



Scenario Based Questions

Topic: Java

1. Write a Java Program to iterate ArrayList using for-loop, while-loop, and advance for-loop to get the result as shown below:

While Loop:

20

30

40

Advanced For Loop:

20

30

40

For Loop:

20

30

40

Solution:

```
package assignmentSolutions;
import java.util.*;

public class ArrayListSolution
{
    public static void main(String[] args)
    {
        List<Integer> l1=new ArrayList();

        l1.add(20);
        l1.add(30);
        l1.add(40);

        System.out.println("While Loop:");
        int n=0;

        while(l1.size()>n)
        {
            System.out.println(l1.get(n));
            n++;
        }

        System.out.println("Advanced For Loop:");
        for(Integer j:l1)
        {
```

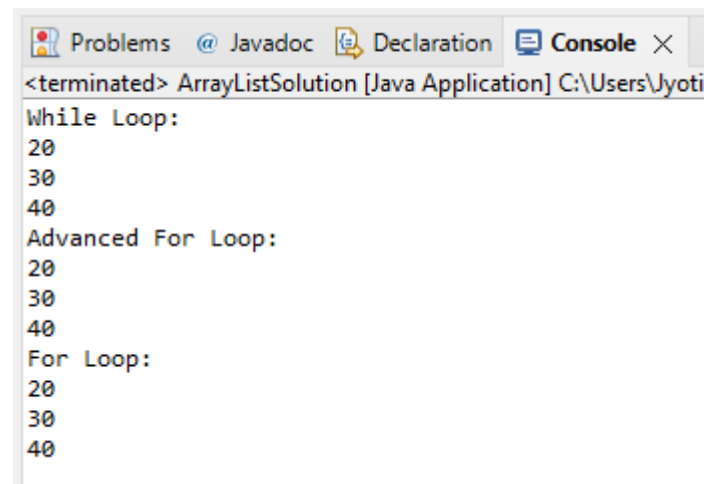
```

        System.out.println(j);
    }

    System.out.println("For Loop:");
    for(int i=0;i<l1.size();i++)
    {
        System.out.println(l1.get(i));
    }
}
}

```

OUTPUT



```

<terminated> ArrayListSolution [Java Application] C:\Users\Jyoti
While Loop:
20
30
40
Advanced For Loop:
20
30
40
For Loop:
20
30
40

```

3. create a doubly linked list and rotate it by n node to get the result as shown below:

```

Original List:
1 2 3 4 5
Updated List:
4 5 1 2 3

```

Solution

```

package assignmentSolutions;

public class DoublyLinkedList1
{
    static class Node
    {
        int element;
        Node pre;
        Node next;
    }
}

```

```

static Node head = null;
static void rotate( int n)
{
    if (n == 0)
        return;
    Node current = head;

    int count = 1;
    while (count < n && current != null)
    {
        current = current.next;
        count++;
    }
    if (current == null)
        return;
    Node nthNode = current;
    while (current.next != null)
        current = current.next;

    current.next = head;
    (head).pre = current;
    head = nthNode.next;
    (head).pre = null;

    nthNode.next = null;
}
static void push(int new_data)
{
    Node new_node = new Node();
    new_node.element = new_data;
    new_node.pre = null;
    new_node.next = (head);
    if ((head) != null)
        (head).pre = new_node;
    head = new_node;
}
static void displayList(Node node)
{
    while (node != null && node.next != null)
    {
        System.out.print(node.element + " ");
        node = node.next;
    }
    if (node != null)
        System.out.print(node.element);
}

public static void main(String[] args)
{
    push(5);
    push(4);
    push(3);
    push(2);
    push(1);

    int n = 3;

    System.out.println("Original List: ");
}

```

```

        displayList(head);
        rotate(n);
        System.out.println();
        System.out.println("Updated list ");
        displayList(head);
    }

}

```

OUTPUT

The screenshot shows a Java IDE window with the title bar "Problems @ Javadoc Declaration Console X". The console output is as follows:

```

<terminated> DoublyLinkedList1 [Java Application] C:\Users\Jyoti
Original List:
1 2 3 4 5
Updated list:
4 5 1 2 3

```

Topic: SQL

3. At St. Xavier's College, a faculty has the following data in My SQL in database named as Class having table student related to Semester Examination.

Enrollment_No	Student_name	section	Subject_id	Marks
1	Tim	A	1	70
2	Jim	A	2	75
3	Kim	B	3	65
4	Tom	B	4	77
5	John	C	5	60
6	Joe	C	1	82
7	James	B	2	76
8	Henry	C	5	68
9	Matt	B	3	71
10	Paul	A	4	79

10 rows in set (0.00 sec)

Solution

```

mysql> select section, Count(Marks) "as No. of Candidate greater than or equal to 75 marks"
from student1
-> where Marks>=75
-> group by Section;

```

OUTPUT

section		as No. of Candidate greater than or equal to 75 marks
A		2
B		2
C		1