File Edit View Help

File Edit View Help

# Linking Structures

# Linking Structures

- Groups Together
  - Data
  - Link(s) / Reference(s) / Pointer(s)
  - "Node"
- Pros
  - Growable
  - Shrinkable
- Cons
  - No Random Access

## Node

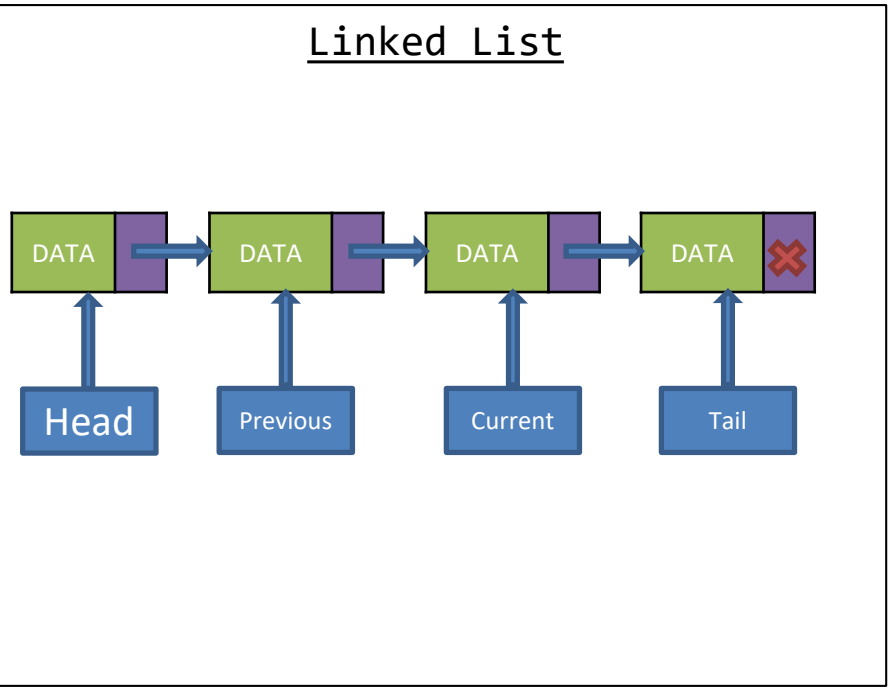

## List of Nodes

File Edit View Help

# Linked Lists

# Linked Lists

- Nodes Contain
  - Data
  - Link
- Special Nodes
  - Head: Always points to the first element of the list
  - Tail: Always points to the last element of the list
  - Current: Movable pointer used to Access and Modify Data in the List
  - Previous: Always stays on node behind Current
- Certain Linked Lists may omit some of these Nodes



# Problem

- How can we make this same structure without having to rewrite the code for every type?
- Generics
  - "Variables for Types"
  - Spoken: "This is a class of <<types>"
- In Java the Generic type must be an Object-Type
  - Everything in Java is assumed to inherit from type "Object"

## Syntax

```
public class <<class identifier>> < <<Generic Type>> >
{

}
```

## Example

```
public class GenLL <T>
{

}
```

# Applying Generics

- Change the type for the data to “T”
  - All functionality previously described works in the exact same way
  - Only difference being the type
- “T” is always an Object-Type in Java
  - The “==“ and “!=“ should only be used to refer to memory addresses
  - All Objects are assumed to have a “.equals(Object)” method in Java
  - All Objects are assumed to have a “.toString()” method

## Example

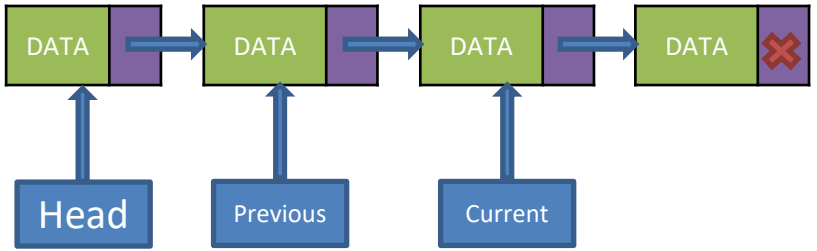
```
public class GenLL <T>
{
	private class ListNode
	{
		T data;
		ListNode link;
		public ListNode(T aData, ListNode aLink)
		{
			data = aData;
			link = aLink;
		}
	}
}
```

File Edit View Help

# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

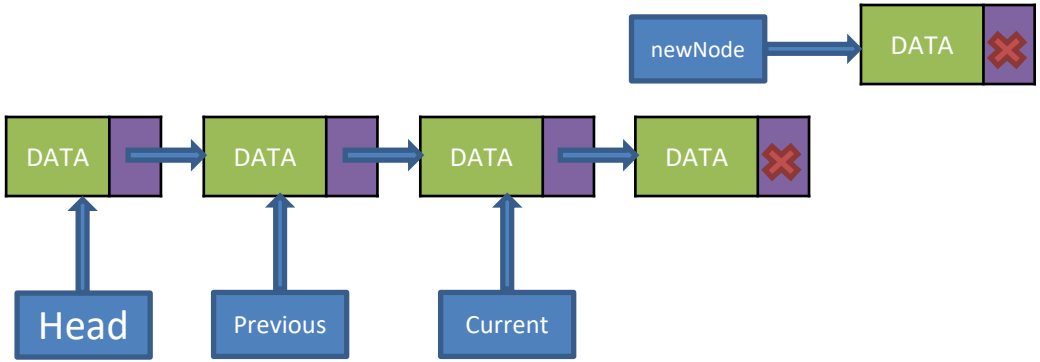
## Concept



# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

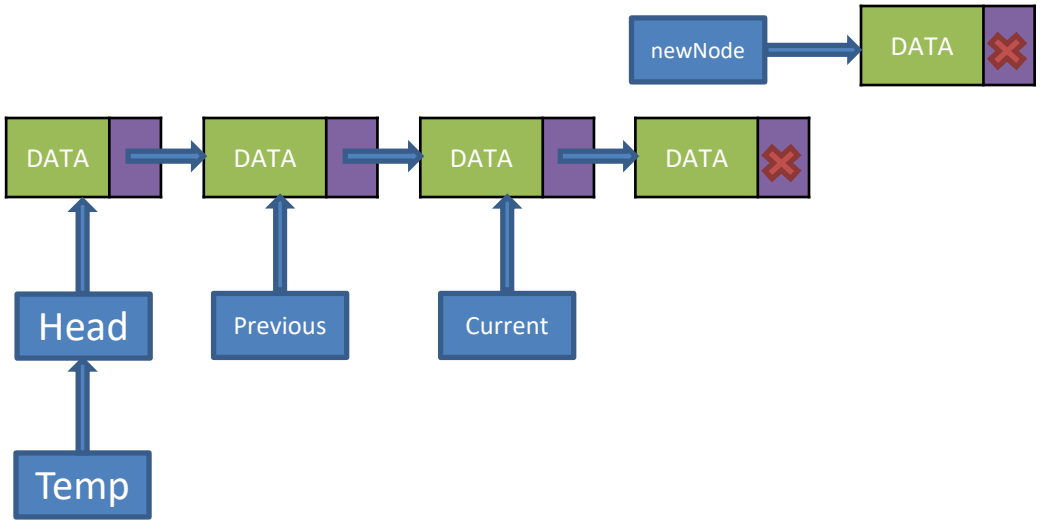
## Concept



# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

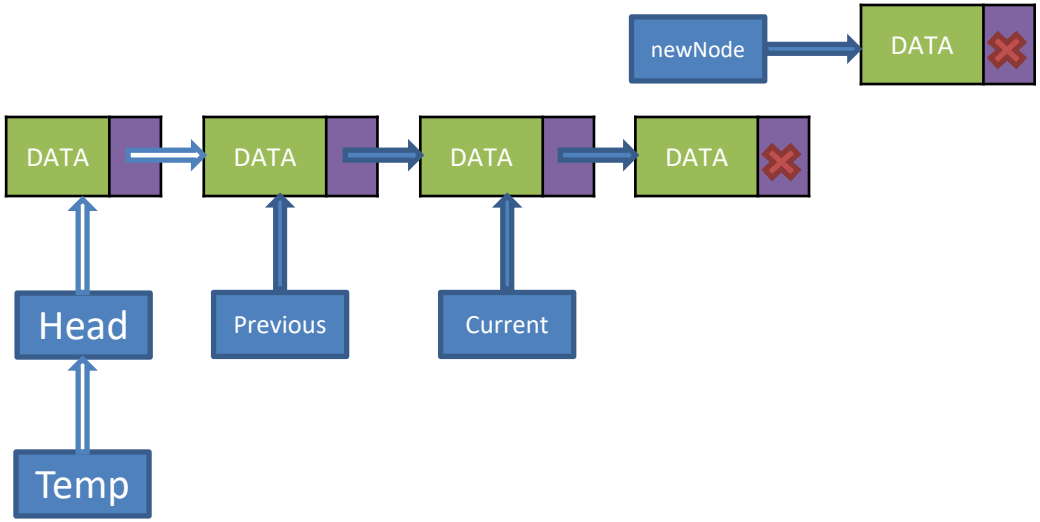
## Concept



# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

## Concept

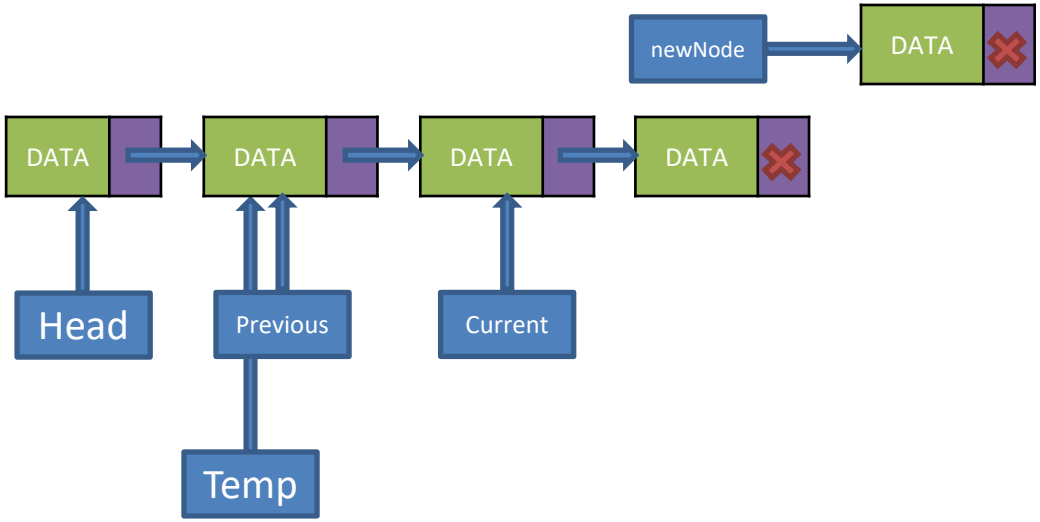


File Edit View Help

# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

## Concept

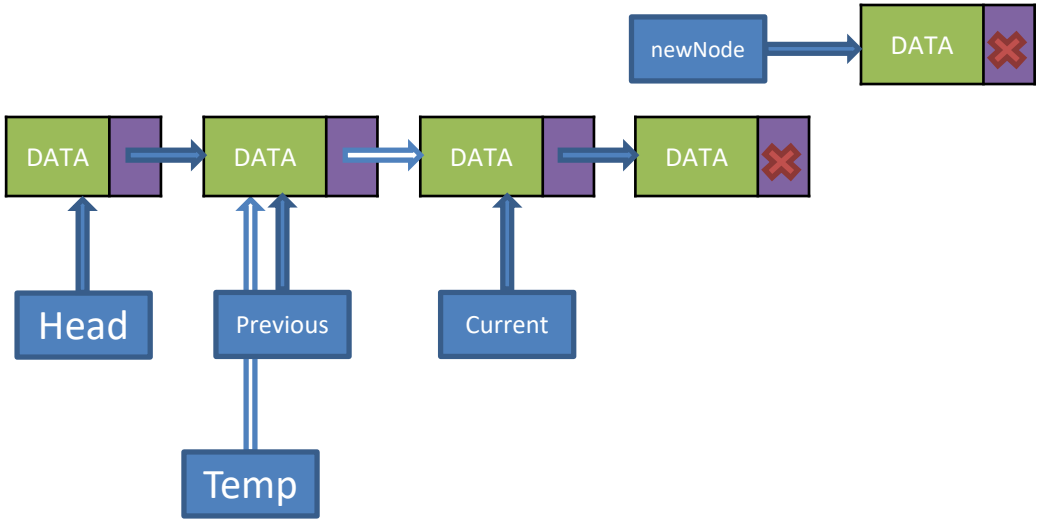


File Edit View Help

# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

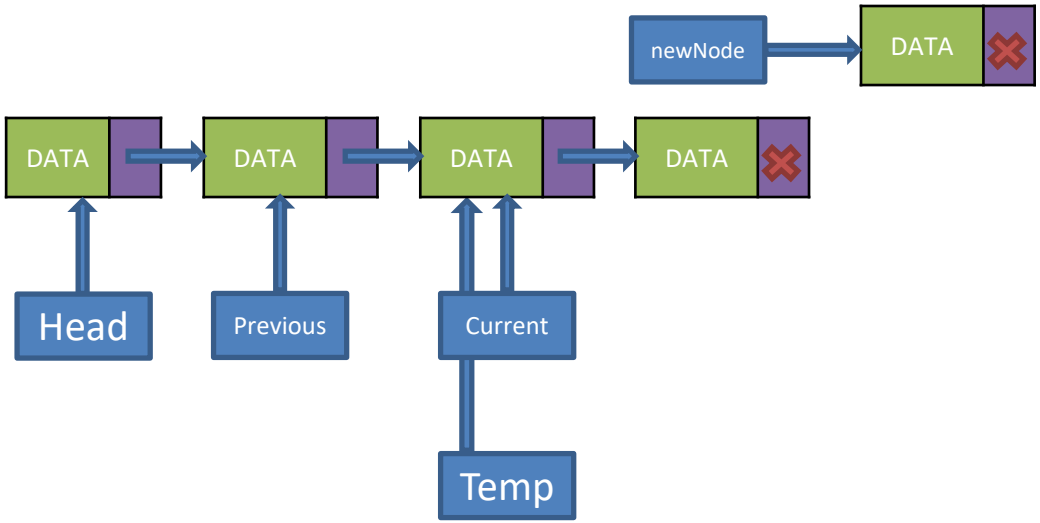
## Concept



# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

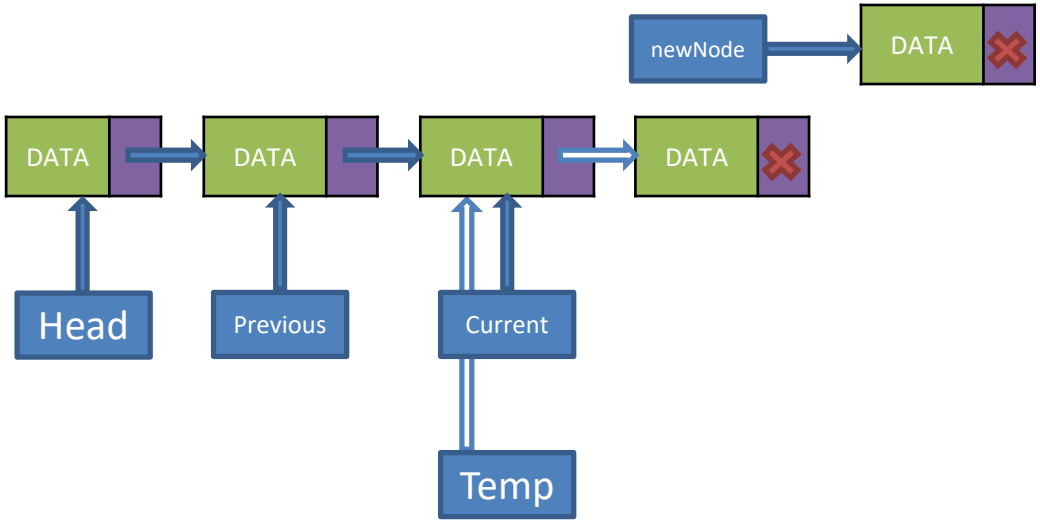
## Concept



# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

## Concept

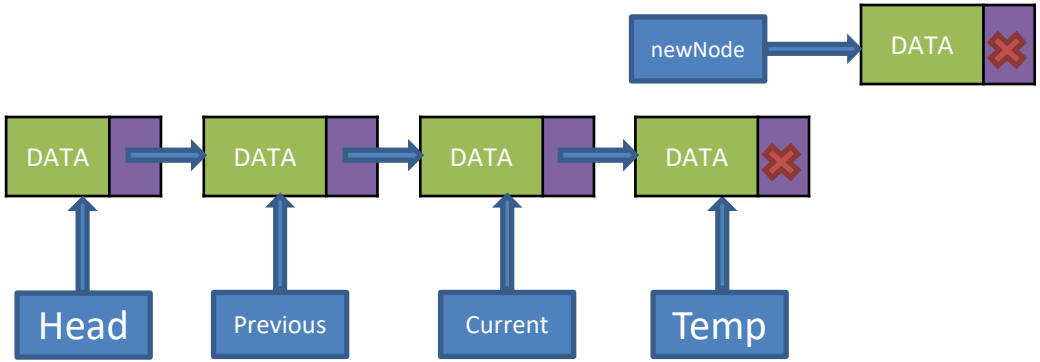


File Edit View Help

# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

## Concept

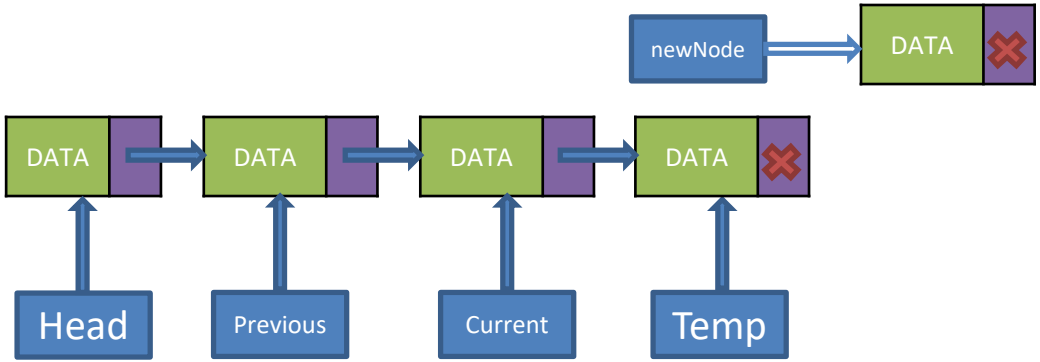


File Edit View Help

# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

## Concept

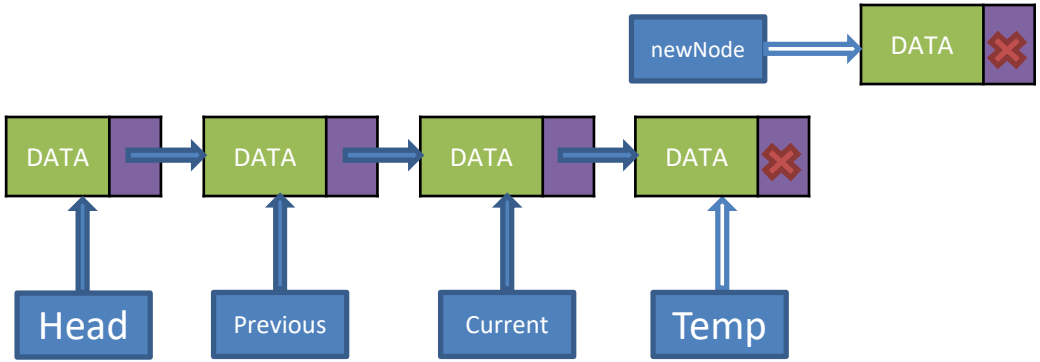


File Edit View Help

# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

## Concept

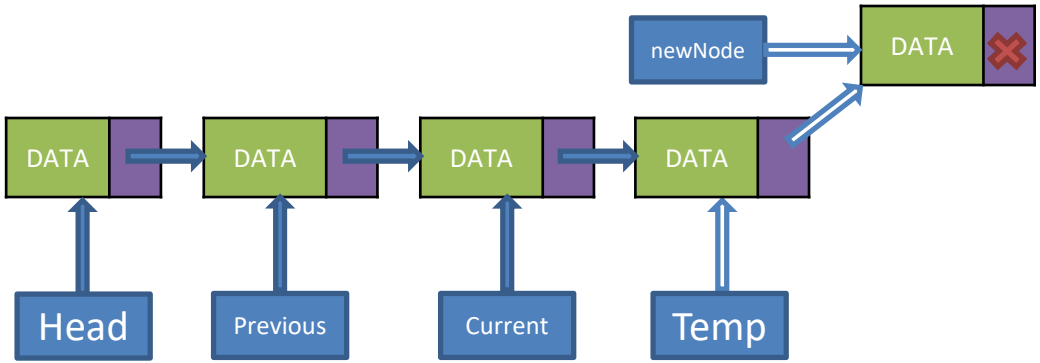


File Edit View Help

# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

## Concept

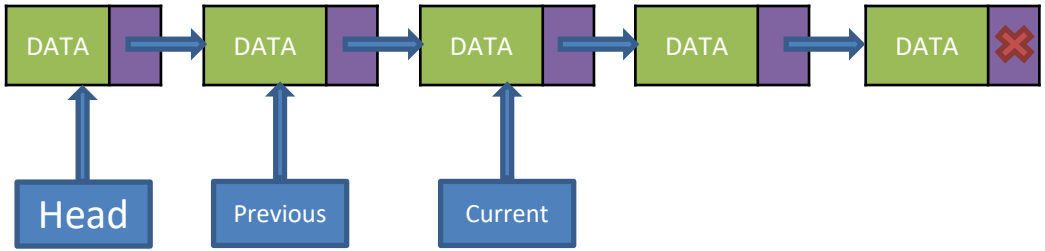


File Edit View Help

# Adding

- Create a new Node with the given Data
- Start from the Head and find the Node with the first Null Link
- Point that Node to the newly Created Node

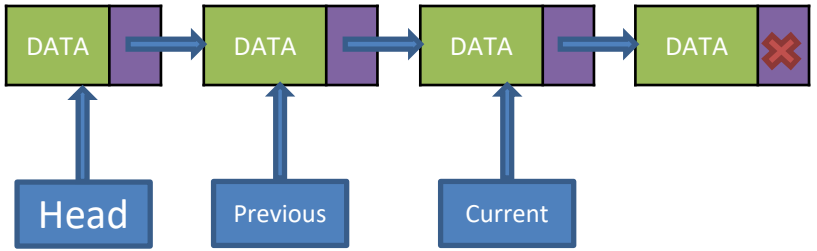
## Concept



# Adding After Current

- Create a new Node with the given Data
- Set new Node's Link to Current's Link
- Point Current's Link to the new Node

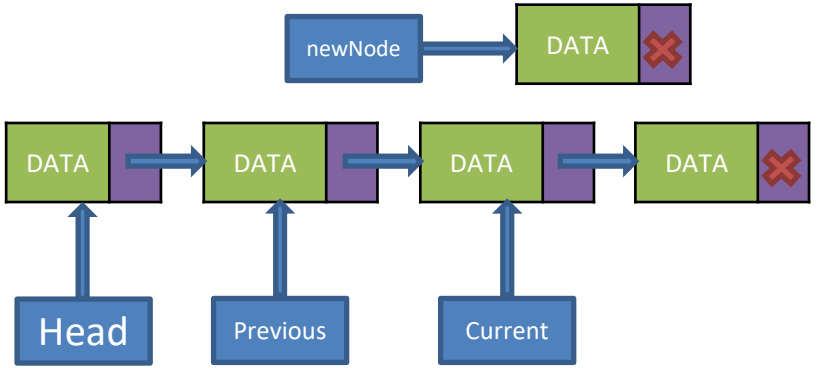
## Concept



# Adding After Current

- Create a new Node with the given Data
- Set new Node's Link to Current's Link
- Point Current's Link to the new Node

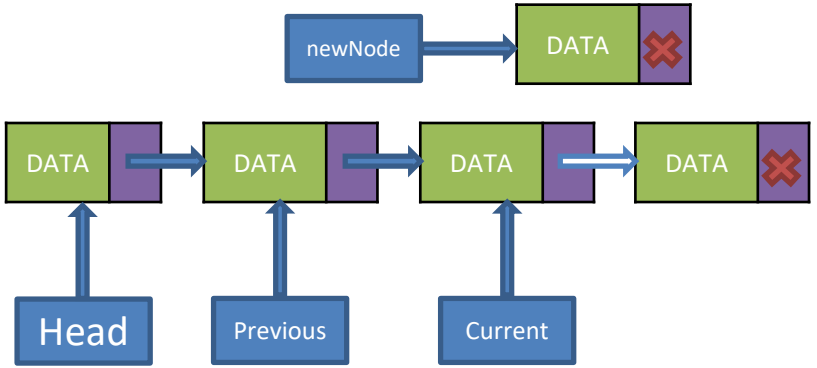
## Concept



# Adding After Current

- Create a new Node with the given Data
- Set new Node's Link to Current's Link
- Point Current's Link to the new Node

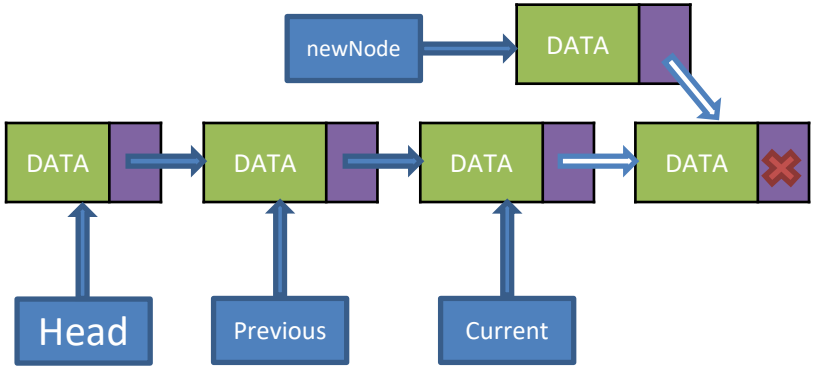
## Concept



# Adding After Current

- Create a new Node with the given Data
- Set new Node's Link to Current's Link
- Point Current's Link to the new Node

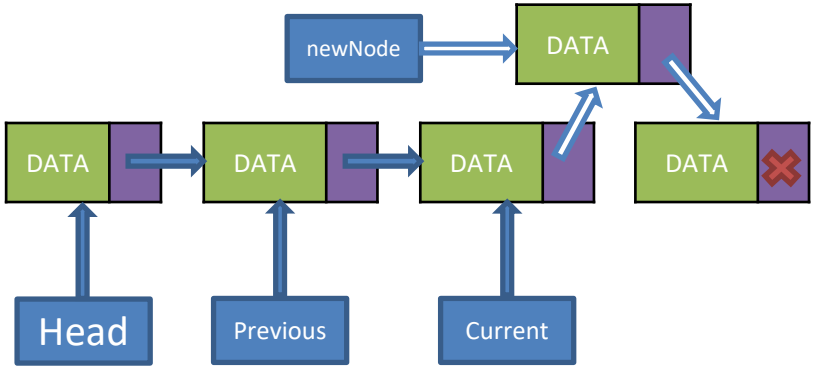
## Concept



# Adding After Current

- Create a new Node with the given Data
- Set new Node's Link to Current's Link
- Point Current's Link to the new Node

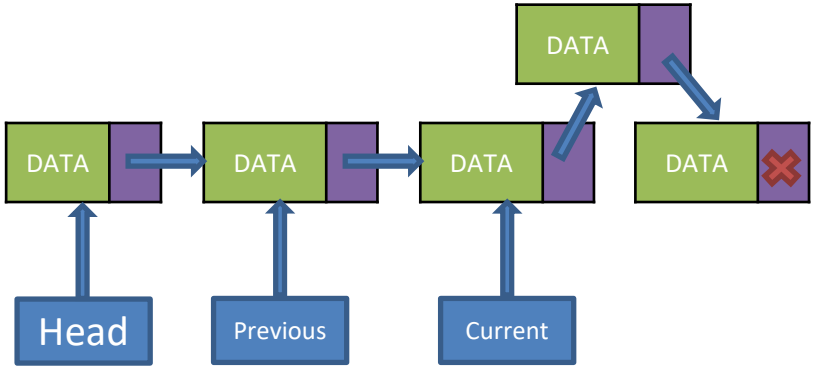
## Concept



# Adding After Current

- Create a new Node with the given Data
- Set new Node's Link to Current's Link
- Point Current's Link to the new Node

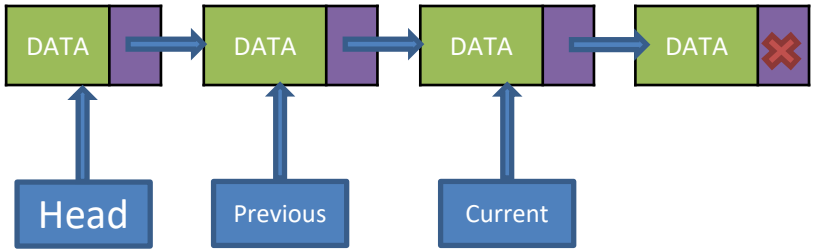
## Concept



# Removing Current

- If the Current is referencing the Head
  - Move Head and Current forward one node
- Set the Previous' Link to Current's Link
- Move Current Forward

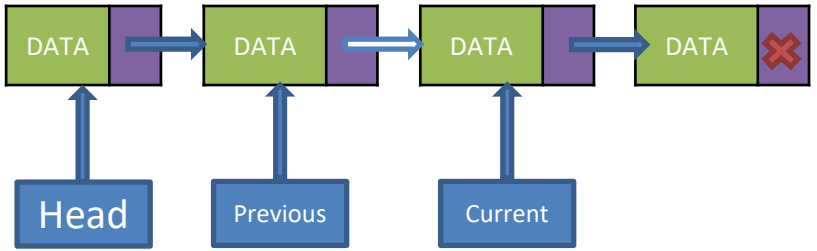
## Concept



# Removing Current

- If the Current is referencing the Head
  - Move Head and Current forward one node
- Set the Previous' Link to Current's Link
- Move Current Forward

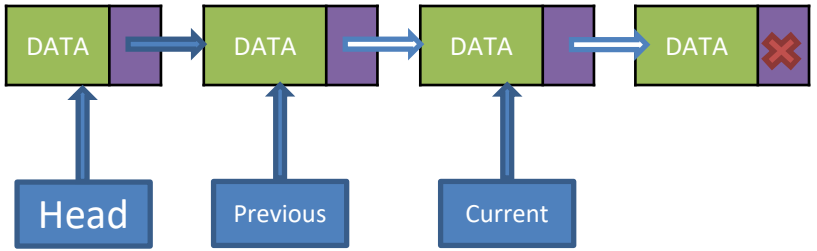
## Concept



# Removing Current

- If the Current is referencing the Head
  - Move Head and Current forward one node
- Set the Previous' Link to Current's Link
- Move Current Forward

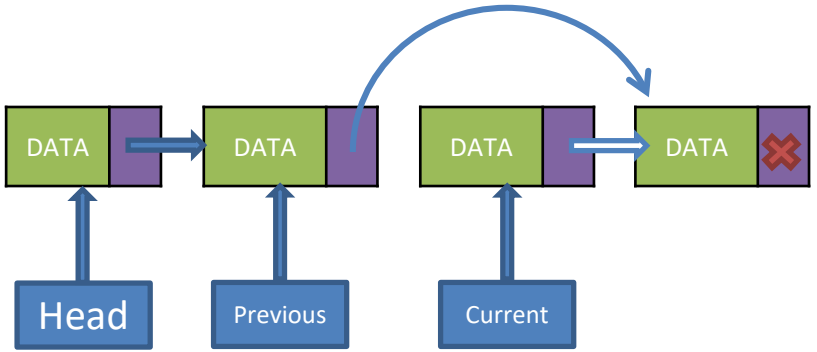
## Concept



# Removing Current

- If the Current is referencing the Head
  - Move Head and Current forward one node
- Set the Previous' Link to Current's Link
- Move Current Forward

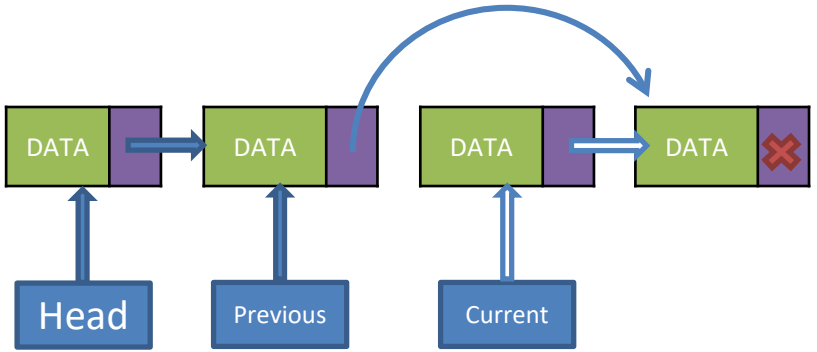
## Concept



# Removing Current

- If the Current is referencing the Head
  - Move Head and Current forward one node
- Set the Previous' Link to Current's Link
- Move Current Forward

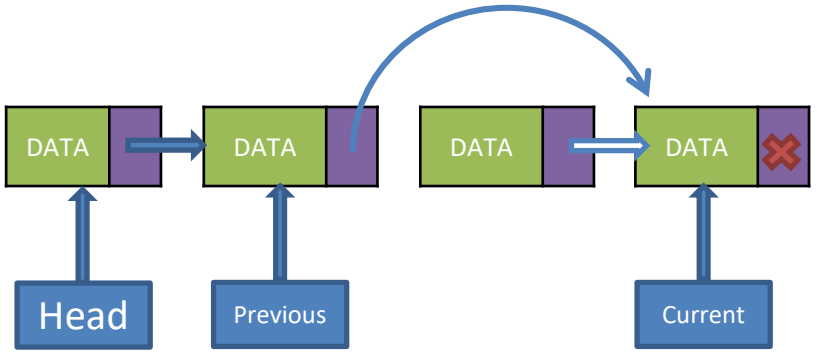
## Concept



# Removing Current

- If the Current is referencing the Head
  - Move Head and Current forward one node
- Set the Previous' Link to Current's Link
- Move Current Forward

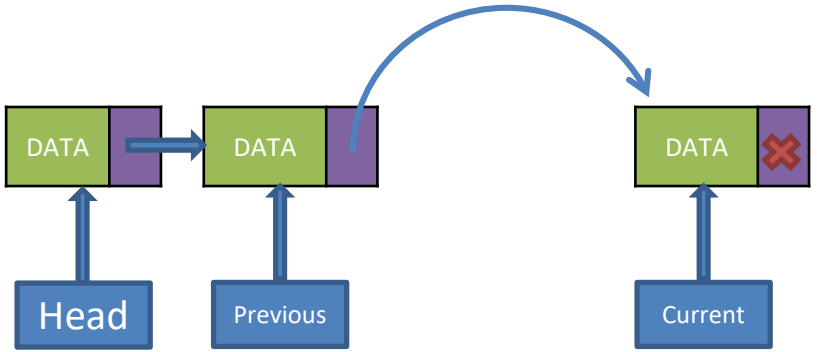
## Concept



# Removing Current

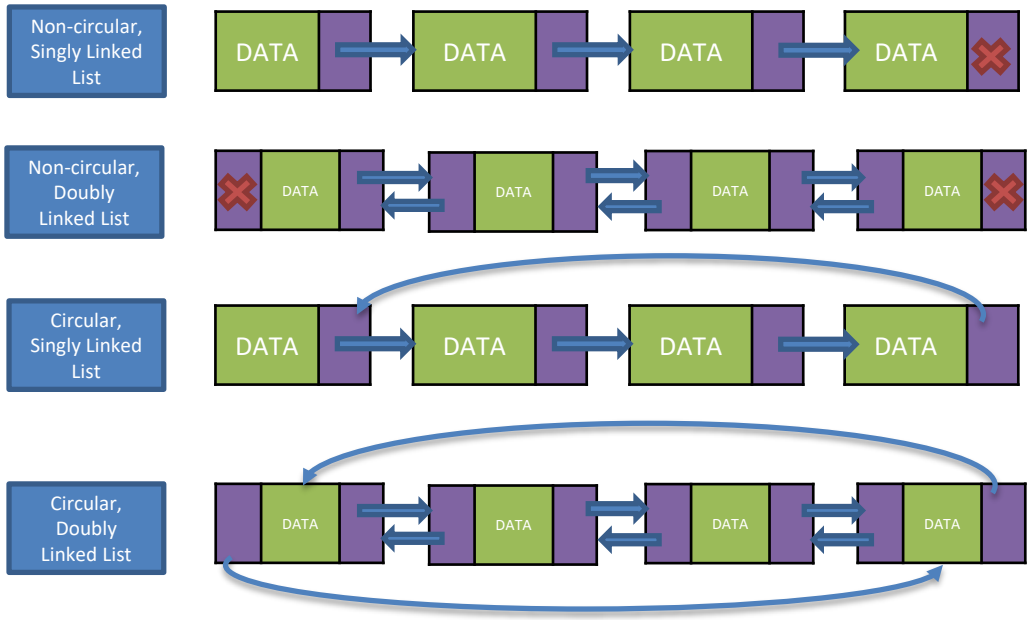
- If the Current is referencing the Head
  - Move Head and Current forward one node
- Set the Previous' Link to Current's Link
- Move Current Forward

## Concept



# Other Linked Lists

- Singly Linked Lists
- Doubly Linked Lists
- Circular Linked Lists



File Edit View Help

# Linking Structures