Bahria University,

Karachi Campus

A picture containing text, room

Description automatically generated

LAB EXPERIMENT NO.

\_\_\_\_**03**\_\_\_\_\_

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 1 | Write a program to create a linked list and pertform:  \*traversing  \* Insertion  \*deletion |
|  |  |
|  |  |
|  |  |
|  |  |

Submitted On:

\_\_\_\_18/10/2022\_\_\_\_

(Date: DD/MM/YY)

**Task No. 1 : Write a program to create a linked list and pertform**

**\*Traversing**

**\* Insertion**

**\*Deletion**

**Solution:**

* **Traversing:**

**Main Class:**

LinkedList list = new LinkedList();

list.head = new Node(5);

Node second = new Node(10);

Node third = new Node(20);

list.head.next = second;

second.next = third;

list.printList();

**Node Class:**

public int data;

public Node next;

public Node(int d){

data = d;

next = null;}

**LinkedList Class:**

public Node head;

public void printList(){

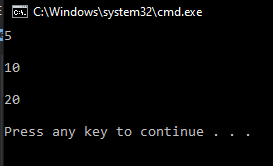
Node n=head;

while(n!=null){

Console.WriteLine(n.data + "\n");

n = n.next;}}

**Output:**



* **Insertion**

**Solution:**

**Main Class:**

LinkedList list = new LinkedList();

list.head = new Node(5);

Node second = new Node(10);

Node third = new Node(20);

list.head.next = second;

second.next = third;

list.printList();

list.append(6);

list.push(7);

list.push(1);

list.insertAfter(list.head.next, 8);

Console.Write("\n\nCreated Linked list is: \n\n");

list.printList();

**LinkedList Class:**

public Node head;

public void printList(){

Node n=head;

while(n!=null){

Console.Write(n.data + " ");

n = n.next;}}

public void push(int new\_data){

Node new\_node = new Node(new\_data);

new\_node.next = head;

head = new\_node;}

public void insertAfter(Node prev\_node, int new\_data){

if (prev\_node == null){

Console.WriteLine("The given previous node cannot be null");

return;}

Node new\_node = new Node(new\_data);

new\_node.next = prev\_node.next;

prev\_node.next = new\_node;}

public void append(int new\_data){

Node new\_node = new Node(new\_data);

if (head == null){

head = new Node(new\_data);

return;}

new\_node.next = null;

Node last = head;

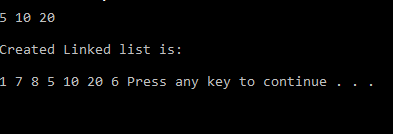
while (last.next != null)

last = last.next;

last.next = new\_node;

return;}

**Output:**



* **Deletion**

**Main Class:**

LinkedList list = new LinkedList();

list.head = new Node(5);

Node second = new Node(10);

Node third = new Node(20);

list.head.next = second;

second.next = third;

list.printList();

list.append(6);

list.push(7);

list.push(1);

list.insertAfter(list.head.next, 8);

Console.Write("\n\nCreated Linked list is: \n\n");

list.printList();

list.deleteNode(1);

**LinkedList Class:**

public void deleteNode(int key){

Node temp = head, prev = null;

if (temp != null &&

temp.data == key){

head = temp.next;

return;}

while (temp != null &&

temp.data != key){

prev = temp;

temp = temp.next;}

if (temp == null)

return;

prev.next = temp.next;}

**Output:**

