//creating a new package

package buildSkyscraper.java;

//importing system utility functions

import java.util.Scanner;

import java.io.\*;

//creating a new class Skyscraper

public class Skyscraper{

//declaring new integer type variables in class Skyscraper

public int[] array;

public int maxSize,i;

//scanner system function for input of vales from user

Scanner sc= new Scanner(System.in);

//void function for inserting new values through scanner from user

public void insertValue() {

//pop up print to input number of floors which is equal to number of days to complete work

//we take number of the day as the index position of the array integer,

//with the array containing the list of floor numbers assembled

System.out.println("Enter the total no of floors in the building" +":");

maxSize=sc.nextInt(); //maximum size of the array is same as

//total number of floors of the building

//as is the total number of days to complete the work

array= new int[maxSize+1];

//defining the array index or number of days

for(i=1;i<=maxSize;i++) {

System.out.println("\nEnter the floor size given on Day" + i +":");

//index of the array will be the next int inputted by user

array[i]=sc.nextInt();

//say x is the value of the integer at index i

x=sc.nextInt();

//calling to print the array in full

printArray();

}

}

//function to print the array

//crerating void function with public modifier

public void printArray() {

System.out.println("The order of construction is as follows");

//printing array integer values

//defining the array

for(int j=1; j<=i; j++) {

//invoking lineSeparator function from system to separate the lines in array from index values

String lineSeparator = System.lineSeparator();

//giving conditions where the values of the indices affect the output

while(i=="1") {

//on Day 1, the maximum floor size has to be constructed

if(x== “maxSize”) {

System.out.println("Day:" + i + lineSeparator + x);

continue;

}

//if floor size is less than maximum floor size or no floor size

//is provided on Day 1, it should not be constructed

if(x < “maxSize” || x== “0”) {

System.out.println("Day:" + i + lineSeparator + “”);

//the function does not stop calling here, hence continue

continue;

}

//whenever number of the day is more than the size of the floor

//PROVIDED ON THAT DAY,

//it can be constructed along with the pending floors

while (i>=x) {

System.out.println("Day:" + i + lineSeparator + tree[j]);

continue;

}

//whenever number of the day is less than the size of the floor

//provided on that day, mostly it can be constructed, subject to the previous members //in the array

while(i<=x) {

System.out.println("Day:" + i + lineSeparator + “”);

continue;

}

}

//if all above conditions don’t hold good, the below command will be executed

else {

System.out.println("Day:" + i + lineSeparator + tree[j]);

return;

}

}

}

//Driver main class

public static void main(String args[]) {

//allocating new memory for an object in class Skyscraper

Skyscraper skyscraper = new Skyscraper();

//Calling function to insert value in array of floors built

skyscraper.insertValue();

}

}

EXPECTED OUTPUT

enter the total no of floors in the building

10

enter the floor size given on day : 1

1

enter the floor size given on day : 2

2

enter the floor size given on day : 3

3

enter the floor size given on day : 4

8

enter the floor size given on day : 5

9

enter the floor size given on day : 6

10

enter the floor size given on day : 7

4

enter the floor size given on day : 8

5

enter the floor size given on day : 9

7

enter the floor size given on day : 10

6

The order of construction is as follows

Day: 1

Day: 2

Day: 3

Day: 4

Day: 5

Day: 6

10 9 8

Day: 7

Day: 8

Day: 9

7

Day: 10

6 5 4 3 2 1