/*Date: 22-7-25

Scenario: College Exam Paper Collection

During semester exams, invigilators collect answer scripts from students. Each script is placed on top of a stack on the exam supervisor's desk. The last script collected must be evaluated first, following the Last-In-First-Out (LIFO) order.

Design and implement this answer script collection system using a linked list-based stack in Java.

Operations to Implement:

• Push: Add a newly collected answer script to the top of the stack.

• Pop: Remove and process the top answer script (the one to be evaluated first).

• Peek: View the most recently collected script (without removing it).

• Display: Print all collected scripts from top to bottom.

Data to Store per Script:

• scriptID (e.g., "EX123")

• studentName (e.g., "Priya Sharma")*/

```
package julyhometask;
import java.util.*;
class CollegeNode
{
	String sc_id;
	String stu_name;
	CollegeNode next;
	CollegeNode(String sc_id,String stu_name)
	{
		this.sc_id=sc_id;
		this.stu_name=stu_name;
		this.next=null;

	}
}
public class july22ht
{
```

```
	CollegeNode top;
	july22ht()
	{
		this.top=null;
	}
	public void push(String sc_id,String stu_name)
	{
		CollegeNode newNode=new CollegeNode(sc_id,stu_name);
		newNode.next=top;
		top=newNode;
		System.out.println("The recently submitted answer sheet with id "+sc_id +" and
name "+stu_name +" is placed at first");


	}
	public void pop()
	{
		if(top==null)
		{
			System.out.println("There are no answer sheets left");
			return;
		}
		String topID=top.sc_id;
		String topName=top.stu_name;
		top=top.next;
		System.out.println("The answer sheet of "+ topName+" with id "+topID+" is
eveluated firstly");
	}
	String[]peek()
	{
		if(top==null)
		{
			System.out.println("There are no answer sheets left behind");
```

```
                return new String[]{"-1"};


        }
        String[]data= {top.sc_id,top.stu_name};
        return data;
        }
    public void display()
    {
        System.out.println("The available answer sheets for correction are");
        CollegeNode temp=top;
        while(temp!=null)
        {
            System.out.println("ID:"+temp.sc_id+" Name:"+temp.stu_name);
            temp=temp.next;
        }
        System.out.println("There are no answer sheets left behind");
    }


    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        july22ht as=new july22ht();
        int choice;
        do
        {


        System.out.println("1.Place the recent submitted sheets\n2.Evaluate the current top
sheet\n3.See the current top sheet\n4.Display all sheets\n5.Exit");
        choice=sc.nextInt();
```

```
            sc.nextLine();
            switch(choice)
            {
            case 1:
                    System.out.println("Enter the answer sheet id");
                    String sc_id=sc.nextLine();
                    System.out.println("Enter the name of the student");
                    String stu_name=sc.nextLine();
                    as.push(sc_id,stu_name);
                    break;
            case 2:
                    as.pop();
                    break;
            case 3:
                    String[]data=as.peek();
                    System.out.println("ID:"+data[0]+" Name:"+data[1]);

                    break;
            case 4:
                    as.display();
                    break;
            case 5:
                    System.out.println("The system is exiting---");
                    break;
            default:
                    System.out.println("Please enter valid option");
                    break;
            }
            }while(choice!=5);
            sc.close();
}
```

```
}
```