**SINGLY LINKED LIST**

**import** java.util.\*;

// class node is defining the structure of the node

**class Node {**

**int data;**

**Node next;**

**}**

**public** **class** SinglyLinkedList {

Node head; // head reference of Node type

// To check list is empty or not

**public** **boolean** isEmpty() {

**return** (head == **null**);

}

// used to insert a node at the start of linked list

**public** **void** insertFirst(**int** data) {

Node newNode = **new** Node();

newNode.data = data;

newNode.next = head;

head = newNode;

}

// used to insert a node at the start of linked list

**public** **void** insertLast(**int** data) {

**if**(isEmpty())

insertFirst(data);

**else** {

Node current = head;

Node newNode = **new** Node();

**while**(current.next!=**null**)

current=current.next;

newNode.data=data;

newNode.next=**null**;

current.next=newNode;

}

}

// insertion after a given node by data

**public** **void** insertAfterData(**int** dataAfter, **int** item) {

**if**(isEmpty())

System.***out***.println("List is empty insertion not possible");

**else** {

Node temp = head;

Node newNode = **new** Node();

**while**(temp.next!=**null** && temp.data!=dataAfter)

temp=temp.next;

**if**(temp.next!=**null**) {

newNode.data=item;

newNode.next=temp.next;

temp.next=newNode;

}

**else** **if**(temp.data!=dataAfter)

System.***out***.println("Node is not present in the list");

**else** {

newNode.data=item;

newNode.next=temp.next;

temp.next=newNode;

}

}

}

// Deletion at the front

**public** **void** deleteFirst() {

**if**(isEmpty()){

System.***out***.println("List is empty");

**return**;

}

Node temp = head;

head = head.next;

System.***out***.println("Data deleted is: "+temp.data);

}

// Deletion at the end

**public** **void** deleteLast() {

**if**(isEmpty()){

System.***out***.println("List is empty");

**return**;

}

Node temp = head;

**if**(temp.next == **null**){

head = **null**;

**return**;

}

Node prev=null;

**while**(temp.next != **null**){

prev = temp;

temp = temp.next;

}

prev.next = **null**;

System.***out***.println("Data deleted is: "+temp.data);

}

// Deletion after a given node position

**public** **void** deleteAfterData(**int** data) {

**if**(isEmpty()){

System.***out***.println("List is empty");

**return**;

}

Node temp = head;

**while**(temp.next != **null** && temp.data != data)

temp = temp.next;

**if**(temp.next != **null**){

System.***out***.println("Data deleted is: "+temp.next.data);

temp.next = temp.next.next;

}

**else** **if**(temp.data != data)

System.***out***.println("Node is not present in the list");

**else**

System.***out***.println("Deletion not possible this is the last node");

}

// For printing Linked List

**public** **void** displayList() {

**if**(isEmpty())

System.***out***.println("Nothing to display");

**else** {

System.***out***.println("List is:");

**for**(Node current=head;current!=**null**;current=current.next)

System.***out***.print(current.data+"-->");

System.***out***.println();

}

}

**public** **static** **void** main(String []args) {

SinglyLinkedList list = **new** SinglyLinkedList();

Scanner sc = **new** Scanner(System.***in***);

**int** op, item;

System.***out***.println("Implementation of Singly Linked List");

**while**(**true**) {

System.***out***.print("1.Insertion at the begining\n2.Insertion at the end\n3.Insertion after given node(data)\n4.Deletion at the begining\n5.Deletion at the end\n6.Deletion after a given node(data)\n7.Display list\n8.Exit\n");

op = sc.nextInt();

**switch**(op) {

**case** 1:

System.***out***.print("Enter value: ");

item = sc.nextInt();

list.insertFirst(item);

list.displayList();

**break**;

**case** 2:

System.***out***.print("Enter value: ");

item = sc.nextInt();

list.insertLast(item);

list.displayList();

**break**;

**case** 3:

System.***out***.print("Enter data after which you want to insert node: ");

**int** dataAfter = sc.nextInt();

System.***out***.print("Enter the value to insert: ");

item = sc.nextInt();

list.insertAfterData(dataAfter, item);

list.displayList();

**break**;

**case** 4:

list.deleteFirst();

list.displayList();

**break**;

**case** 5:

list.deleteLast();

list.displayList();

**break**;

**case** 6:

System.***out***.print("Enter data after which you want to do deletion: ");

item = sc.nextInt();

list.deleteAfterData(item);

list.displayList();

**break**;

**case** 7:

list.displayList();

**break**;

**case** 8:

System.*exit*(1);

}

}

}

}