

Search...

[DSA](#) [Interview Problems on Linked List](#) [Practice Linked List](#) [MCQs on Linked List](#) [Linked List Tutorial](#) [Type](#)

# Deletion at beginning (Removal of first node) in a Linked List

Last Updated : 30 Aug, 2024

Given a **linked list**, The task is to **remove** the first node from the given **linked list**.

**Examples:**

***Input :** head : 3 -> 12 -> 15 -> 18 -> NULL*

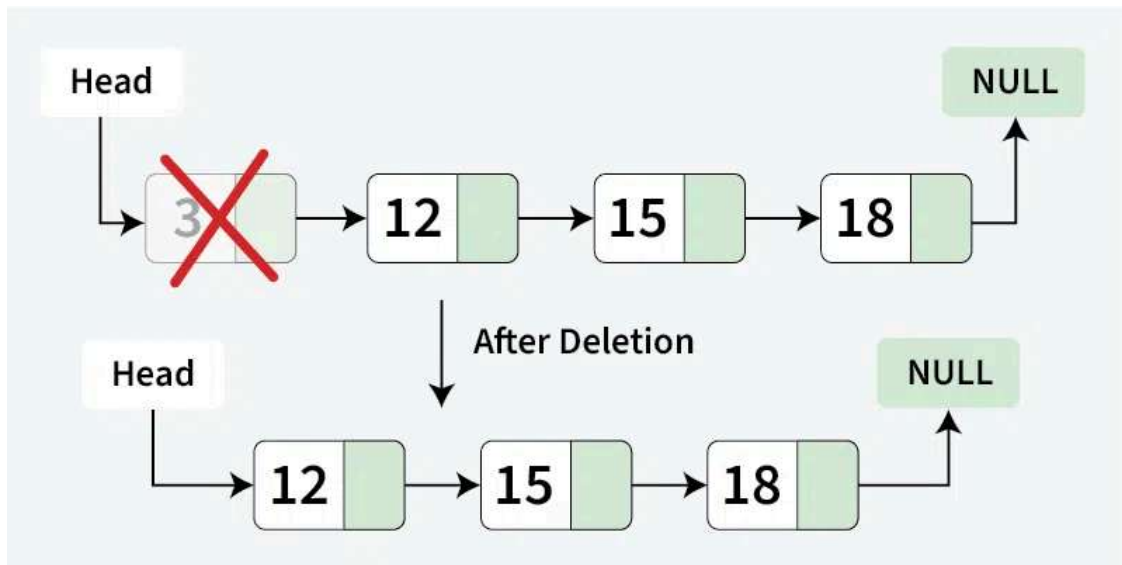
***Output :** 12 -> 15 -> 18 -> NULL*

***Input :** head : 2 -> 4 -> 6 -> 8 -> 33 -> 67 -> NULL*

***Output :** 4 -> 6 -> 8 -> 33 -> 67 -> NULL*

**By Shifting head node to next node of head – O(1) Time and O(1) Space**

*To remove the first node of a **linked list**, store the current **head** in a temporary variable (**temp**), move the **head** pointer to the next node, **delete** the temporary **head** node and finally , return the new **head** of the linked list.*



Deletion at beginning in a Linked List

Below is the implementation of the above approach:

C++

C

Java

Python

C#

JavaScript

```
// C# program to delete the head node
// from a linked list

using System;

class Node {
    public int data;
    public Node next;

    public Node(int data) {
        this.data = data;
        this.next = null;
    }
}

class Gfg {

    // Delete the head node and return the new head
    static Node DeleteHead(Node head) {

        // Check if the list is empty
        if (head == null)
            return null;

        // Move the head pointer to the next node
        head = head.next;

        return head;
    }

    static void PrintList(Node curr) {
        while (curr != null) {
            Console.Write(curr.data + " ");
            curr = curr.next;
        }
    }
}
```

```
    }  
}  
  
static void Main() {  
  
    // Create a hard-coded Linked List:  
    // 3 -> 12 -> 15 -> 18  
    Node head = new Node(3);  
    head.next = new Node(12);  
    head.next.next = new Node(15);  
    head.next.next.next = new Node(18);  
    head = DeleteHead(head);  
    PrintList(head);  
}  
}
```

## Output

12 15 18

**Time Complexity:**  $O(1)$ , because the operation to delete the head node is performed in constant time.

**Space Complexity:**  $O(1)$

Delete First Node of Singly Linked List

[Visit Course](#)

[Comment](#)

[More info](#)

[Advertise with us](#)

**Next Article**

[Min and Max in a List in Java](#)

## Similar Reads

### Deletion at beginning (Removal of first node) in a Doubly Linked List

Given a doubly linked list, the task is to delete the node from the beginning of the linked list. Examples: Input : $\hat{A}$  1  $\leftrightarrow$  2  $\leftrightarrow$  3  $\rightarrow$  NULL Output : $\hat{A}$  2  $\leftrightarrow$  3  $\leftrightarrow$  NULL Input : $\hat{A}$  2  $\leftrightarrow$  4  $\leftrightarrow$  6  $\leftrightarrow$  8  $\leftrightarrow$  33  $\leftrightarrow$  67  $\leftrightarrow$  ...

7 min read

---

### Deletion at end (Removal of last node) in a Linked List

Given a linked list, the task is to delete the last node of the given linked list. Examples: $\hat{A}$   $\hat{A}$  Input: 1  $\rightarrow$  2  $\rightarrow$  3  $\rightarrow$  4  $\rightarrow$  5  $\rightarrow$  NULL Output: 1  $\rightarrow$  2  $\rightarrow$  3  $\rightarrow$  4  $\rightarrow$  NULL Explanation: The last node of the linked list is 5, so 5 is...

8 min read

---

### Delete Kth nodes from the beginning and end of a Linked List

Given a singly Linked List and an integer K denoting the position of a Linked List, the task is to delete the Kth node from the beginning and end of the Linked List. Examples: Input: 1 ? 2 ? 3 ? 4 ? 5 ? 6, K = 3 Output: 1 ? 2 ...

13 min read

---

### Insert a Node at Front/Beginning of Doubly Linked List

Given a Doubly Linked List, the task is to insert a new node at the beginning/start/front of the linked list. Examples: Input: Linked List = 2  $\leftrightarrow$  3  $\leftrightarrow$  4  $\rightarrow$  NULL , New Node = 1 Output: 1  $\leftrightarrow$  2  $\leftrightarrow$  3  $\leftrightarrow$  4  $\rightarrow$  NULL...

9 min read

---

### Deletion at end (Removal of last node) in a Doubly Linked List

Given a doubly linked list, the task is to delete the last node of the given linked list. Examples: Input: $\hat{A}$  1  $\leftrightarrow$  2  $\leftrightarrow$  3  $\leftrightarrow$  NULL Output: $\hat{A}$  1  $\leftrightarrow$  2  $\leftrightarrow$  NULL Explanation: $\hat{A}$  The last node of the linked list is 3, so 3 is deleted...

7 min read

---

### Insert a Node at Front/Beginning of a Linked List

Given a linked list, the task is to insert a new node at the beginning/start/front of the linked list. Example: Input: LinkedList = 2 $\rightarrow$ 3 $\rightarrow$ 4 $\rightarrow$ 5, NewNode = 1 Output: 1 2 3 4 5 Input: LinkedList = 2 $\rightarrow$ 10, NewNode = ...

8 min read

---

### Swap first odd and even valued nodes from the beginning and end of a Linked List

Given a singly Linked List, the task is to swap the first odd valued node from the beginning and the first even valued node from the end of the Linked List. If the list contains node values of a single parity, then no...

13 min read

---

### Insert Node at the End of a Linked List

Given a linked list, the task is to insert a new node at the end of the linked list. Examples: Input: LinkedList = 2 -> 3 -> 4 -> 5, NewNode = 1 Output: LinkedList = 2 -> 3 -> 4 -> 5 -> 1 Input: LinkedList = NULL, NewNode = ...

9 min read

## Move first element to end of a given Linked List

Write a C function that moves first element to end in a given Singly Linked List. For example, if the given Linked List is 1->2->3->4->5, then the function should change the list to 2->3->4->5->1. Algorithm: Traver...

14 min read

## Insert a node in Linked List before a given node

Given a linked list, the task is to insert a new node with a specified value into a linked list before a node with a given key. Examples Input: head: 1 -> 2 -> 3 -> 4 -> 5 , newData = 6, key = 2 Output: 1 -> 6 -> 2 -> 3 -> 4 -> ...

15+ min read



Corporate & Communications Address:

A-143, 7th Floor, Sovereign Corporate  
Tower, Sector- 136, Noida, Uttar Pradesh  
(201305)

Registered Address:

K 061, Tower K, Gulshan Vivante  
Apartment, Sector 137, Noida, Gautam  
Buddh Nagar, Uttar Pradesh, 201305



Advertise with us

### Company

About Us  
Legal  
Privacy Policy  
In Media  
Contact Us  
Advertise with us  
GFG Corporate Solution

### Languages

Python  
Java  
C++  
PHP  
GoLang  
SQL  
R Language

Placement Training Program

Android Tutorial

Tutorials Archive

## DSA

Data Structures

Algorithms

DSA for Beginners

Basic DSA Problems

DSA Roadmap

Top 100 DSA Interview Problems

DSA Roadmap by Sandeep Jain

All Cheat Sheets

## Data Science & ML

Data Science With Python

Data Science For Beginner

Machine Learning

ML Maths

Data Visualisation

Pandas

NumPy

NLP

Deep Learning

## Web Technologies

HTML

CSS

JavaScript

TypeScript

ReactJS

NextJS

Bootstrap

Web Design

## Python Tutorial

Python Programming Examples

Python Projects

Python Tkinter

Python Web Scraping

OpenCV Tutorial

Python Interview Question

Django

## Computer Science

Operating Systems

Computer Network

Database Management System

Software Engineering

Digital Logic Design

Engineering Maths

Software Development

Software Testing

## DevOps

Git

Linux

AWS

Docker

Kubernetes

Azure

GCP

DevOps Roadmap

## System Design

High Level Design

Low Level Design

UML Diagrams

Interview Guide

Design Patterns

OOAD

System Design Bootcamp

Interview Questions

## Interview Preparation

Competitive Programming

Top DS or Algo for CP

Company-Wise Recruitment Process

Company-Wise Preparation

Aptitude Preparation

Puzzles

## School Subjects

Mathematics

Physics

Chemistry

Biology

## GeeksforGeeks Videos

DSA

Python

Java

C++

Social Science  
English Grammar  
Commerce  
World GK

Web Development  
Data Science  
CS Subjects

---

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved