JAVA PRACTICE PROGRAM

Submitted by,

Tamilvanan B.

2018503566.

MO Batch.

11-19-2020.

Write a simple Java program that fetches the students course-wise marks from a university database and computes the grade for each student using JDBC concept.

Program

import java.sql.\*;

public class Database{

static String DRIVER = "com.mysql.cj.jdbc.Driver";

static String DBMS\_URL = "jdbc:mysql://localhost/university";

static String username = "root";

static String password = "";

static Connection conn = null;

public static void main(String… args) throws Exception{

Class.forName(DRIVER);

conn = DriverManager.getConnection(DBMS\_URL, username, password);

float[] grades = Student.getMarks();

updateGrades(grades);

System.out.println("Grades Successfully Updated!");

}

public static void updateGrades(float[] grades) throws Exception{

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

String query = "Update marksheet set grades = " + grades[i] + " where id = " + (i + 1);

Statement stmt = conn.createStatement();

stmt.executeUpdate(query);

}

}

}

class Student{

public static float[] getMarks() throws Exception{

float[] grades;

Statement stmt = Database.conn.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

String query = "Select \* from marksheet";

ResultSet rs = stmt.executeQuery(query);

rs.last();

int size = rs.getRow();

rs.beforeFirst();

grades = new float[size];

int i = 0;

int k = 1;

while(rs.next()){

String name = rs.getString("name");

float English = rs.getFloat("english");

float Language = rs.getFloat("language");

float Maths = rs.getFloat("maths");

float grade = makeGrade(English, Language, Maths);

grades[i++] = grade;

System.out.println(k + "." + name + " - " + grade);

k++;

}

return grades;

}

public static float makeGrade(float x, float y, float z){

x /= 100;

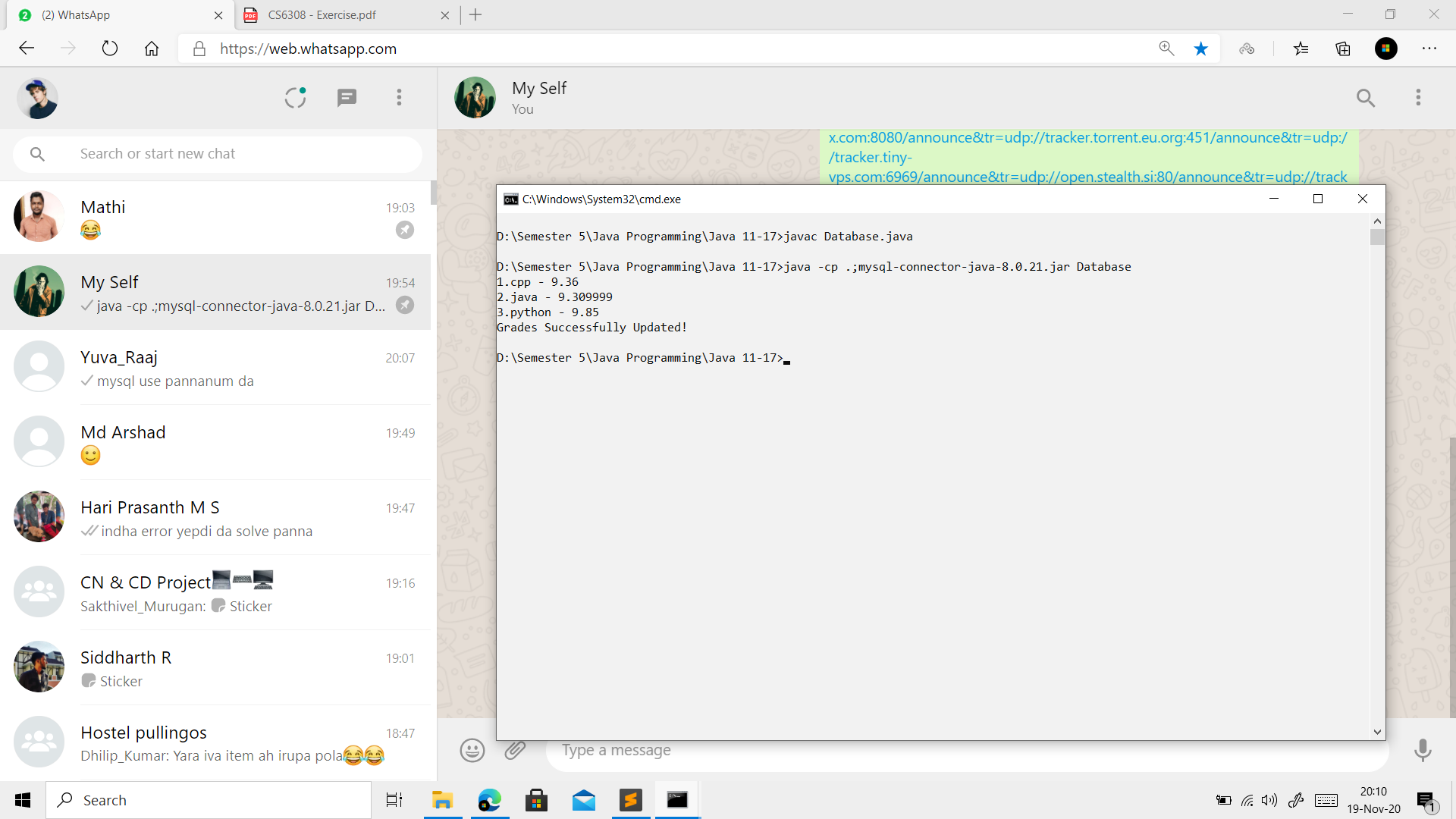
y /= 100;

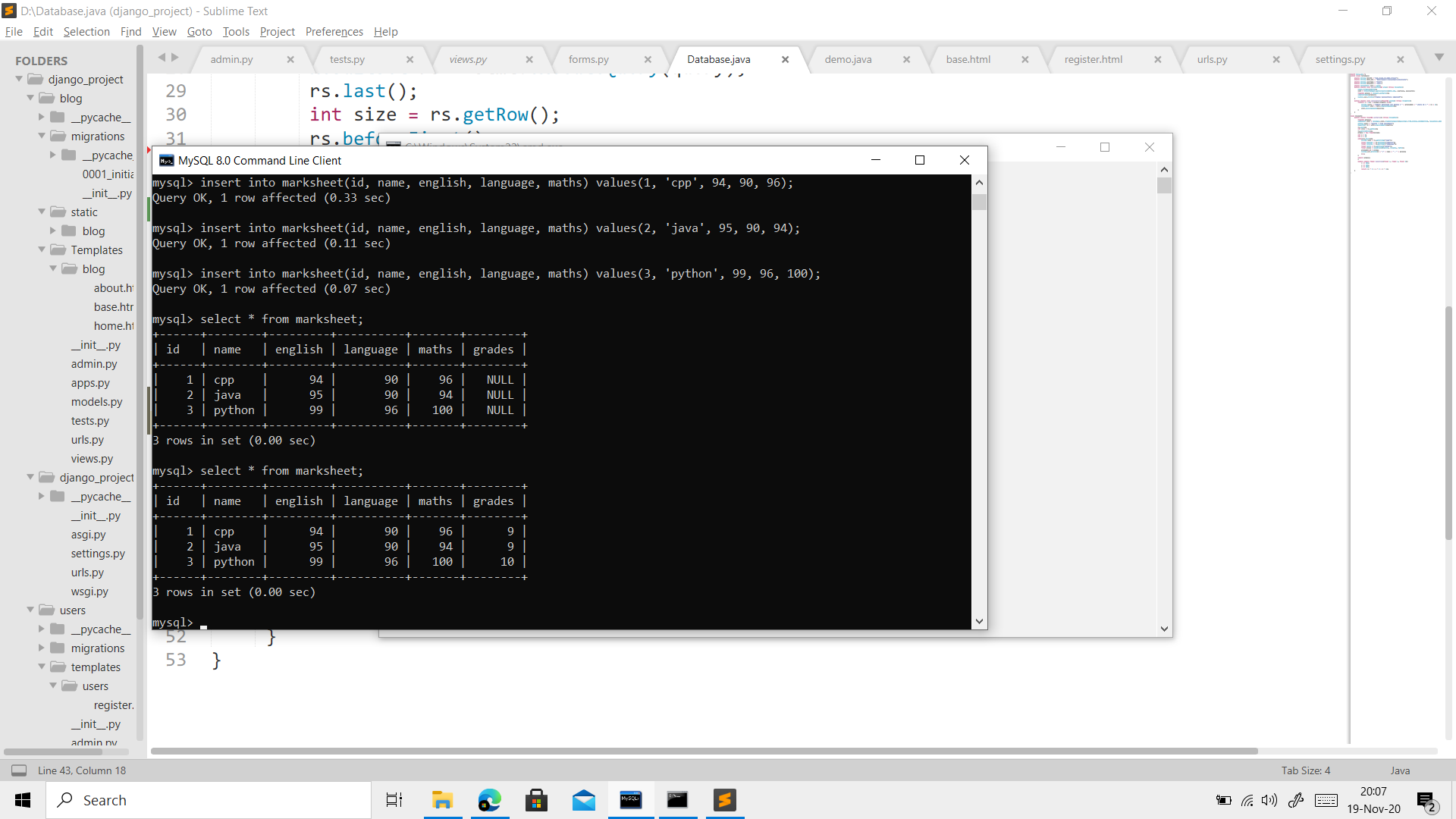
z /= 100;

return (x \* 3 + y \* 3 + z \* 4);

}

}

Output



Create a distributed application in JAVA using RMI concept to find the presence of number in a given string. The program should use the concept of remote object invocation to implement this process.

Program

Client.java

import java.rmi.\*;

import java.rmi.registry.\*;

import java.util.Scanner;

public class Client {

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

public static void main(String… args) throws Exception{

System.out.print("\n\nEnter the string to search for numbers : ");

String s = sc.nextLine();

Registry reg = LocateRegistry.getRegistry(null);

RemoteInterface stub = (RemoteInterface) reg.lookup("RemoteInterface");

int[] ans = stub.findNumbers(s);

int i = 0;

while(ans[i] != -1){

System.out.println("" + (i + 1) + " Number : " + ans[i]);

i++;

}

if(i == 0)

System.out.println("No Numbers Found!");

System.out.print("\n\n");

}

}

Server.java

import java.rmi.\*;

import java.rmi.registry.\*;

import java.rmi.server.\*;

public class Server{

public static void main(String… args) throws Exception{

RemoteClass obj = new RemoteClass();

RemoteInterface stub = (RemoteInterface) UnicastRemoteObject.exportObject(obj, 0);

Registry reg = LocateRegistry.getRegistry();

reg.bind("RemoteInterface", stub);

}

}

class RemoteClass implements RemoteInterface{

public int[] findNumbers(String input) throws Exception{

int[] ans = new int[20];

int index = 0;

String temp = "";

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

char now = input.charAt(i);

if(isNumber(now))

temp += now;

else if(temp.length() > 0){

ans[index++] = Integer.parseInt(temp);

temp = "";

}

}

ans[index++] = -1;

return ans;

}

boolean isNumber(char x){

return (x >= '0' && x <= '9');

}

}

RemoteInterface.java

import java.rmi.Remote;

import java.rmi.RemoteException;

public interface RemoteInterface extends Remote{

int[] findNumbers(String input) throws Exception;

}

Output

