# Java Datatypes (Easy)

Given an input integer, you must determine which primitive data types are capable of properly storing that input.

Case link: https://www.hackerrank.com/challenges/java-datatypes/problem

import java.util.\*;

import java.io.\*;

class Solution{

    public static void main(String []argh)

    {

        Scanner sc = new Scanner(System.in);

        int t=sc.nextInt();

        for(int i=0;i<t;i++)

        {

            int n = 2;

            try

            {

                long x=sc.nextLong();

                System.out.println(x+" can be fitted in:");

                if(x>=-128 && x<=127)System.out.println("\* byte");

                if(x >= -Math.pow(n, 15) && x <= Math.pow(n, 15) - 1)

                System.out.println("\* short");

                if(x >= -Math.pow(n, 31) && x <= Math.pow(n, 31) - 1)

                System.out.println("\* int");

                if(x >= -Math.pow(n, 63) && x <= Math.pow(n, 63) - 1)

                System.out.println("\* long");

            }

            catch(Exception e)

            {

                System.out.println(sc.next()+" can't be fitted anywhere.");

            }

        }

    }

}

# Java End-of-file (Easy)

The challenge here is to read “n” lines of input until you reach EOF, then number and print all “n” lines of content.

Case link: https://www.hackerrank.com/challenges/java-end-of-file/problem

import java.io.\*;

import java.util.\*;

public class Solution {

    public static void main(String[] args) {

        /\* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. \*/

        Scanner input = new Scanner(System.in);

        int i=0;

        while(input.hasNextLine()){

            i++;

            System.out.println(i+" "+input.nextLine());

        }

    }

}

**Welcome to Java!** (Easy)

Welcome to the world of Java! In this challenge, we practice printing to stdout.

The code stubs in your editor declare a *Solution* class and a *main* method. Complete the *main* method by copying the two lines of code below and pasting them inside the body of your *main* method.

Case link: https://www.hackerrank.com/challenges/welcome-to-java/problem

public class Solution {

    public static void main(String[] args) {

        /\* Enter your code here. Print output to STDOUT. Your class should be named Solution. \*/

        System.out.println("Hello, World.");

        System.out.println("Hello, Java.");

    }

}

# Java Stdin and Stdout I (Easy)

In this challenge, you must read ”3” integers from stdin and then print them to stdout. Each integer must be printed on a new line. To make the problem a little easier, a portion of the code is provided for you in the editor below.

Case link: https://www.hackerrank.com/challenges/java-stdin-and-stdout-1/problem

import java.util.\*;

public class Solution {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);

        int a = scan.nextInt();

        // Complete this line

        // Complete this line

        int b = scan.nextInt();

        int c = scan.nextInt();

        System.out.println(a);

        // Complete this line

        // Complete this line

        System.out.println(b);

        System.out.println(c);

    }

}

# Java Static Initializer Block (Easy)

# 

Case link: <https://www.hackerrank.com/challenges/java-static-initializer-block/problem>

import java.io.\*;

import java.util.\*;

import java.text.\*;

import java.math.\*;

import java.util.regex.\*;

public class Solution {

static int B;

static int H;

static boolean flag;

    static{

    Scanner input = new Scanner(System.in);

    B = input.nextInt();

    H = input.nextInt();

    flag = true;

    input.close();

     if(B>0 && H>0){

         flag = true;

     } else if((B<=0 && H>=0)||(B>=0 && H<=0))

     { flag=false; System.out.println("java.lang.Exception: Breadth and height must be positive"); } else

     { flag=false; System.out.println("java.lang.Exception: Breadth and height must be positive"); }

    }

public static void main(String[] args){

        if(flag){

            int area=B\*H;

            System.out.print(area);

        }

    }//end of main

}//end of class

# Java If-Else (Easy)

# 

# Case link: https://www.hackerrank.com/challenges/java-if-else/problem

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 awal {

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

    public static void main(String[] args) {

        int N = scanner.nextInt();

        scanner.skip("(\r\n|[\n\r\u2028\u2029\u0085])?");

        scanner.close();

        if(N % 2 == 1) {

            System.out.println("Weird");

        }

        else if (N % 2 == 0 && N >=2 && N <=5) {

            System.out.println("Not Weird");

        }

        else if (N % 2 == 0 && N >=6 && N <=20) {

            System.out.println("Weird");

        }

        else if (N % 2 == 0 && N > 20) {

            System.out.println("Not Weird");

        }

    }

}

# Java Stdin and Stdout II (Easy)

In this challenge, you must read an integer, a double, and a String from stdin, then print the values according to the instructions in the Output Format section below. To make the problem a little easier, a portion of the code is provided for you in the editor.

Case link: https://www.hackerrank.com/challenges/java-stdin-stdout/problem

import java.util.Scanner;

    public class awal {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        int i = input.nextInt();

        double d = input.nextDouble();

        input.nextLine();

        String s = input.nextLine();

        // Write your code here.

        System.out.println("String: " + s);

        System.out.println("Double: " + d);

        System.out.println("Int: " + i);

    }

}

# Java Regex (Medium)

Write a class called *MyRegex* which will contain a string pattern. You need to write a regular expression and assign it to the pattern such that it can be used to validate an IP address. Use the following definition of an IP address:

IP address is a string in the form "A.B.C.D", where the value of A, B, C, and D may range from 0 to 255. Leading zeros are allowed. The length of A, B, C, or D can't be greater than 3.

Case link: https://www.hackerrank.com/challenges/java-regex/problem

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import java.util.Scanner;

class Solution{

    public static void main(String[] args){

        Scanner in = new Scanner(System.in);

        while(in.hasNext()){

            String IP = in.next();

            System.out.println(IP.matches(new MyRegex().pattern));

        }

    }

}

//Write your code here

class MyRegex{

    String regex = "([01]?\\d{1,2}|2[0-4]\\d|25[0-5])";

    String pattern = regex+"."+regex+"."+regex+"."+regex;

}

# Java Loops I (Easy)

# 

Case link: https://www.hackerrank.com/challenges/java-loops-i/problem

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();

        scanner.skip("(\r\n|[\n\r\u2028\u2029\u0085])?");

        for(int i=1; i<=10; i++) {

            System.out.println(N+" x "+i+" = "+N\*i);

        }

        scanner.close();

    }

}