

InLabP2 - LinkList

Generated by Doxygen 1.8.13



# Contents

<b>1</b>	<b>Class Index</b>	<b>1</b>
1.1	Class List . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	LinkedList< T > Class Template Reference . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.1.2	Member Function Documentation . . . . .	6
3.1.2.1	insertFront() . . . . .	6
3.1.2.2	insertRear() . . . . .	6
3.1.2.3	main() . . . . .	6
3.1.2.4	removeFront() . . . . .	7
3.1.2.5	removeRear() . . . . .	7
3.2	Node< T > Class Template Reference . . . . .	7
3.2.1	Detailed Description . . . . .	8
3.2.2	Member Function Documentation . . . . .	8
3.2.2.1	getNext() . . . . .	8
3.2.2.2	getPrev() . . . . .	8
3.2.2.3	getValue() . . . . .	8
3.2.2.4	setNext() . . . . .	8
3.2.2.5	setPrev() . . . . .	9
3.2.2.6	setValue() . . . . .	9
<b>4</b>	<b>File Documentation</b>	<b>11</b>
4.1	LinkedList.java File Reference . . . . .	11
4.1.1	Detailed Description . . . . .	11
	<b>Index</b>	<b>13</b>



# Chapter 1

## Class Index

### 1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">LinkedList&lt; T &gt;</a>		
A simple <a href="#">LinkedList</a> class	. . . . .	<a href="#">5</a>
<a href="#">Node&lt; T &gt;</a>		
<a href="#">Node</a> class for <a href="#">LinkedList</a>	. . . . .	<a href="#">7</a>



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">LinkedList.java</a>	
This file contains both the <a href="#">LinkedList</a> class and a <a href="#">Node</a> class	11





## Chapter 3

# Class Documentation

### 3.1 `LinkedList< T >` Class Template Reference

A simple `LinkedList` class.

#### Public Member Functions

- void `insertFront` (T item)  
*Insert an item to the front of `LinkedList`.*
- void `insertRear` (T item)  
*Insert an item to the rear of `LinkedList`.*
- void `removeFront` ()  
*Removes an item (if present) from the front of `LinkedList`.*
- void `removeRear` ()  
*Removes an item (if present) from the rear of `LinkedList`.*

#### Static Public Member Functions

- static void `main` (String a[])  
*Driver class to demonstrate the `LinkedList` class.*

#### Private Attributes

- `Node< T >` `front`
- `Node< T >` `rear`

#### 3.1.1 Detailed Description

A simple `LinkedList` class.

#### Author

Shivam Sood

This is a generic class `LinkedList` class. It contains methods to manage the list. The class also contains a driver function called `main`.

### 3.1.2 Member Function Documentation

#### 3.1.2.1 insertFront()

```
void LinkList< T >.insertFront (
    T item )
```

Insert an item to the front of [LinkList](#).

##### Parameters

<i>item</i>	This item is added to the front.
-------------	----------------------------------

##### Returns

void

#### 3.1.2.2 insertRear()

```
void LinkList< T >.insertRear (
    T item )
```

Insert an item to the rear of [LinkList](#).

##### Parameters

<i>item</i>	This item is added to the rear.
-------------	---------------------------------

##### Returns

void

#### 3.1.2.3 main()

```
static void LinkList< T >.main (
    String a[] ) [static]
```

Driver class to demonstrate the [LinkList](#) class.

##### Parameters

<i>a[]</i>	This can be used to pass arguments.
------------	-------------------------------------

**Returns**

void

**3.1.2.4 removeFront()**

```
void LinkedList< T >.removeFront ( )
```

Removes an item (if present) from the front of [LinkedList](#).

**Returns**

void

**3.1.2.5 removeRear()**

```
void LinkedList< T >.removeRear ( )
```

Removes an item (if present) from the rear of [LinkedList](#).

**Returns**

void

The documentation for this class was generated from the following file:

- [LinkedList.java](#)

## 3.2 Node< T > Class Template Reference

[Node](#) class for [LinkedList](#).

**Public Member Functions**

- [Node< T > getPrev \(\)](#)  
*Getter for prev.*
- void [setPrev](#) ([Node< T > prev](#))  
*Setter for prev.*
- [Node< T > getNext \(\)](#)  
*Getter for prev.*
- void [setNext](#) ([Node< T > next](#))  
*Setter for next.*
- T [getValue \(\)](#)  
*Getter for value.*
- void [setValue](#) (T value)  
*Setter for next.*

## Private Attributes

- `Node< T > prev`
- `Node< T > next`
- `T value`

### 3.2.1 Detailed Description

`Node` class for `LinkedList`.

Author

Shivam Sood

This is a generic `Node` class. It contains methods contains getter and setters for `prev` and `next`.

### 3.2.2 Member Function Documentation

#### 3.2.2.1 getNext()

```
Node<T> Node< T >.getNext ( )
```

Getter for `prev`.

Returns

`Node<t>`

#### 3.2.2.2 getPrev()

```
Node<T> Node< T >.getPrev ( )
```

Getter for `prev`.

Returns

`Node<t>`

#### 3.2.2.3 getValue()

```
T Node< T >.getValue ( )
```

Getter for `value`.

Returns

`T`

#### 3.2.2.4 setNext()

```
void Node< T >.setNext (
    Node< T > next )
```

Setter for `next`.

**Parameters**

<i>next</i>	This param is to be set as next.
-------------	----------------------------------

**Returns**

void

**3.2.2.5 setPrev()**

```
void Node< T >.setPrev (
    Node< T > prev )
```

Setter for prev.

**Parameters**

<i>prev</i>	This param is to be set as prev.
-------------	----------------------------------

**Returns**

Node<t>

**3.2.2.6 setValue()**

```
void Node< T >.setValue (
    T value )
```

Setter for next.

**Parameters**

<i>value</i>	This param is to be set as value.
--------------	-----------------------------------

**Returns**

void

The documentation for this class was generated from the following file:

- [LinkedList.java](#)



## Chapter 4

# File Documentation

### 4.1 LinkedList.java File Reference

This file contains both the [LinkedList](#) class and a [Node](#) class.

#### Classes

- class [LinkedList< T >](#)  
*A simple [LinkedList](#) class.*
- class [Node< T >](#)  
*[Node](#) class for [LinkedList](#).*

#### 4.1.1 Detailed Description

This file contains both the [LinkedList](#) class and a [Node](#) class.

#### Author

Shivam Sood [ssood@cse.iitb.ac.in](mailto:ssood@cse.iitb.ac.in)

#### Version

1.0





# Index

- getNext
  - Node, [8](#)
- getPrev
  - Node, [8](#)
- getValue
  - Node, [8](#)
- insertFront
  - LinkedList, [6](#)
- insertRear
  - LinkedList, [6](#)
- LinkedList
  - insertFront, [6](#)
  - insertRear, [6](#)
  - main, [6](#)
  - removeFront, [7](#)
  - removeRear, [7](#)
- LinkedList< T >, [5](#)
- LinkedList.java, [11](#)
- main
  - LinkedList, [6](#)
- Node
  - getNext, [8](#)
  - getPrev, [8](#)
  - getValue, [8](#)
  - setNext, [8](#)
  - setPrev, [9](#)
  - setValue, [9](#)
- Node< T >, [7](#)
- removeFront
  - LinkedList, [7](#)
- removeRear
  - LinkedList, [7](#)
- setNext
  - Node, [8](#)
- setPrev
  - Node, [9](#)
- setValue
  - Node, [9](#)