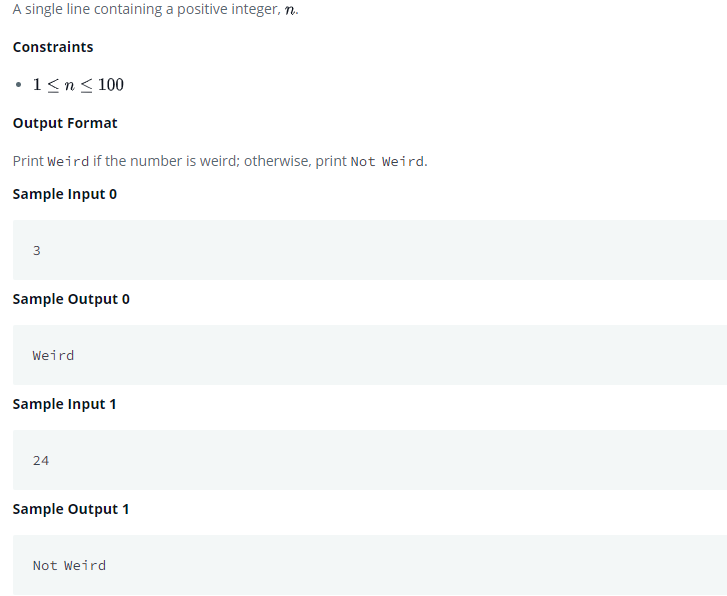
**NAMA : JAMALUDIN**

**TYPE : EASY**

1. **Question “Java If-Else”**

****

**Answer**

**import java.io.\*;**

**import java.math.\*;**

**import java.security.\*;**

**import java.text.\*;**

**import java.util.\*;**

**import java.util.concurrent.\*;**

**import java.util.regex.\*;**

**public class Solution {**

**private static final Scanner scanner = new Scanner(System.in);**

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

**int N = scanner.nextInt();**

**String ans ="";**

**if(N >= 1 && N<=100){**

**if(N%2==1){**

**System.out.print("Weird");**

**}**

**else{**

**if(N>=2 && N<=5){**

**System.out.print("Weird");**

**}**

**else if(N>20){**

**System.out.print("Not Weird");**

**}**

**}**

**}**

**else{**

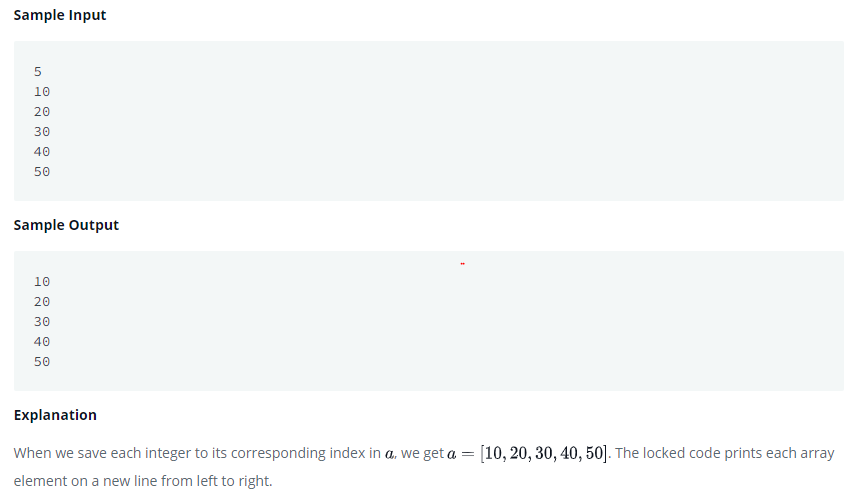
**System.out.print("Invalid");**

**}**

**}**

**}**

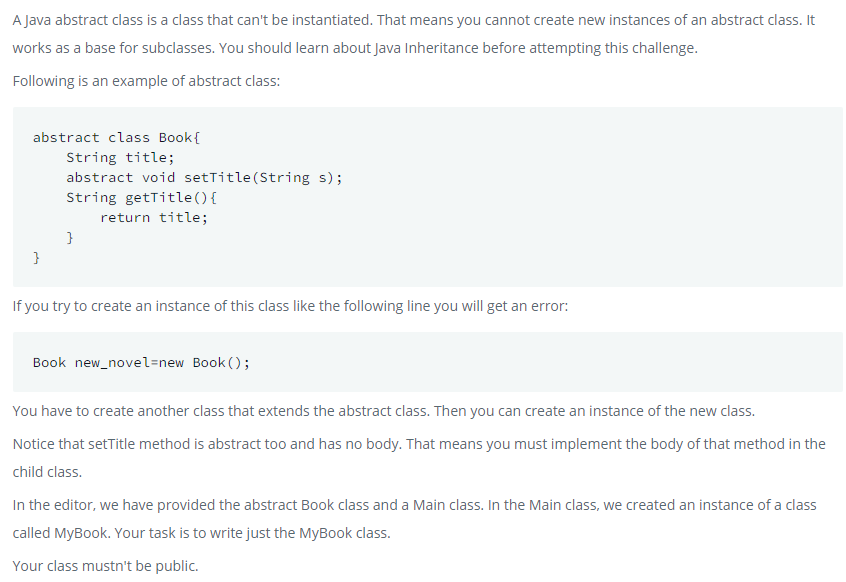
1. **Question “Java 1D Array”**



**Answer**

1. **import** java.util.\*;
2. **public** **class** Solution {
3. **public** **static** **void** main(String[] args) {
5. Scanner scan = **new** Scanner(System.in);
6. **int** n = scan.nextInt();
7. **int**[] a=**new** **int**[n];
8. **for**(**int** i=0;i<n;i++)
9. {
10. a[i]=scan.nextInt();
11. }
12. scan.close();
13. **for** (**int** i = 0; i < a.length; i++) {
14. System.out.println(a[i]);
15. }
16. }
17. }

**3. Question “Java Abstract Class”**



**Answer**

**import** java.util.\*;

**abstract** **class** Book{

    String title;

**abstract** **void** setTitle(String s);

    String getTitle(){

**return** title;

    }

}

**class** MyBook **extends** Book{

**void** setTitle(String s) {

        title = s;

     }

}

**public** **class** Main{

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

*//Book new\_novel=new Book(); This line prHMain.java:25: error: Book is abstract; cannot be instantiated*

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

        String title=sc.nextLine();

        MyBook new\_novel=**new** MyBook();

        new\_novel.setTitle(title);

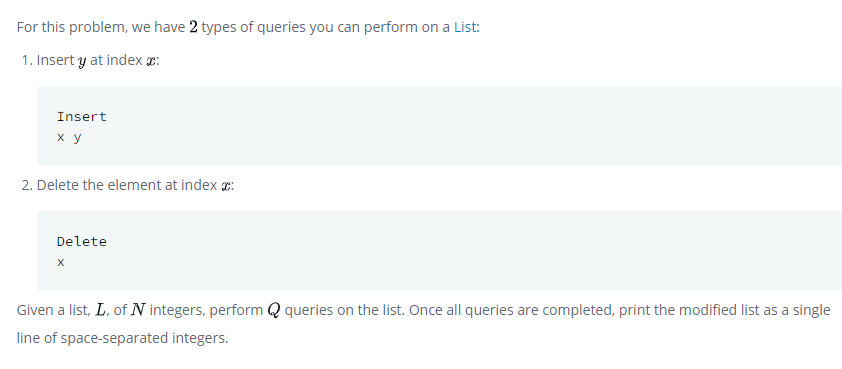
        System.out.println("The title is: "+new\_novel.getTitle());

        sc.close();

    }

}

**4. Question “Java List”**



**Answer**

**import** java.io.\*;

**import** java.util.\*;

**import** java.text.\*;

**import** java.math.\*;

**import** java.util.regex.\*;

**public** **class** Solution {

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

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

**int** N = scan.nextInt();

        LinkedList<Integer> list = **new** LinkedList<>();

**for** (**int** i = 0; i < N; i++) {

**int** value = scan.nextInt();

            list.add(value);

        }

**int** Q = scan.nextInt();

**for** (**int** i = 0; i < Q; i++) {

            String action = scan.next();

**if** (action.equals("Insert")) {

**int** index = scan.nextInt();

**int** value = scan.nextInt();

                list.add(index, value);

            } **else** {

**int** index = scan.nextInt();

                list.remove(index);

            }

        }

        scan.close();

**for** (Integer num : list) {

            System.out.print(num + " ");

        }

    }

}

**5. Question “Java Method Overriding”**



**Answer**

**import** java.util.\*;

**class** Sports{

    String getName(){

**return** "Generic Sports";

    }

**void** getNumberOfTeamMembers(){

        System.out.println( "Each team has n players in " + getName() );

    }

}

**class** Soccer **extends** Sports{

    @Override

    String getName(){

**return** "Soccer Class";

    }

 @Override

**void** getNumberOfTeamMembers(){

        System.out.println( "Each team has 11 players in " + getName() );

    }

}

**public** **class** Solution{

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

        Sports c1 = **new** Sports();

        Soccer c2 = **new** Soccer();

        System.out.println(c1.getName());

        c1.getNumberOfTeamMembers();

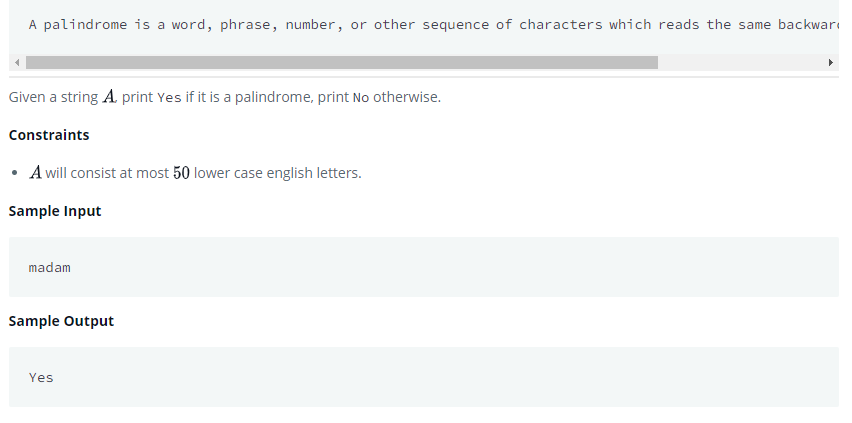
        System.out.println(c2.getName());

        c2.getNumberOfTeamMembers();

    }

}

**6. Question “Java String Reverse”**



**Answer**

**import** java.io.\*;

**import** java.util.\*;

**public** **class** Solution {

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

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

        String A= sc.next();

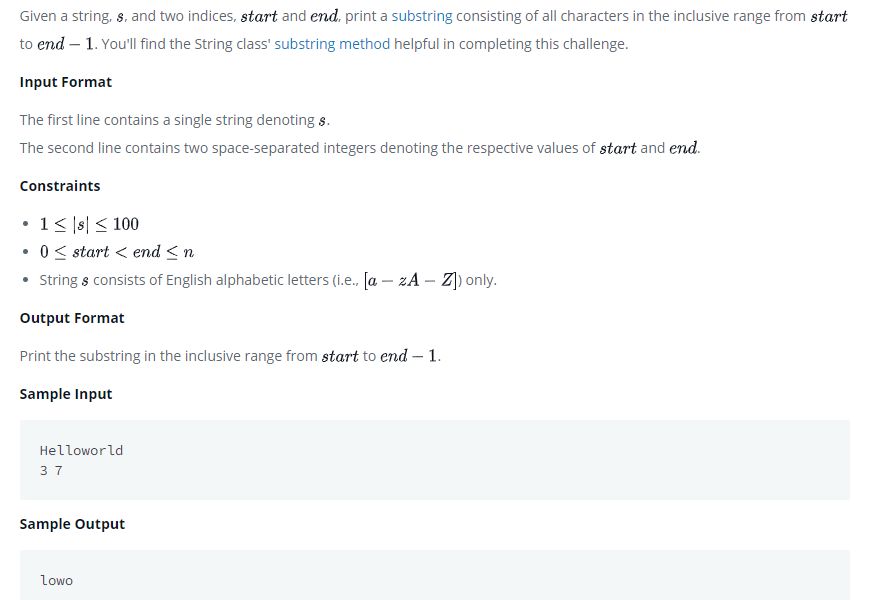
         System.out.println(A.equalsIgnoreCase(**new** StringBuilder(A).reverse().toString())

          ? "Yes" : "No");

    }

}

**7. Question “Java Substring”**



**Answer**

**import** java.io.\*;

**import** java.util.\*;

**import** java.text.\*;

**import** java.math.\*;

**import** java.util.regex.\*;

**public** **class** Solution {

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

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

        String S = in.next();

**int** start = in.nextInt();

**int** end = in.nextInt();

        System.out.println(S.substring(start, end));

    }

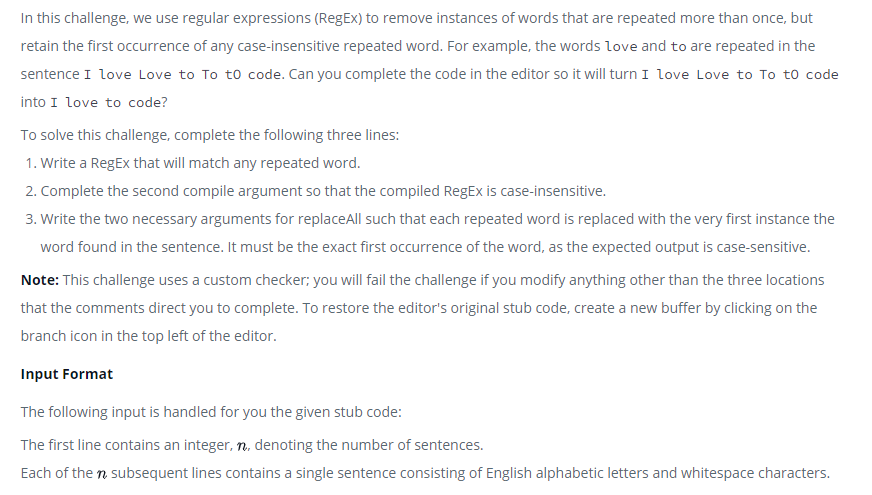
}

**8.**

**NAMA : JAMALUDIN**

**TYPE : MEDIUM**

**1.Question “Java Regex 2”**



**Answer**

**import** java.util.Scanner;

**import** java.util.regex.Matcher;

**import** java.util.regex.Pattern;

**public** **class** DuplicateWords {

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

        String regex = "\\b(\\w+)(?:\\W+\\1\\b)+";

        Pattern p = Pattern.compile(regex,  Pattern.CASE\_INSENSITIVE);

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

**int** numSentences = Integer.parseInt(in.nextLine());

**while** (numSentences-- > 0) {

            String input = in.nextLine();

            Matcher m = p.matcher(input);

*// Check for subsequences of input that match the compiled pattern*

**while** (m.find()) {

                input = input.replaceAll(m.group(), m.group(1) );

            }

*// Prints the modified sentence.*

            System.out.println(input);

        }

        in.close();

    }

}