OOPS PRACTICAL - 5 & 6

Name : Prathamesh Rajbhoj

Branch : CSE

Roll No : A - 53

Aim :

5. Create a class IPL. The datamembers are Team\_Name, total matches played, number of matches won and lost. Create a method Register\_Team, which will take Team Name as a Parameter. If the name contains any alphabet or number it will throw a user defined exception “InvalidTeamName”. The exception class will display the message which parameter is not satisfied.

Create a method update count, which will update the number of matches won and lost. If the sum of number of matches won and lost is not equal to total matches played then throw user defined exception “InvalidCountException”. If the numbers of matches lost are more than 40% then display the message you are disqualified.

Use proper Exception handling mechanism to handle the exception.

6. Create a class IPL. The data Team\_Name, number of matches won and lost is written in the file name “IPL.txt”. Read the file and store the data in object. Create array of objects and demonstrate all functionalities of IPL Class defined in Problem definition 5.

**Question - 1**

**IPL.java :**

public class IPL {

String team\_name;

int total\_matches;

int won;

int lost;

boolean registered=false;

IPL() {

}

IPL(String team\_name, int total\_matches, int won, int lost) {

this.team\_name = team\_name;

this.total\_matches = total\_matches;

this.won = won;

this.lost = lost;

}

void DisplayTeam() {

if(this.registered==false){

System.out.println("Team " + this.team\_name + "is not registered\n");

}

System.out.println();

System.out.println("Team Name : " + this.team\_name);

System.out.println("Total Matches : " + this.total\_matches);

System.out.println("Total Won : " + this.won);

System.out.println("Total Loss : " + this.lost);

System.out.println();

}

void Register\_Team() {

try {

String name = this.team\_name;

int n = name.length();

System.out.print(" ---> ");

for (int i = 0; i < n; i++) {

char c = name.charAt(i);

if (c >= 'a' && c <= 'z') {

throw new InvalidTeamName("Alphabet Not Allowed in Team Name - { " + this.team\_name + " }\n");

}

else if (c >= 'A' && c <= 'Z') {

throw new InvalidTeamName("Alphabet Not Allowed in Team Name - { " + this.team\_name + " }\n");

}

else if (c >= '0' && c <= '9') {

throw new InvalidTeamName("Numbers Not Allowed in Team Name - { " + this.team\_name + " }\n");

}

}

this.registered=true;

System.out.println("Team { " + this.team\_name + " } Registered Succesfully\n");

} catch (InvalidTeamName e) {

e.printStackTrace();

}

}

void updateCount(String str) {

this.total\_matches++;

if (str.equals("Won")) {

this.won++;

} else {

this.lost++;

}

try {

if (this.total\_matches != (this.won + this.lost)) {

throw new InvalidCountException("\nEXCEPTION -- INVALID STATISTICS\n");

} else {

System.out.println("\n Team " + this.team\_name + " Statistics Verified\n");

double lost\_percentage = ((this.lost\* 100.00) / this.total\_matches);

// System.out.println(this.lost);

// System.out.println(this.total\_matches);

// System.out.println(lost\_percentage + "% \n");

if (lost\_percentage > 40) {

System.out.println("\n^^^^ TEAM " + this.team\_name + " DISQUALIFIED ^^^^\n");

return;

}

}

} catch (InvalidCountException e) {

e.printStackTrace();

}

}

}

class InvalidTeamName extends Exception {

InvalidTeamName(String msg) {

super(msg);

System.out.println();

}

}

class InvalidCountException extends Exception {

InvalidCountException(String msg) {

super(msg);

System.out.println();

}

}

**IPL.txt :**

& 6 6 0

&&ABC 8 4 4

&&& 8 5 3

@ 8 0 8

@@ 5 4 1

@@@ 5 3 2

# 6 5 1

##12 10 8 2

### 10 2 8

####a1B 4 1 3

**Test.java :**

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

public class Test {

public static void main(String[] args) throws IOException {

System.out.println();

System.out.println();

FileReader fr = null;

BufferedReader br = null;

try {

fr = new FileReader("IPL.txt");

br = new BufferedReader(fr);

String line;

IPL team[] = new IPL[10];

int index = 0;

while ((line = br.readLine()) != null) {

String arr[] = line.split(" ");

int a = Integer.parseInt(arr[1]);

int b = Integer.parseInt(arr[2]);

int c = Integer.parseInt(arr[3]);

team[index] = new IPL(arr[0], a, b, c);

team[index].Register\_Team();

index++;

}

System.out.println();

System.out.println();

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

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Displaying All Teams \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

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

for (int i = 0; i < 10; i++) {

team[i].DisplayTeam();

}

System.out.println();

System.out.println();

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

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Performing Operations \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

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

System.out.println();

System.out.println();

team[3].updateCount("Won");

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

team[0].updateCount("Loss");

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

team[7].updateCount("Won");

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

} finally {

fr.close();

br.close();

}

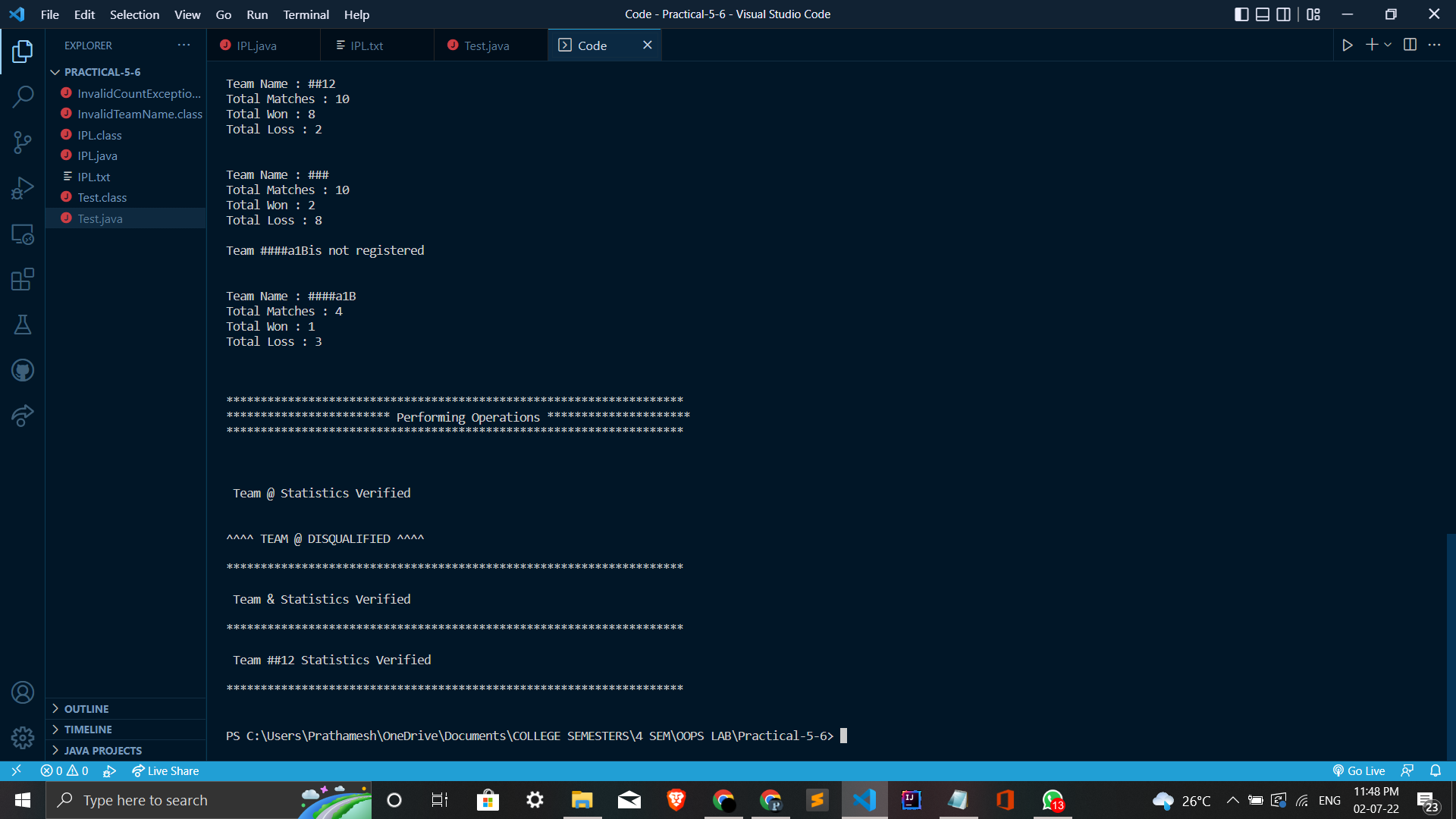
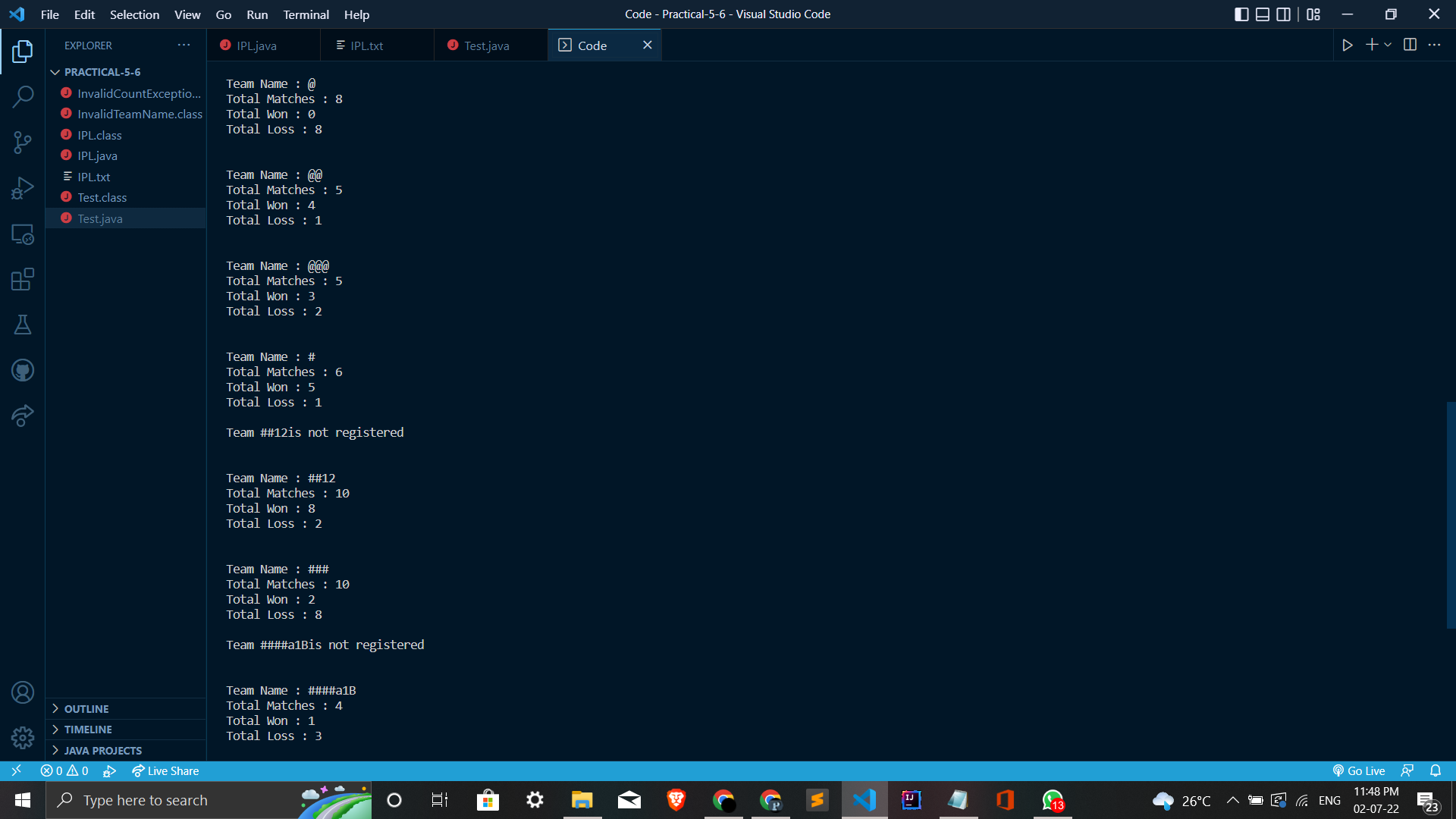
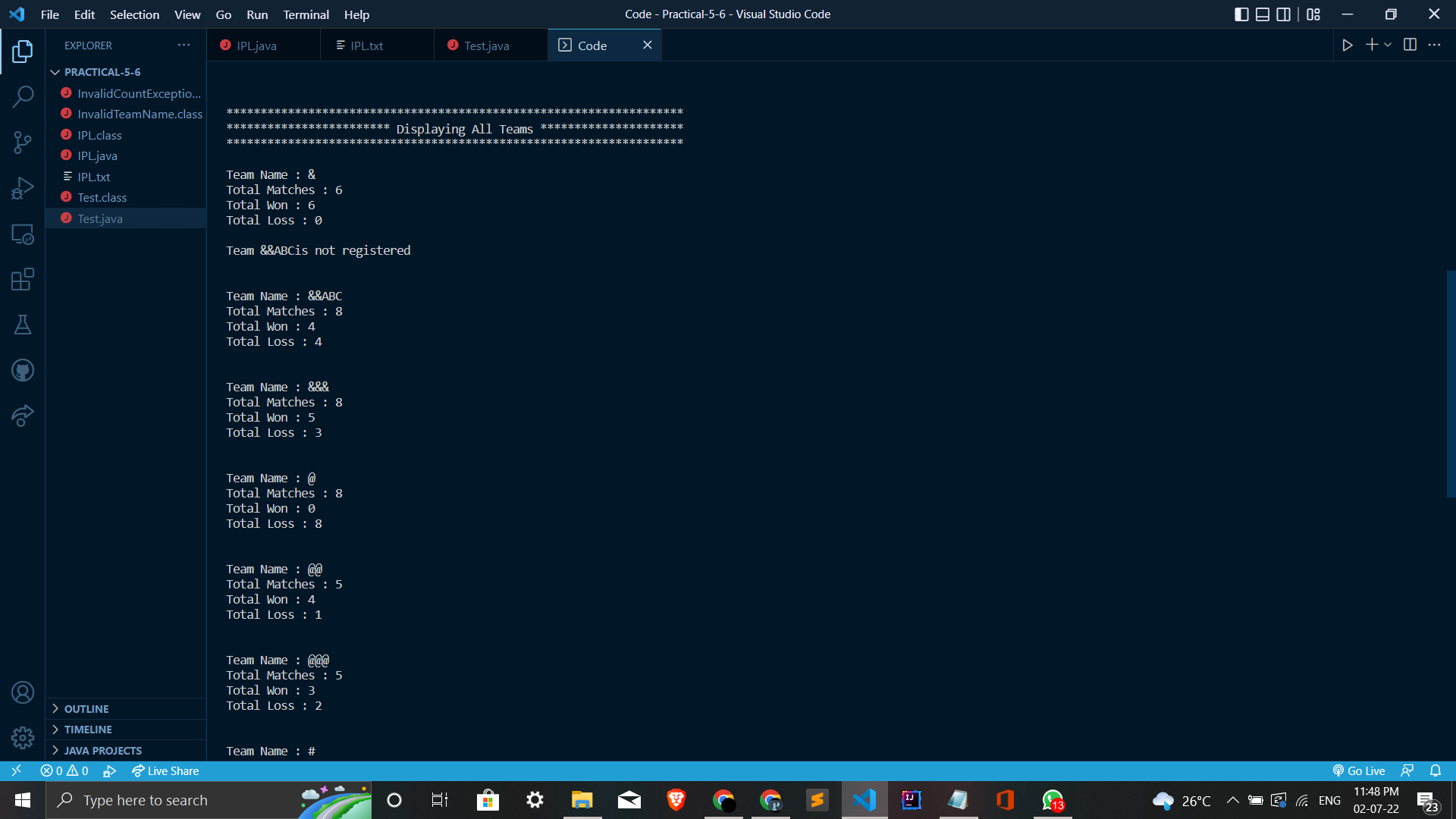
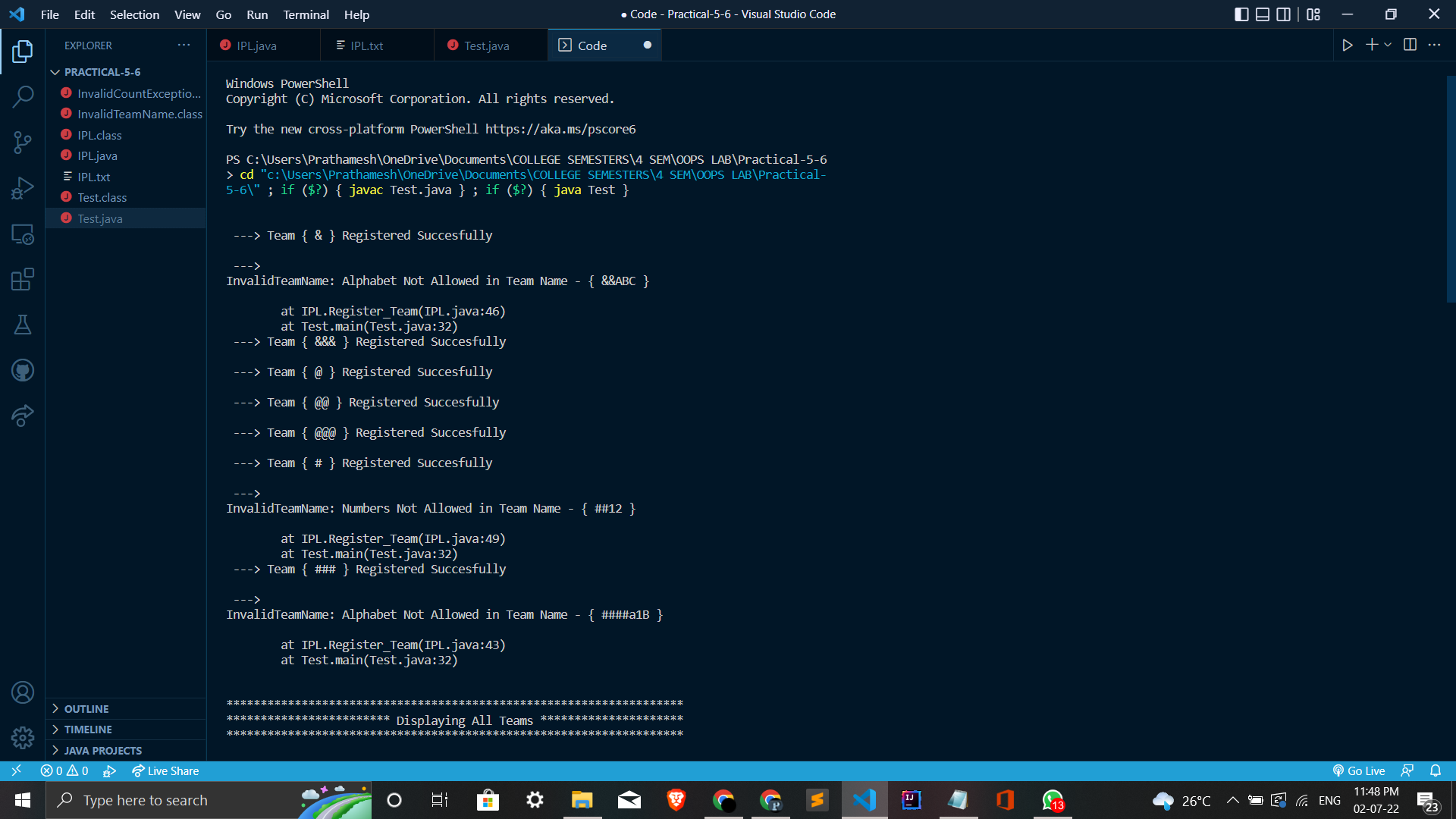
System.out.println();

System.out.println();

}

}

**Output :**

****