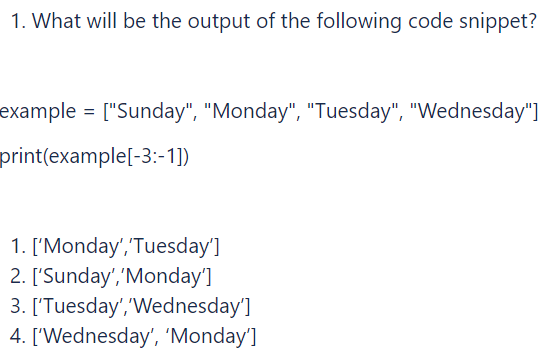
Set2

1)

p-

Solution :-

Output :- 1) [“Monday”,”tuesday”]

2) Define and implement a class(es), representing binary tree with integer-values nodes, e.g.

             5

           /       \

       3            2

    /     \

10        11

Define and implement methods:

* find maximum value
* calculate tree depth

Solution :-

class HelloWorld {

static int maxVal=Integer.MIN\_VALUE;

public static void insert(Node node, int value) {

if (value < node.value) { if (node.left != null) { insert(node.left, value); } else {

node.left = new Node(value); }

} else if (value > node.value) {

if (node.right != null) {

insert(node.right, value);

} else {

node.right = new Node(value);

}

}

}

public static int depth(Node root){

if(root==null){

return 0;

}

maxVal=Math.max(maxVal,root.value);

return 1+Math.max(depth(root.left),depth(root.right));

}

public static void main(String[] args) {

Node tree = new Node(25);

insert(tree,50);

insert(tree,30);

insert(tree,20);

insert(tree,40);

insert(tree,70);

insert(tree,60);

insert(tree,80);

int d=depth(tree);

System.out.println("Depth of tree is :- "+d);

System.out.println("Max Value of tree is :- "+maxVal);

}

}

class Node {

int Node;

Node left, right;

int value;

Node(int value){

this.value = value;

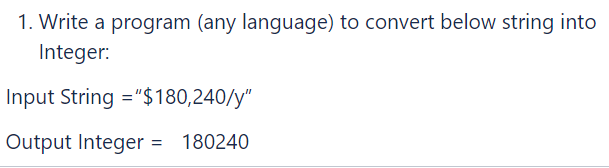
left = null;

right = null;

}

}

3)



// Online Java Compiler

// Use this editor to write, compile and run your Java code online

import java.util.Scanner;

class HelloWorld {

public static void main(String[] args) {

Scanner sc=new Scanner(System.in);

String s=sc.nextLine();

int ans=0;

for(int i=0;i<s.length();i++){

if(Character.isDigit(s.charAt(i))){

ans=ans\*10+(s.charAt(i)-'0');

}

}

System.out.println(ans);

}

}