|  |  |
| --- | --- |
| Student Name | Uzair Hussain |
| Roll Number | 21SW085 |
| Section # | 3rd or III |
| Lab # | 8th – Doubly LinkedList |

**Task#01**

**Code:**

import java.util.Scanner;

class DoublyLinkedList{

    Scanner sc = new Scanner(System.in);

    static class Node{

        Node next, prev;

        String image;

        public Node(String image){

            this.image = image;

        }

    }     // end of class Node

    Node head, tail;

    int length;

    public DoublyLinkedList(){

        this.head = null;

        this.tail = null;

        this.length = 0;

    }

    public boolean isEmpty(){

        return (length==0);

    }

    public void length(){

        System.out.println(length);

    }

    public void slideShow(){

        Node p = head;

        while (p!=null){

            if(p.next==null)

                System.out.println(p.image+" -> null");

            else

                System.out.print(p.image+" -> ");

            p = p.next;

        }

        System.out.println();

    }

    public void insert(){

        System.out.print("Add image: ");

        String image = sc.nextLine();

        Node p = new Node(image);

        if (isEmpty()){

            head = p;

        } else {

            tail.next = p;

            p.prev = tail;

        }

        tail = p;

        length++;

    }     // end of prev

    public void search(){

        System.out.print("Enter name of image to search: ");

        String image = sc.nextLine();

        boolean flag = false;

        for (Node p=head; p!=null; p=p.next){

            if (p.image.equalsIgnoreCase(image)) {

                flag = true;

                break;

            }

        }

        if (flag)

            System.out.println("Image found");

        else

            System.out.println("Image not found");

    }     // end of search

    public Node delete(){

        System.out.print("Enter name of image to delete: ");

        String image = sc.nextLine();

        // If image found at head (OR node 1)

        if (head.image.equals(image)){

            head = head.next;

            length--;

            return head;

        } else if (tail.image.equals(image)) {

            Node p = head;

            for (p = head; p.next.next != null; p = p.next) {

            }

            p.next = null;

        }

        for (Node p=head; p.next!=null; p=p.next) {

            if (p.next.image.equals(image)) {

                p.next = p.next.next;

                length--;

            }

        }

        return head;

    }    // end of delete()

    public Node rename(){

        System.out.println("Enter image to rename: ");

        String image = sc.nextLine();

        System.out.println("Enter new image name: ");

        String newImage = sc.nextLine();

        for (Node p=head; p!=null; p=p.next) {

            if (p.image.equals(image)){

                p.image = newImage;

                break;

            }

        }

        return head;

    }

    public Node next(){

        System.out.println("Current image is "+head.image);

        head = head.next;

        return head;

    }

    public Node previous(){

        System.out.println("Current image is "+tail.image);

        tail = tail.prev;

        return tail;

    }

}     // end of class DoublyLinkedList

public class Task1\_Doubly\_LinkedList {

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

    public static void display(){

        System.out.println("1. Add an image");

        System.out.println("2. Delete");

        System.out.println("3. Search");

        System.out.println("4. Rename");

        System.out.println("5. Total number of images stored");

        System.out.println("6. Slideshow");

        System.out.println("7. Next image");

        System.out.println("8. Previous image");

    }

    public static void main(String[] args) {

        DoublyLinkedList doublyLinkedList = new DoublyLinkedList();

//        doublyLinkedList.insertAtEnd("freeCodeCamp logo.jpg");

//        doublyLinkedList.insertAtEnd("Dell.png");

//        doublyLinkedList.insertAtEnd("Iphone.jpg");

//        doublyLinkedList.insertAtEnd("Shinning Sky.jpg");

        int choice;

        do{

            display();

            System.out.print("Enter your choice: ");

            choice = sc.nextInt();

            switch (choice){

                case 1 -> doublyLinkedList.insert();

                case 2 -> doublyLinkedList.delete();

                case 3 -> doublyLinkedList.search();

                case 4 -> doublyLinkedList.rename();

                case 5 -> doublyLinkedList.length();

                case 6 -> doublyLinkedList.slideShow();

                case 7 -> doublyLinkedList.next();

                case 8 -> doublyLinkedList.previous();

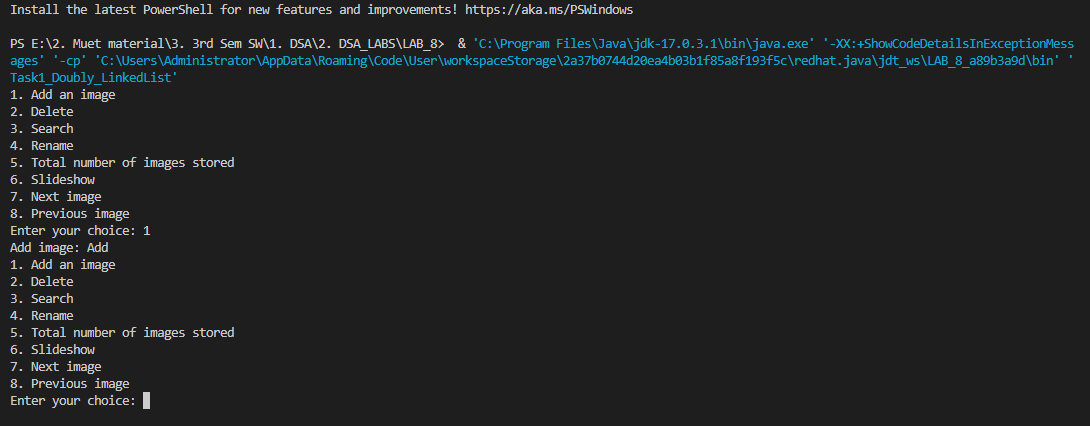
            }

        }while (choice!=0);

    }     // end of main()

}

**Output 1:**

****

**Github Repository for all Lab Tasks: (from lab 1 to continue)**

[**https://github.com/UzairHussain193/DSA\_LABS\_21SW**](https://github.com/UzairHussain193/DSA_LABS_21SW)

**The End!**