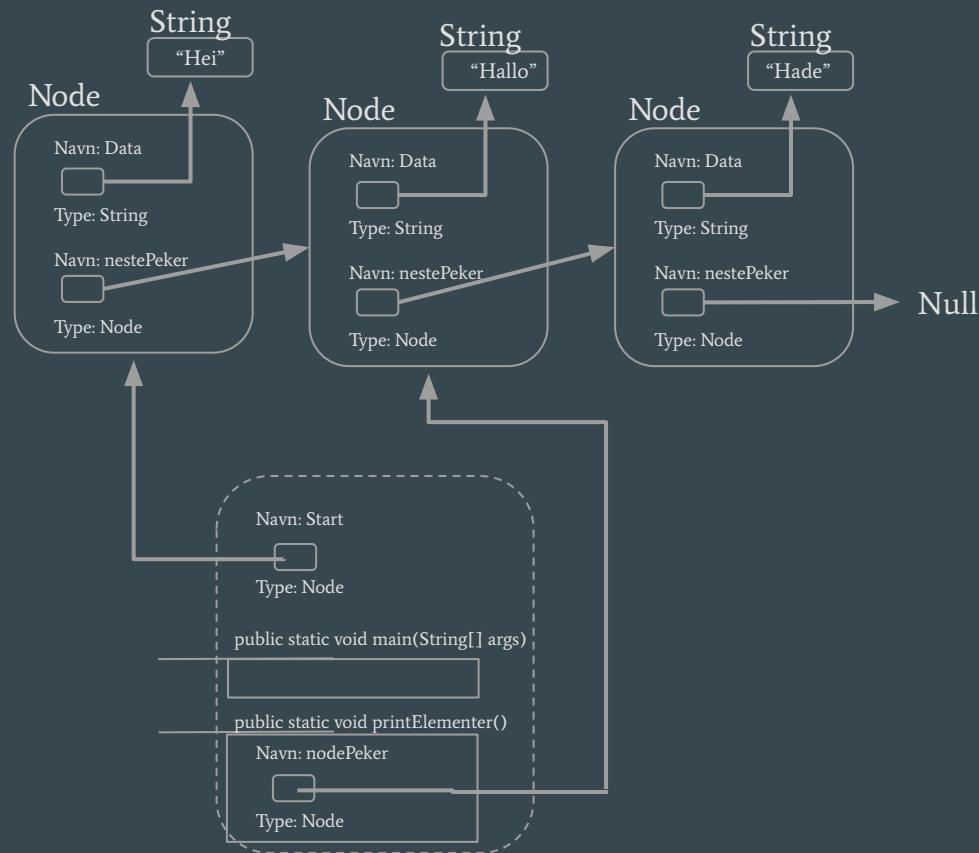
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



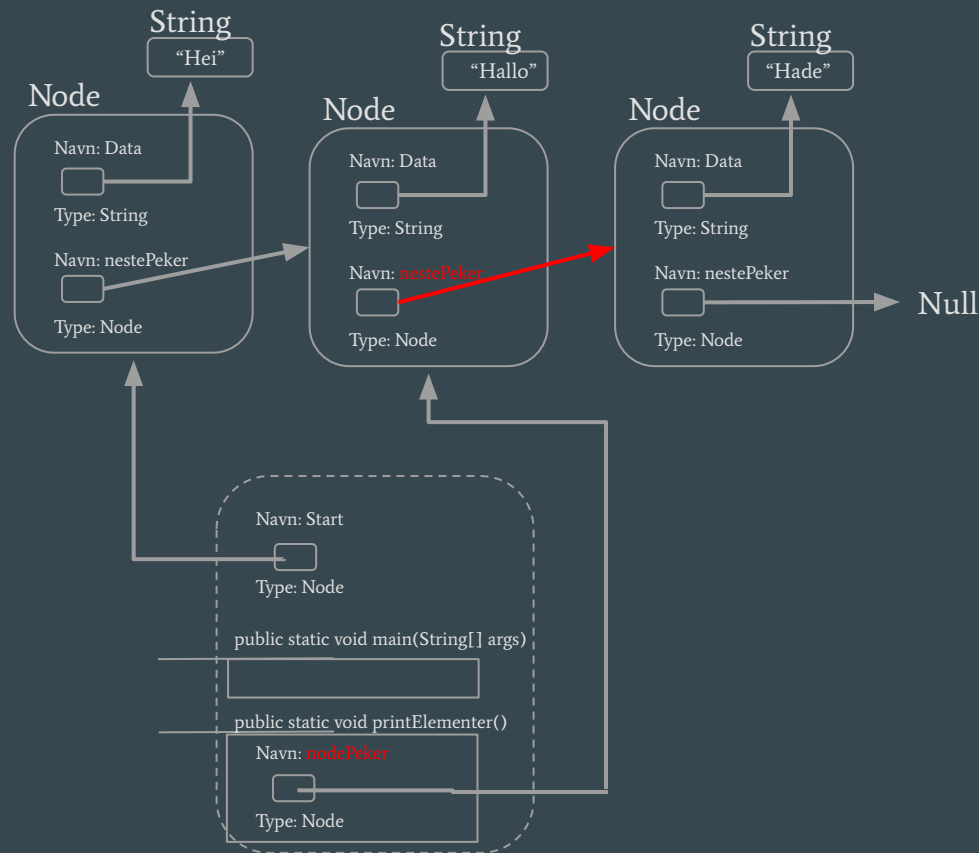
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



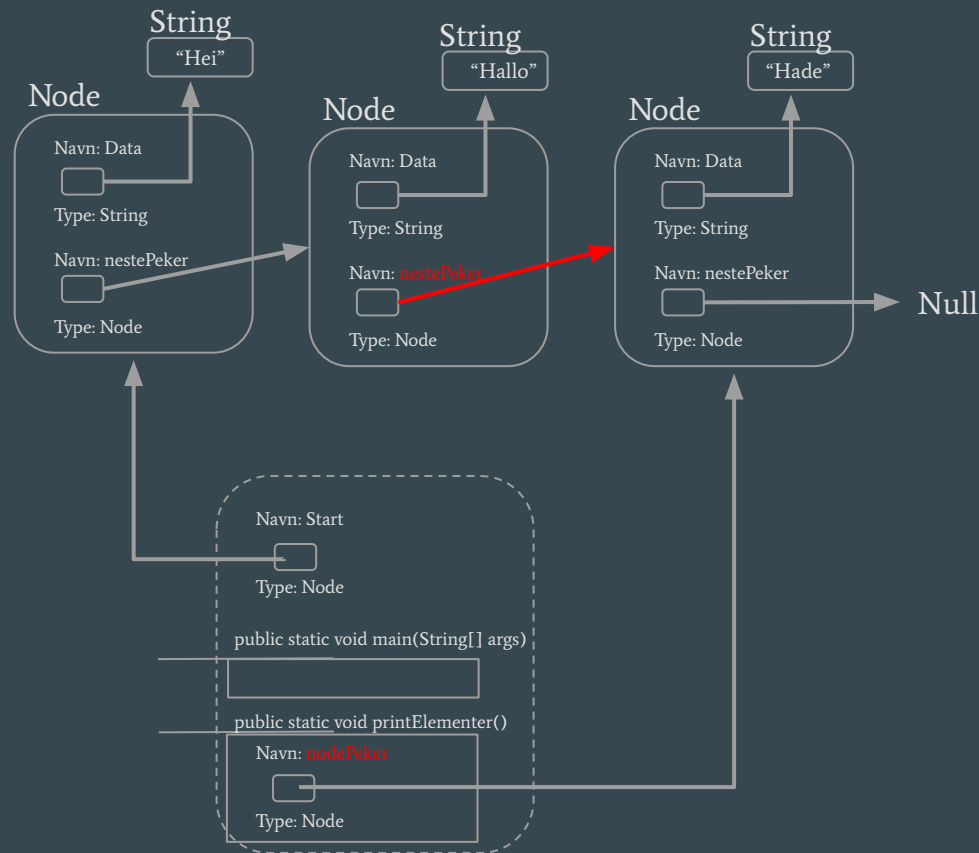
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



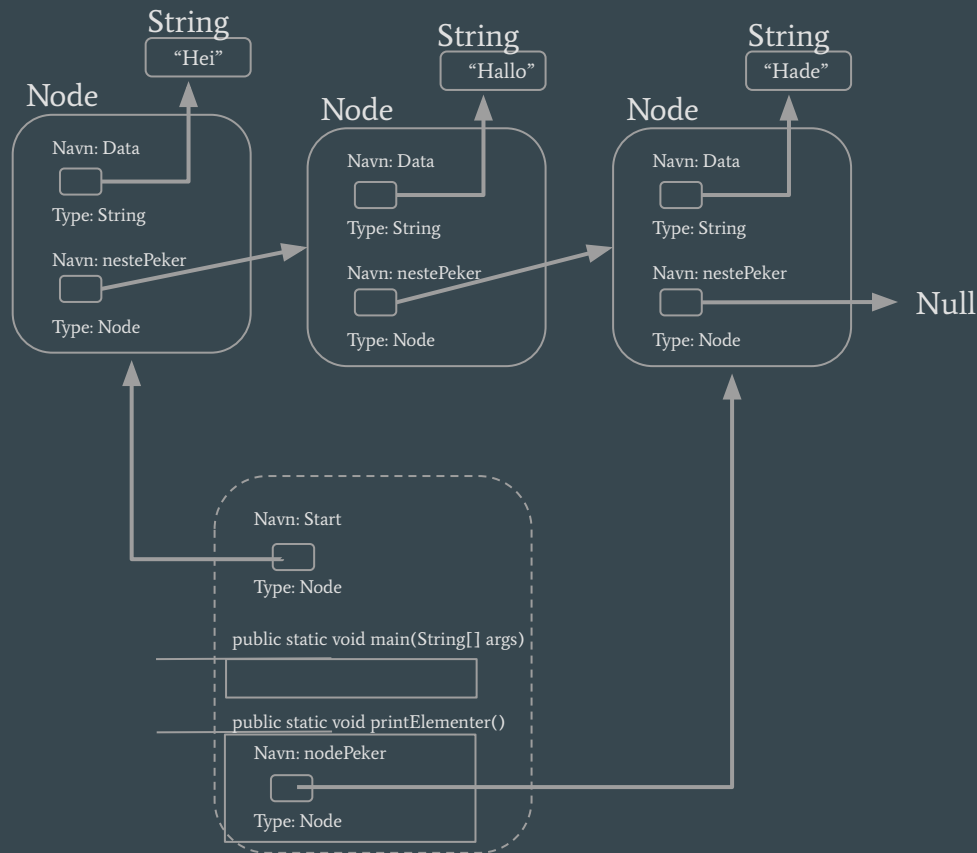
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



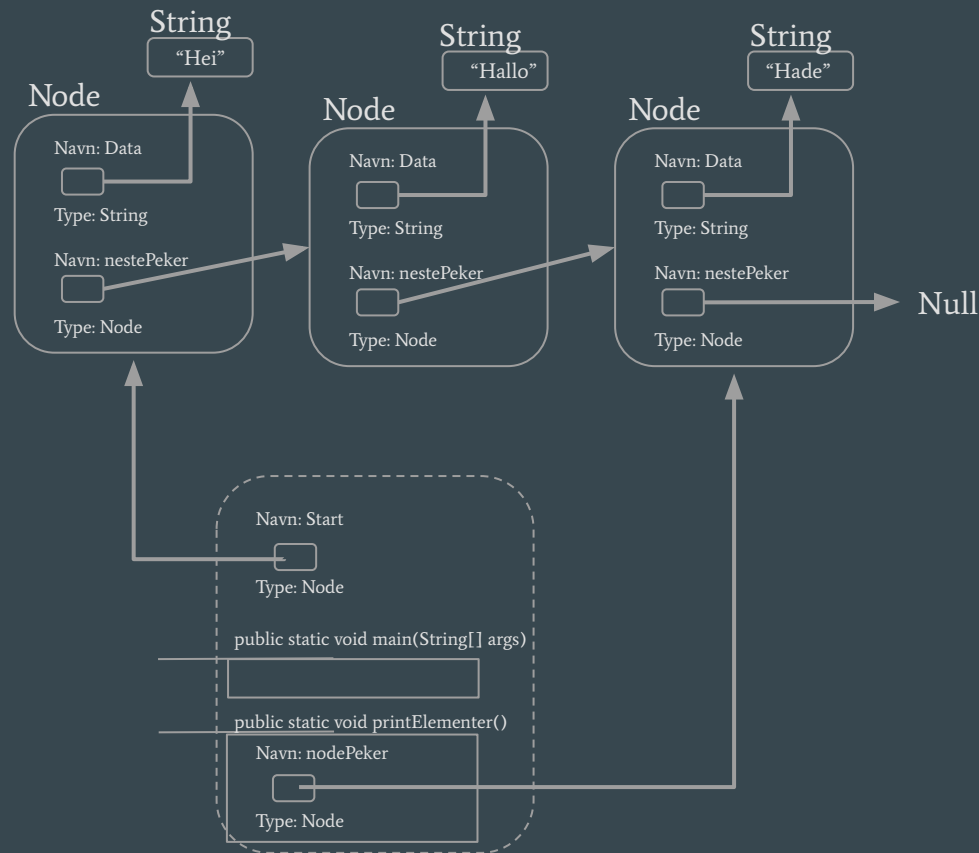
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



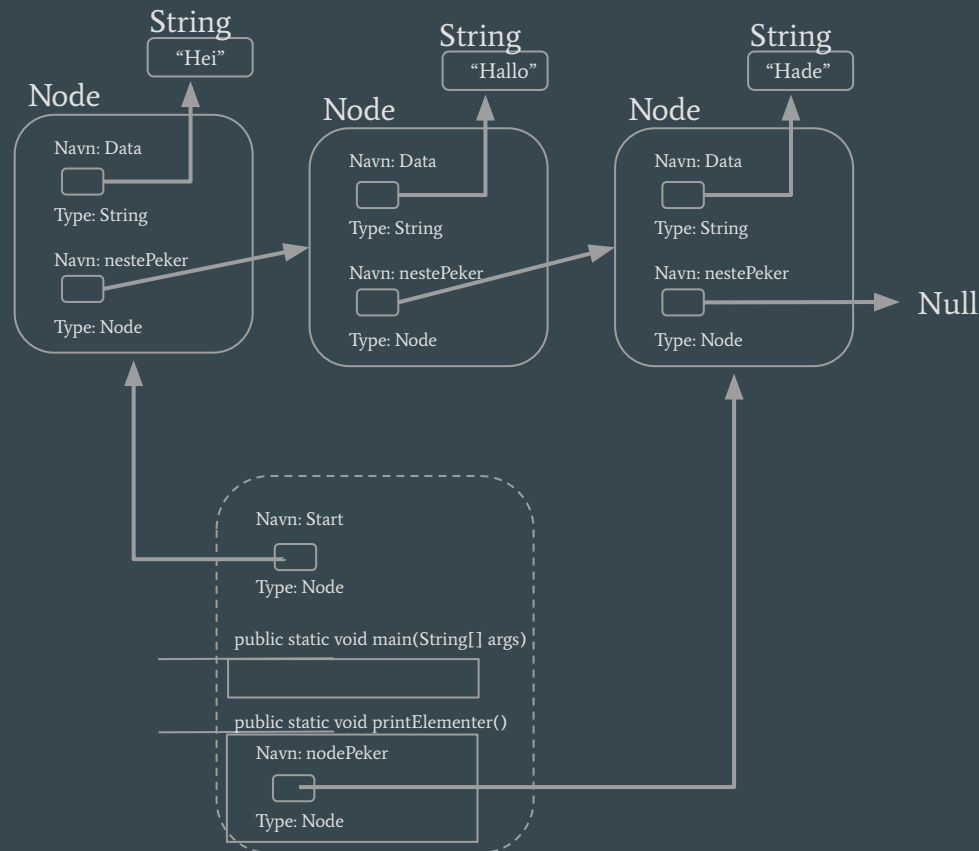
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



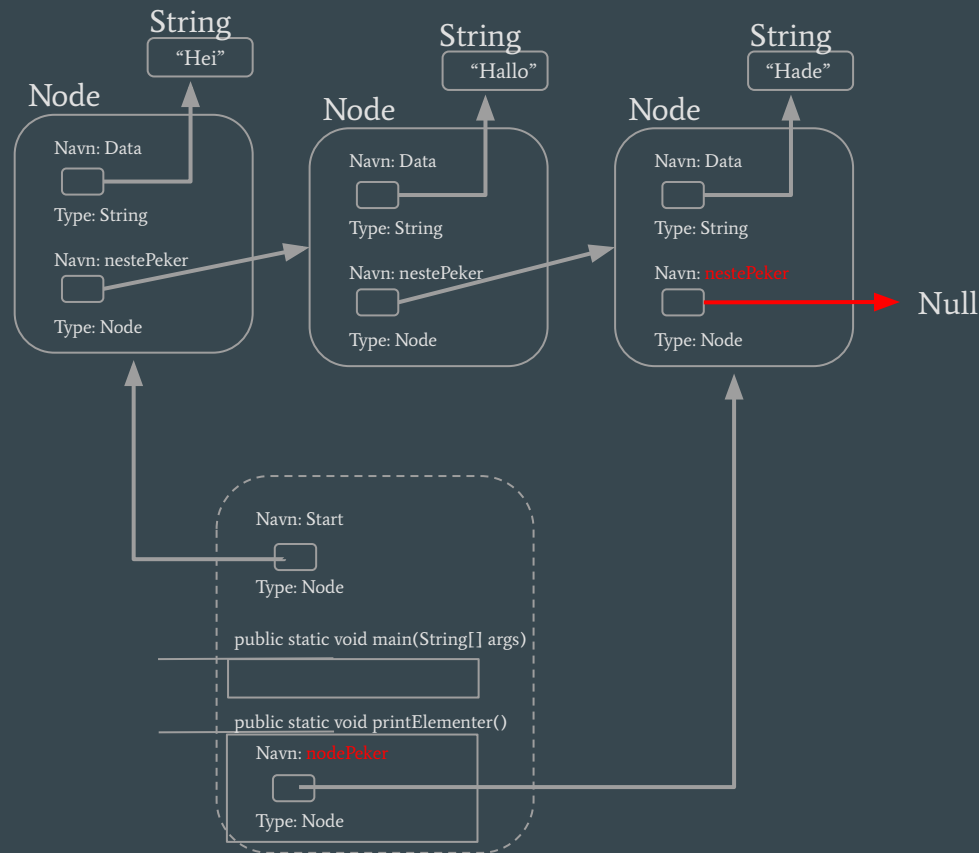
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



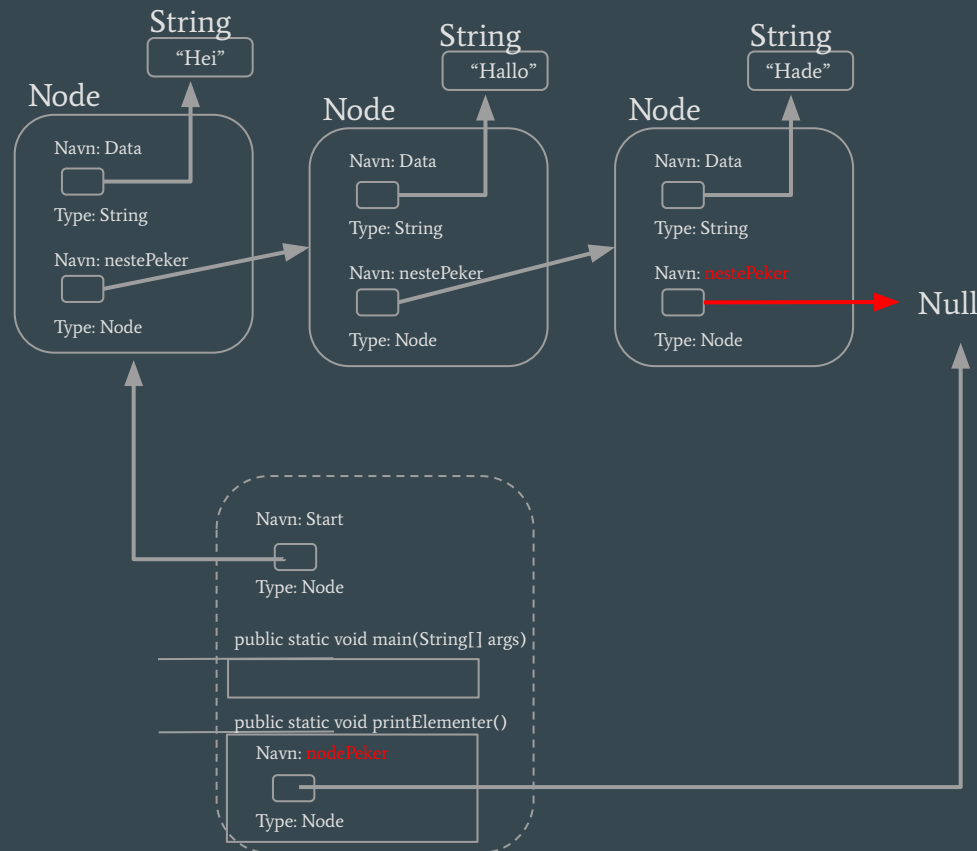
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



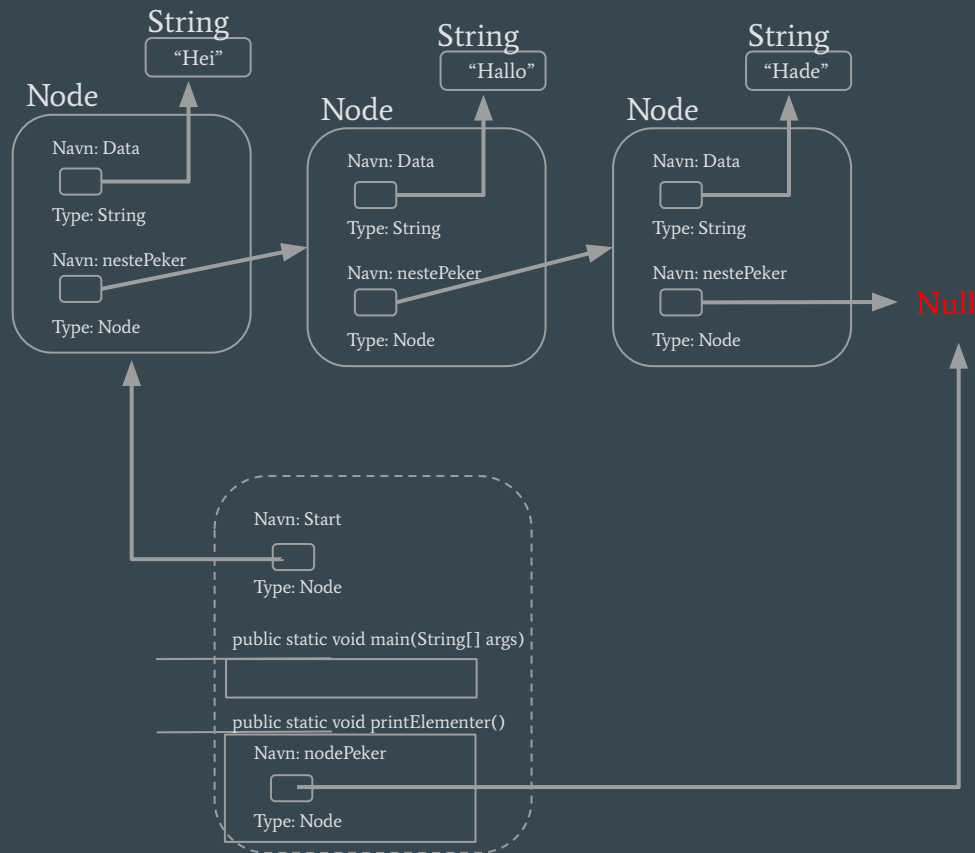
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



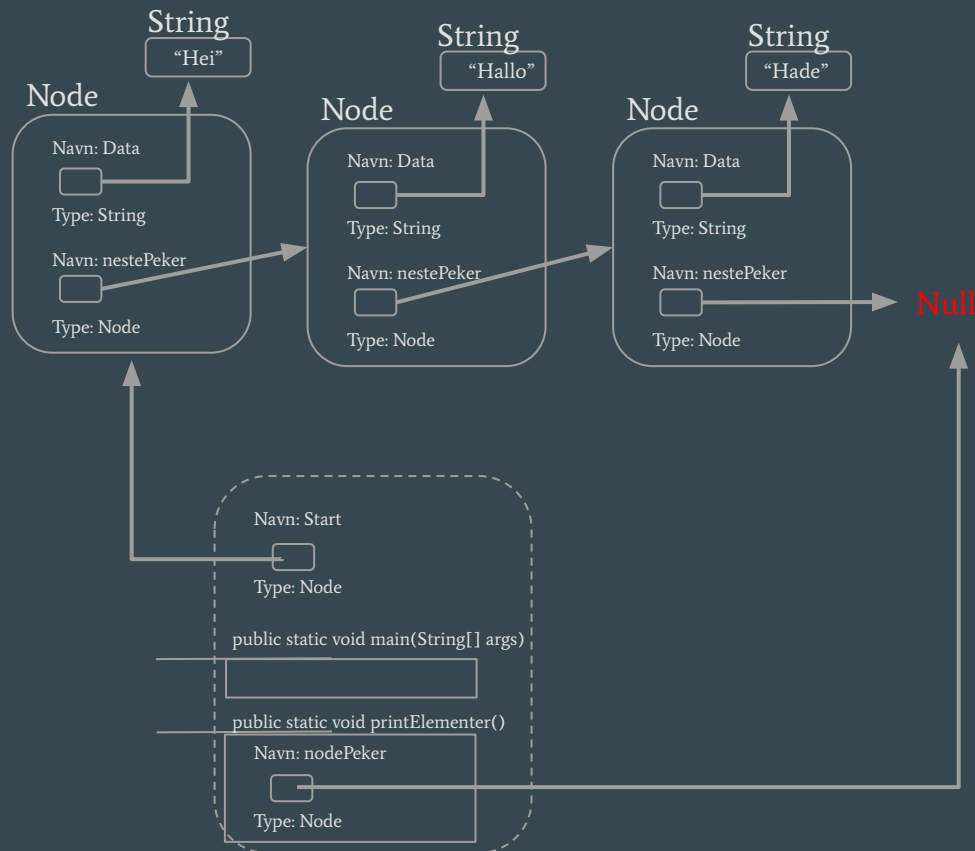
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



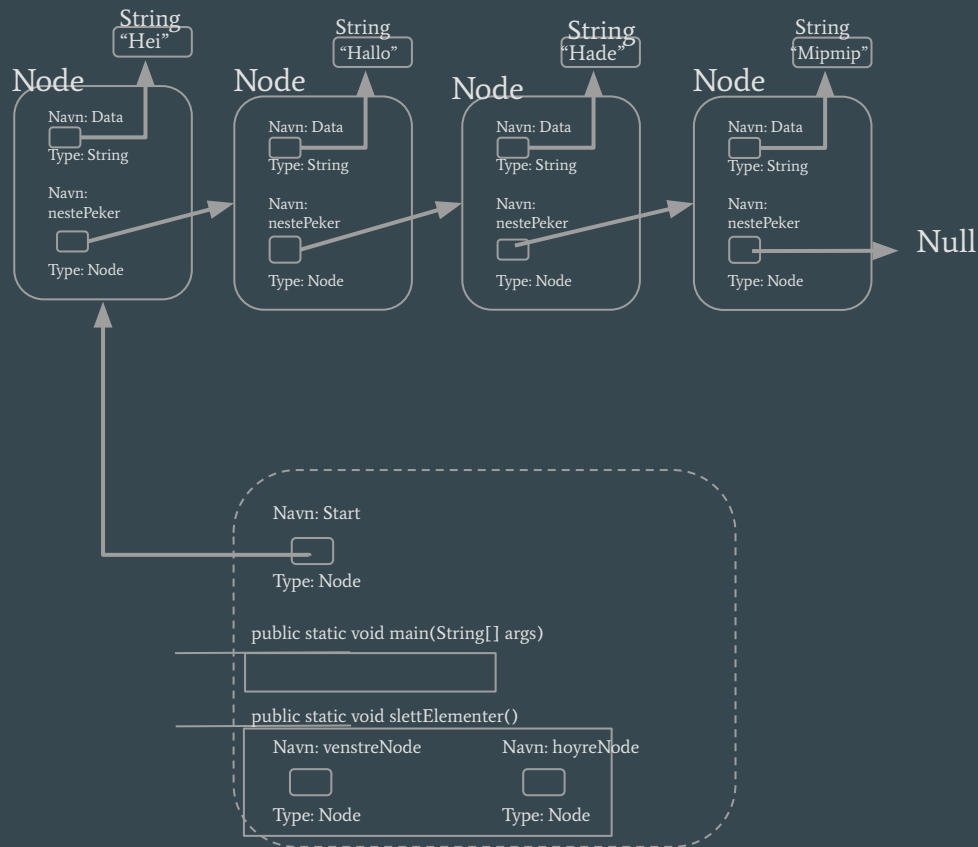
Iterere gjennom lenkeliste

```
1 class Lenkeliste{
2     public static Node start;
3
4     // Main for Node med data type String:
5     public static void main(String[] args) {
6         Node node1 = new Node("Hei");
7         Node node2 = new Node("Hallo");
8         Node node3 = new Node("Hade");
9         node1.settNestePeker(node2);
10        node2.settNestePeker(node3);
11        start = node1;
12
13        printElementer();
14    }
15
16    public static void printElementer(){
17        Node nodePeker = start;
18        while (nodePeker != null){
19            System.out.println(nodePeker.hentData());
20            nodePeker = nodePeker.hentNestePeker();
21        }
22    }
23 }
```



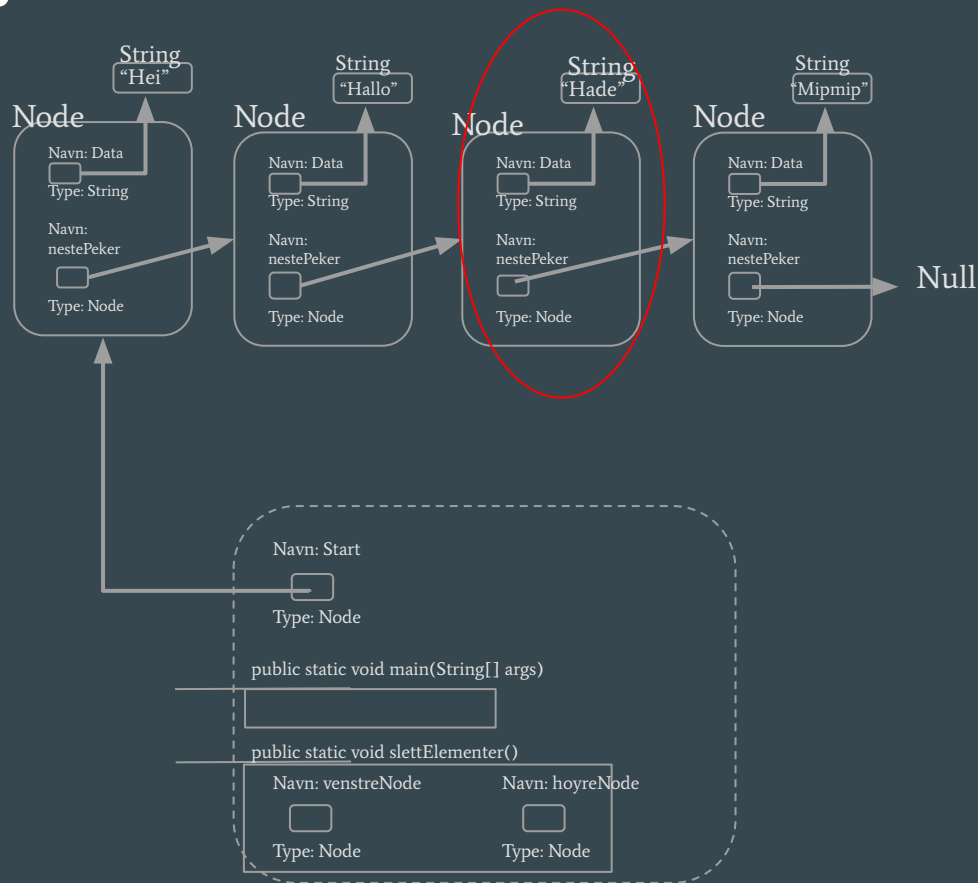
Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14    -
15    printElementer();-
16    slettElement();-
17    printElementer();-
18    }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23    }-
24
```



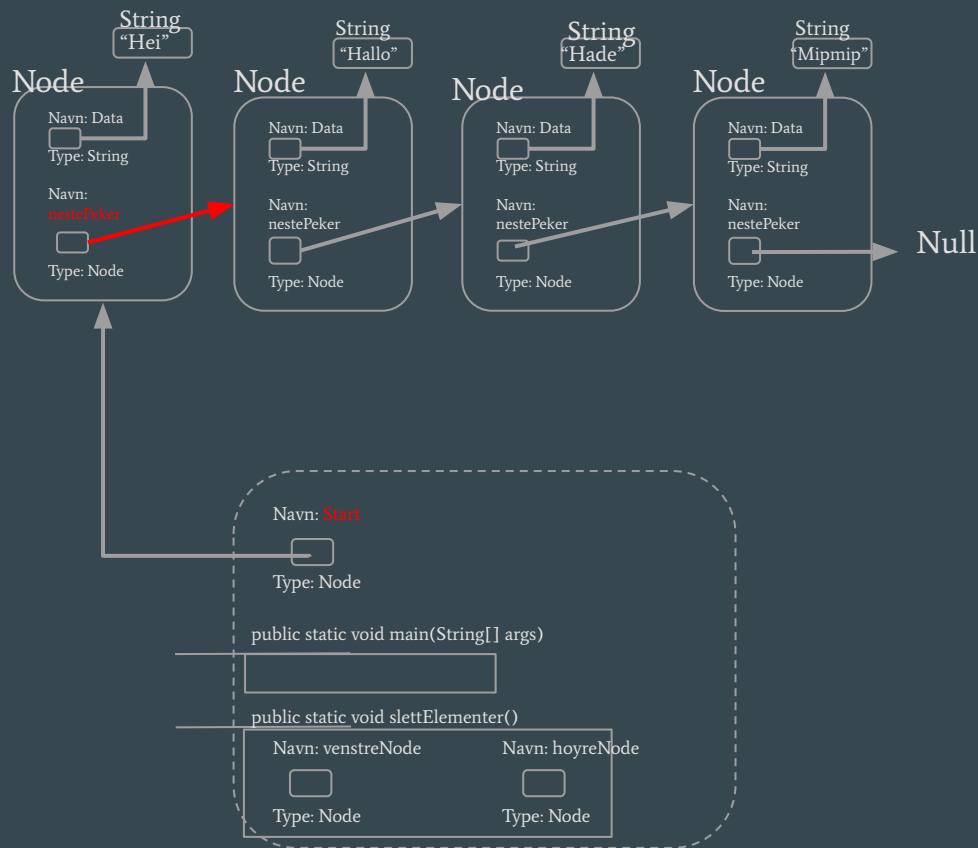
Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14    -
15    printElementer();-
16    slettElement();-
17    printElementer();-
18    }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23    }-
24
```



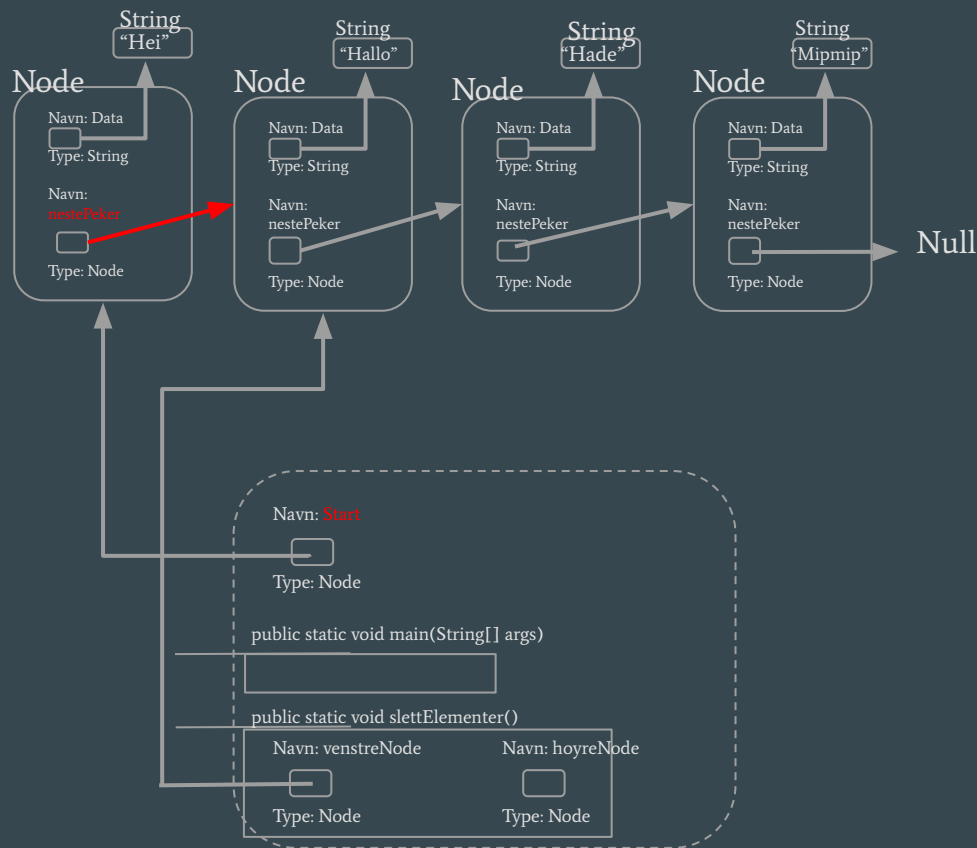
Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14    -
15    printElementer();-
16    slettElement();-
17    printElementer();-
18    }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23    }-
24
```



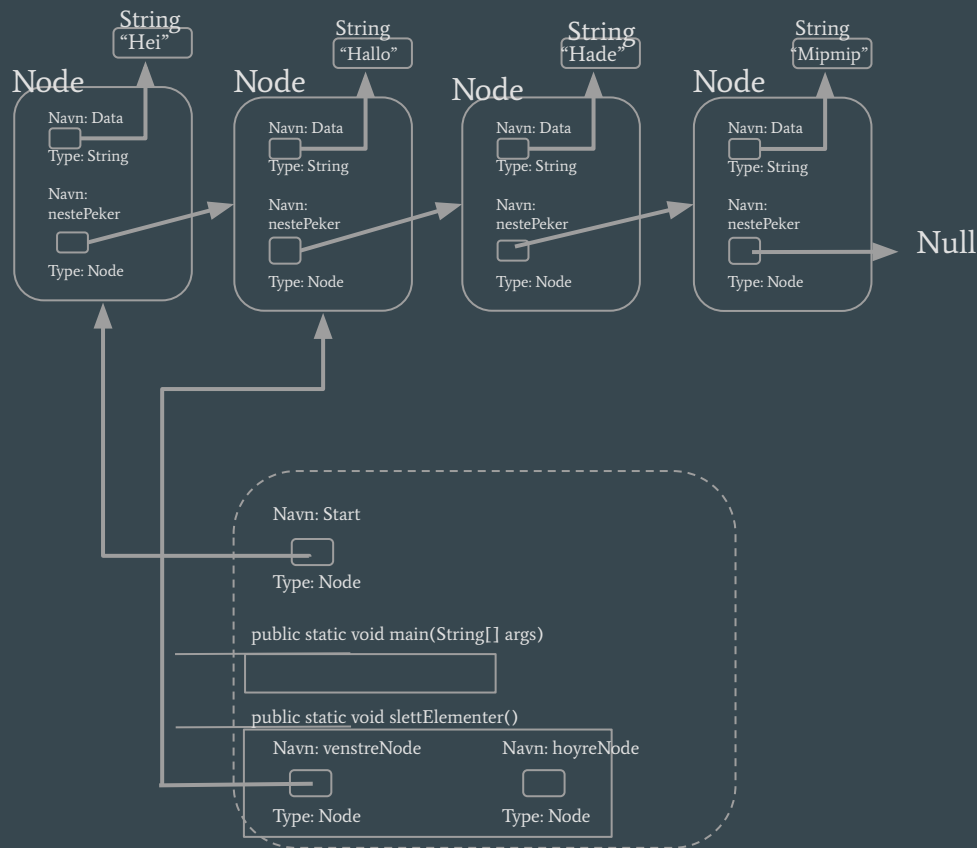
Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14 }-
15 printElementer();-
16 slettElement();-
17 printElementer();-
18 }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23 }-
24
```



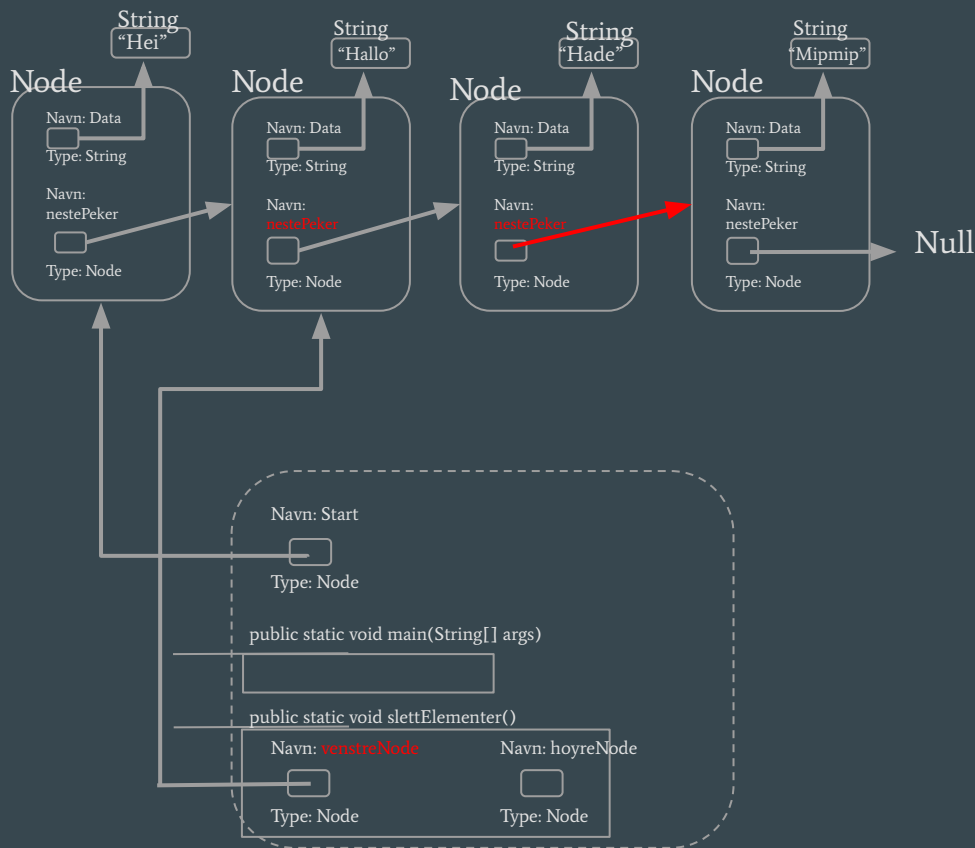
Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14 }-
15 printElementer();-
16 slettElement();-
17 printElementer();-
18 }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23 }-
24
```



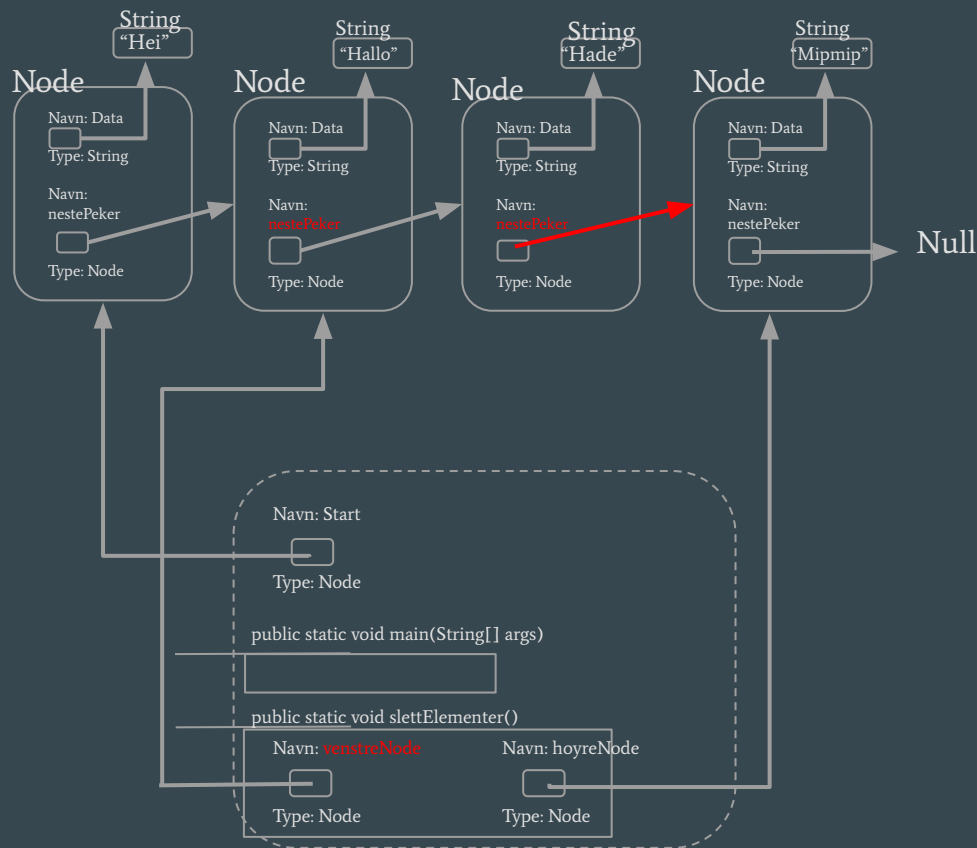
Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14 }-
15 printElementer();-
16 slettElement();-
17 printElementer();-
18 }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23 }-
24
```



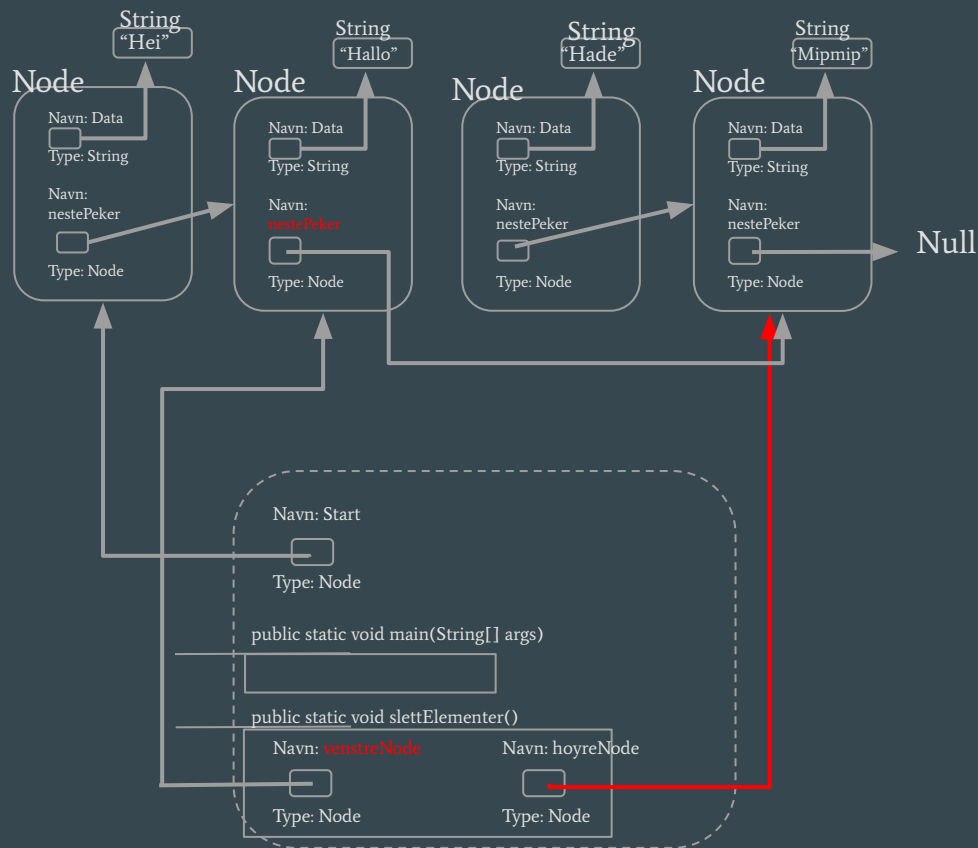
Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14    -
15    printElementer();-
16    slettElement();-
17    printElementer();-
18    }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23    }-
24
```



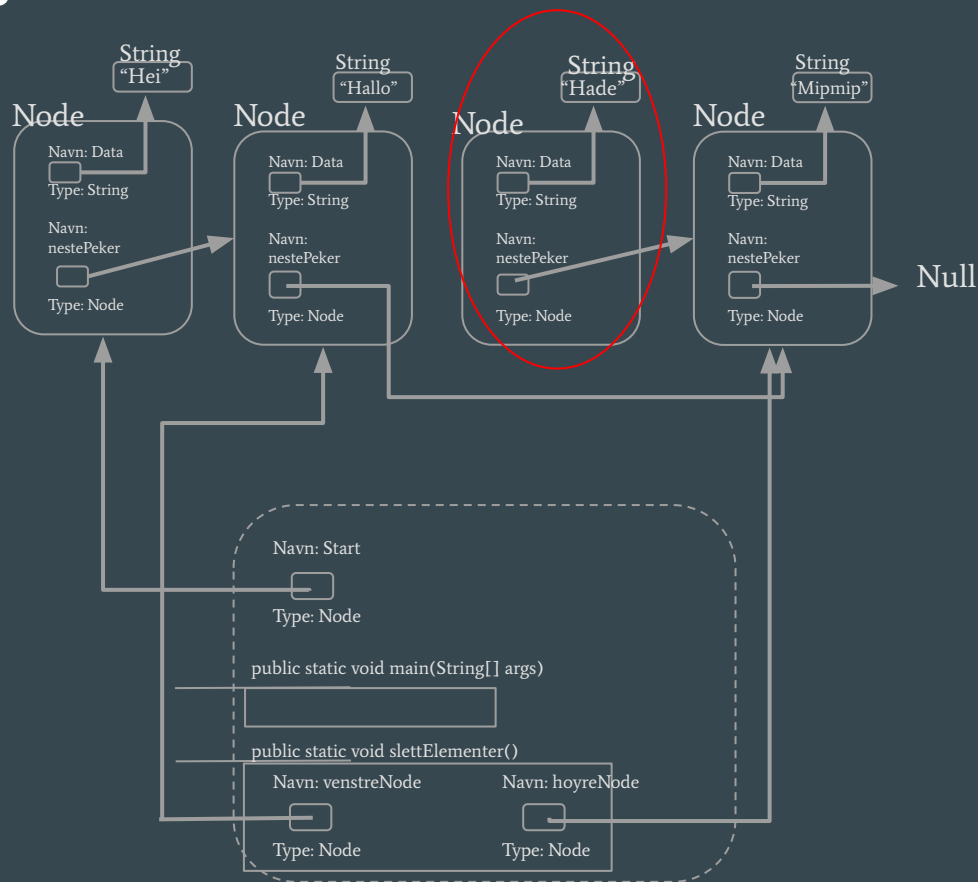
Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14 }-
15 printElementer();-
16 slettElement();-
17 printElementer();-
18 }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23 }-
24 }
```



Slette element i lenkeliste

```
4 // Main for Node med data type String:-
5 public static void main(String[] args) {-
6     Node node1 = new Node("Hei");-
7     Node node2 = new Node("Hallo");-
8     Node node3 = new Node("Hade");-
9     Node node4 = new Node("Mipmip");-
10    node1.settNestePeker(node2);-
11    node2.settNestePeker(node3);-
12    node3.settNestePeker(node4);-
13    start = node1;-
14    -
15    printElementer();-
16    slettElement();-
17    printElementer();-
18    }-
19 public static void slettElement() {-
20     Node venstreNode = start.hentNestePeker();-
21     Node hoyreNode = venstreNode.hentNestePeker();-
22     venstreNode.settNestePeker(hoyreNode);-
23     }-
24 }
```

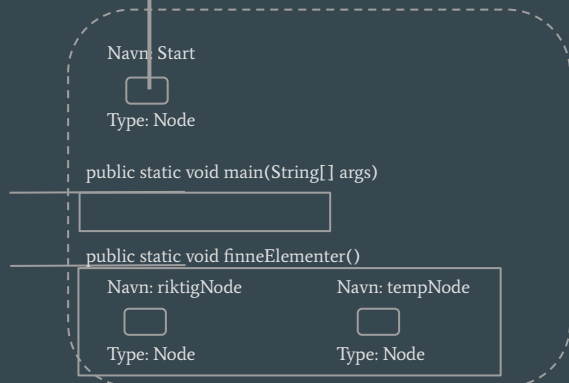
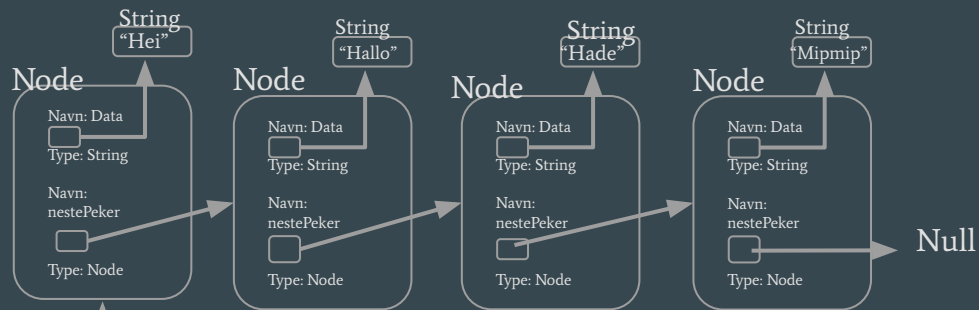


Generisk klasse

```
2 //String versjon av Node:-
3 class Node{-
4   private String data;-
5   private Node nestePeker;-
6   -
7   public Node(String data){-
8     ... this.data = data;-
9   }-
10  public void settNestePeker(Node nestePeker){-
11    ... this.nestePeker = nestePeker;-
12  }-
13  public Node hentNestePeker(){-
14    ... return nestePeker;-
15  }-
16  public String hentData(){-
17    ... return data;-
18  }-
19 }
```

```
21 //Generisk klasse Node:-
22 class Node<T>{-
23   private T data;-
24   private Node nestePeker;-
25   -
26   public Node(T data){-
27     ... this.data = data;-
28   }-
29   public void settNestePeker(Node nestePeker){-
30     ... this.nestePeker = nestePeker;-
31   }-
32   public Node hentNestePeker(){-
33     ... return nestePeker;-
34   }-
35   public T hentData(){-
36     ... return data;-
37   }-
38 }
```

Finne element i lenkeliste



Mentimeter neste uke

<https://www.menti.com/wpzy45bsao>

Jobbe selv

Jobb med hva dere vil og rekk opp hånda hvis dere trenger hjelp med noe/har spørsmål så møtes vi i breakoutroom! 😊

