2 questions Total score: 100

1 Programming question

1. Character at K + 50.0

1 Java project question

2. Shapes + 50.0

If *N = 3 , str = [abcc, trea, zape], Q = 2.*

- Question 1 : *L = 2, R = 3, K = 5.* Concatenated string : treazape. Concatenated string in sorted order : aaeeprtz. 5th character is *p.*
- Question 2 : *L = 2, R = 2, K = 1.* Concatenated string : trea. Concatenated string in sorted order : aert . 1st character is a.

Function Description

Complete the char_at_K function provided in the editor. The function takes the following 4 parameters and returns K^{th} character of the concatenated string.

- N : Represents the number of strings
- str : Represents the list of *N* strings
- Q : Represents the number of questions
- query : Represents Q questions each containing three integers L, R and K.

Input format

- First line: N (number of strings in the list)
- Next N lines: String S_i
- Next line: Q (number of questions)
- Next Q lines: Three space-separated integers, L, R, and K

Output format

For each question, print the K^{th} character of the concatenated string in a new line.

Constraints

$1 \leq N \leq 10^5$

$1 < len(S_i) < 50$

2 questions Total score: 100

1 Programming question

1. Character at K + 50.0

1 Java project question

2. Shapes + 50.0

Question 1 Max. score: 50.00

Character at K

You are given a list that contains N strings of lowercase English alphabets. Any number of contiguous strings can be grouped together to form a new string. The grouping function accepts two integers X and Y and concatenates all strings between indices X and Y (inclusive) and returns a modified string in which the alphabets of the concatenated string are sorted.

You are asked Q questions each containing two integers L and R. Determine the K^{th} character in the concatenated string if we pass L and R to the grouping function.

Example

If *N = 3 , str = [abcc, trea, zape], Q = 2.*

- Question 1 : *L = 2, R = 3, K = 5.* Concatenated string : treazape. Concatenated string in sorted order : aaeeprtz. 5th character is *p.*
- Question 2 : *L = 2, R = 2, K = 1.* Concatenated string : trea. Concatenated string in sorted order : aert . 1st character is a.

Function Description

Complete the char_at_K function provided in the editor. The function takes the following 4 parameters and returns K^{th} character of the concatenated string.

- N : Represents the number of strings
- str : Represents the list of *N* strings
- Q : Represents the number of questions
- query : Represents Q questions each containing three integers L, R and K.

Input format

- First line: N (number of strings in the list)

2 questions Total score: 100

1 Programming question

1. Character at K + 50.0

1 Java project question

2. Shapes + 50.0

Input format

- First line: N (number of strings in the list)
- Next N lines: String S_i
- Next line: Q (number of questions)
- Next Q lines: Three space-separated integers, L, R, and K

Output format

For each question, print the K^{th} character of the concatenated string in a new line.

Constraints

$1 \leq N \leq 10^5$

$1 \leq len(S_i) \leq 50$

$1 \leq Q \leq 10^5$

$1 \leq L \leq R \leq N$

It is always guaranteed that the K^{th} position is valid.

Sample input 1 Copy

```
5
aaaaa
bbbbb
ccccc
ddddd
eeeee
3
```

Sample output 1 Copy

```
c
d
e
```

2 questions Total score: 100

1 Programming question

1. Character at K + 50.0

1 Java project question

2. Shapes + 50.0

It is always guaranteed that the K^{th} position is valid.

Sample input 1 Copy

```
5
aaaaa
bbbbb
ccccc
ddddd
eeeee
3
3 3 3
1 5 16
3 5 15
```

Sample output 1 Copy

```
c
d
e
```

Explanation

Question 1: Grouped String - *ccccc*. 3rd character is **c**

Question 2: Grouped String - *aaaaabbbbbccccdddddeeeee*. 16th character is **d**

Question 3: Grouped String - *cccccddddeeeee*. 15th character is **e**

Note: Your code must be able to print the sample output from the provided sample input. However, your code is run against multiple hidden test cases. Therefore, your code must pass these hidden test cases to solve the problem statement.

Time Limit: 1.0 sec(s) for each input file
Memory Limit: 256 MB
Source Limit: 1024 KB
Marking Scheme: Score is assigned if any testcase passes
Allowed Languages: Java, Java 8, Java 14

New Submission All Submissions

Save Java (openjdk 1.7.0_95) Full Screen

```
1  import java.io.*;
```

2 questions Total score: 100

1 Programming question

1. Character at K + 50.0

1 Java project question

2. Shapes + 50.0

Save Java (openjdk 1.7.0_95) Full Screen

```
1  import java.io.*;
2  import java.util.*;
3
4
5  public class TestClass {
6      public static void main(String[] args) throws IOException {
7          BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
8          PrintWriter wr = new PrintWriter(System.out);
9           int N = Integer.parseInt(br.readLine().trim());
10          String[] str = new String[N];
11          for(int i_str = 0; i_str < N; i_str++)
12          {
13              str[i_str] = br.readLine();
14          }
15          int Q = Integer.parseInt(br.readLine().trim());
16          int[][] query = new int[Q][3];
17          for(int i_query = 0; i_query < Q; i_query++)
18          {
19              String[] arr_query = br.readLine().split(" ");
20              for(int j_query = 0; j_query < arr_query.length; j_query++)
21              {
22                  query[i_query][j_query] = Integer.parseInt(arr_query[j_query]);
23              }
24          }
25
26          char[] out_ = char_at_K(N, str, Q, query);
```

1:1 vscode

Provide custom input

COMPILE & TEST SUBMIT

2 questions Total score: 100

1 Programming question

1. Character at K + 50.0

1 Java project question

2. Shapes + 50.0

Save Java (openjdk 1.7.0_95) Full Screen

```
16            int[][] query = new int[Q][3];
17            for(int i_query = 0; i_query < Q; i_query++)
18            {
19                String[] arr_query = br.readLine().split(" ");
20                for(int j_query = 0; j_query < arr_query.length; j_query++)
21                {
22                    query[i_query][j_query] = Integer.parseInt(arr_query[j_query]);
23                }
24            }
25
26            char[] out_ = char_at_K(N, str, Q, query);
27            System.out.print(out_[0]);
28            for(int i_out_ = 1; i_out_ < out_.length; i_out_++)
29            {
30                System.out.print("\n" + out_[i_out_]);
31            }
32
33            wr.close();
34            br.close();
35        }
36        static char[] char_at_K(int N, String[] str, int Q, int[][] query){
37            // Write your code here
38
39
40        }
41    }
```

1:1 vscode

Provide custom input

COMPILE & TEST SUBMIT

Next Question >

2 questions Total score: 100

1 Programming question

1. Character at K + 50.0

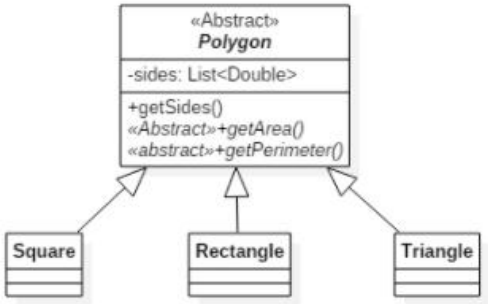
1 Java project question

2. Shapes + 50.0

Question 2 Max. score: 50.00

Shapes

Your task is to write an application that simulates the following shapes:



The **Polygon** is an abstract class that contains the following attribute:

- **sides**: List of integers that represent the length of each side of the **Polygon** class

The following operations can be performed on the **Polygon** class:

- **getSides()**: Returns the list of sides of the polygon
- **area()**: Returns the area of the polygon
- **perimeter()**: Returns the perimeter of the polygon

The three shapes, **Square**, **Rectangle**, and **Triangle** extends the **Polygon** class and implements the abstract methods.

Note:

- Use the names of the classes that are mentioned in the UML diagram
- Created application is tested based on a set of test cases

2 questions Total score: 100

1 Programming question

1. Character at K + 50.0

1 Java project question

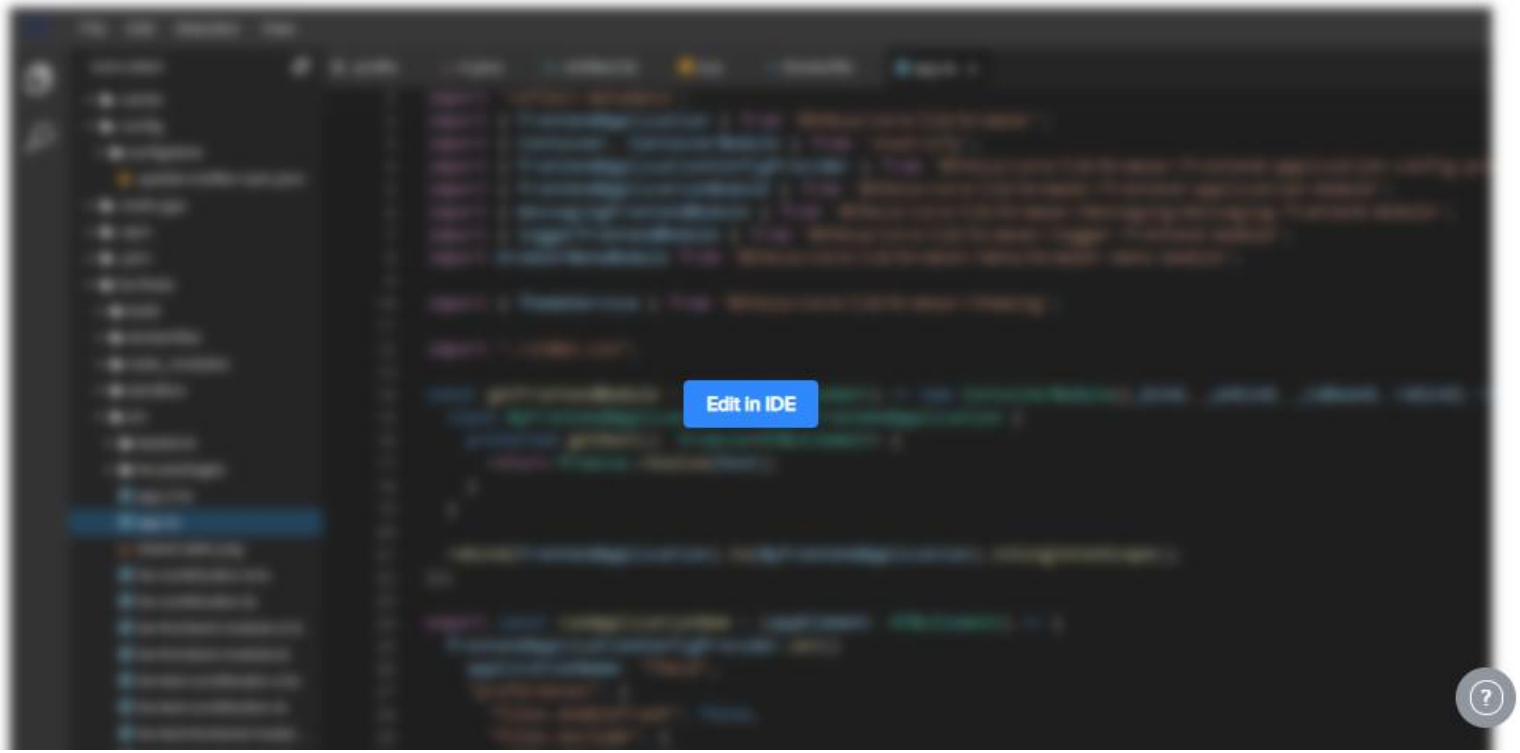
2. Shapes + 50.0

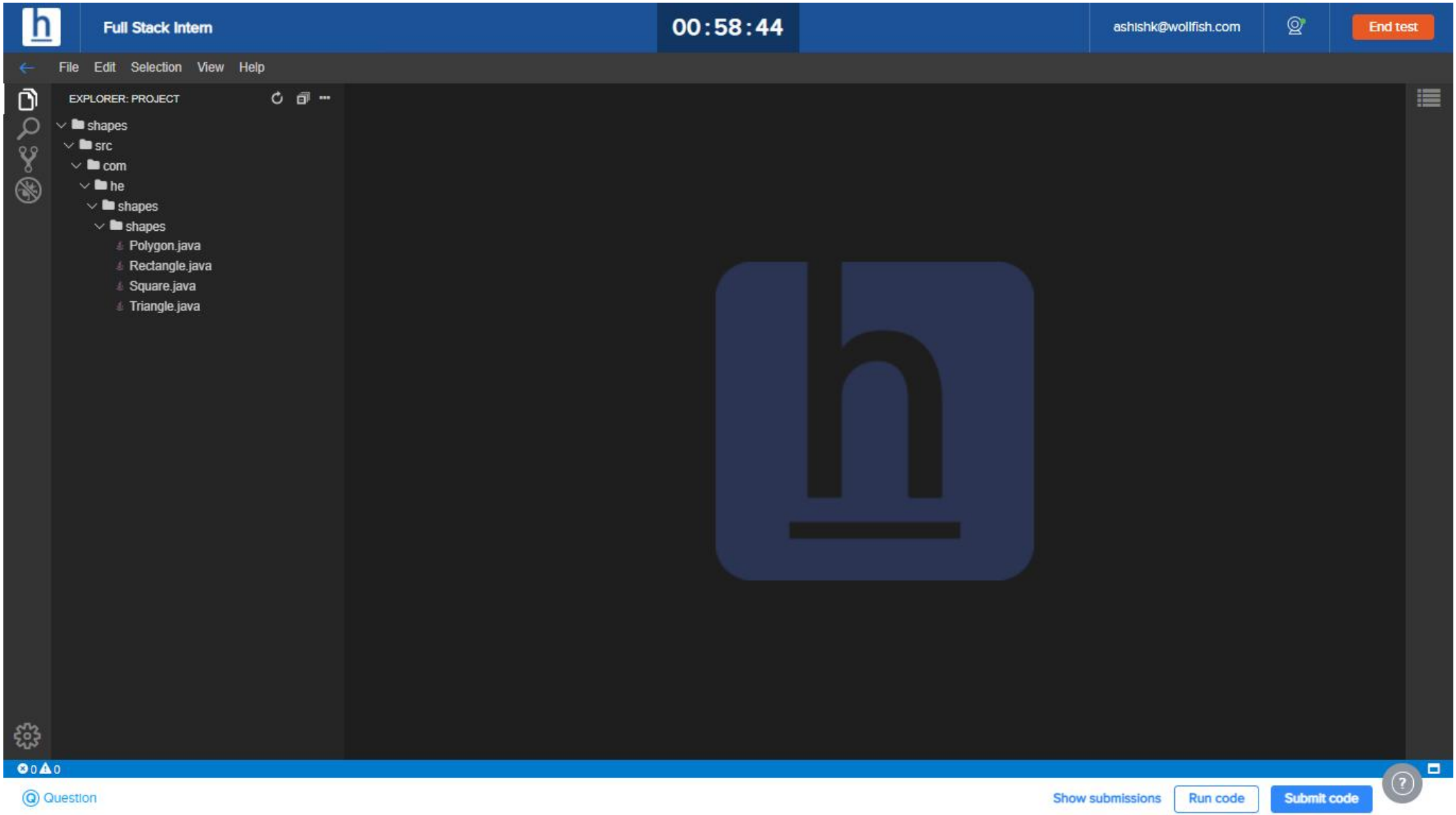
The three shapes, **Square**, **Rectangle**, and **Triangle** extends the **Polygon** class and implements the abstract methods.

Note:

- Use the names of the classes that are mentioned in the UML diagram
- Created application is tested based on a set of test cases
- Test your application with the sample test case that is visible to you
- Once you submit your application, the remaining internal test cases are executed and the score is assigned.
- Do not change the folder structure and class names.

Note: This question will be evaluated based on the number of test cases that your code passes.




Full Stack Intern
00:58:44
ashishk@wollfish.com
End test
File Edit Selection View Help
EXPLORER: PROJECT
shapes
src
com
he
shapes
shapes
Polygon.java
Rectangle.java
Square.java
Triangle.java
0 0
Question
Show submissions
Run code
Submit code

Full Stack Intern

00:58:40

ashishk@wolfish.com

End test

File Edit Selection View Help

EXPLORER: PROJECT

- shapes
  - src
    - com
      - he
        - shapes
          - shapes
            - Polygon.java
            - Rectangle.java
            - Square.java
            - Triangle.java

Polygon.java

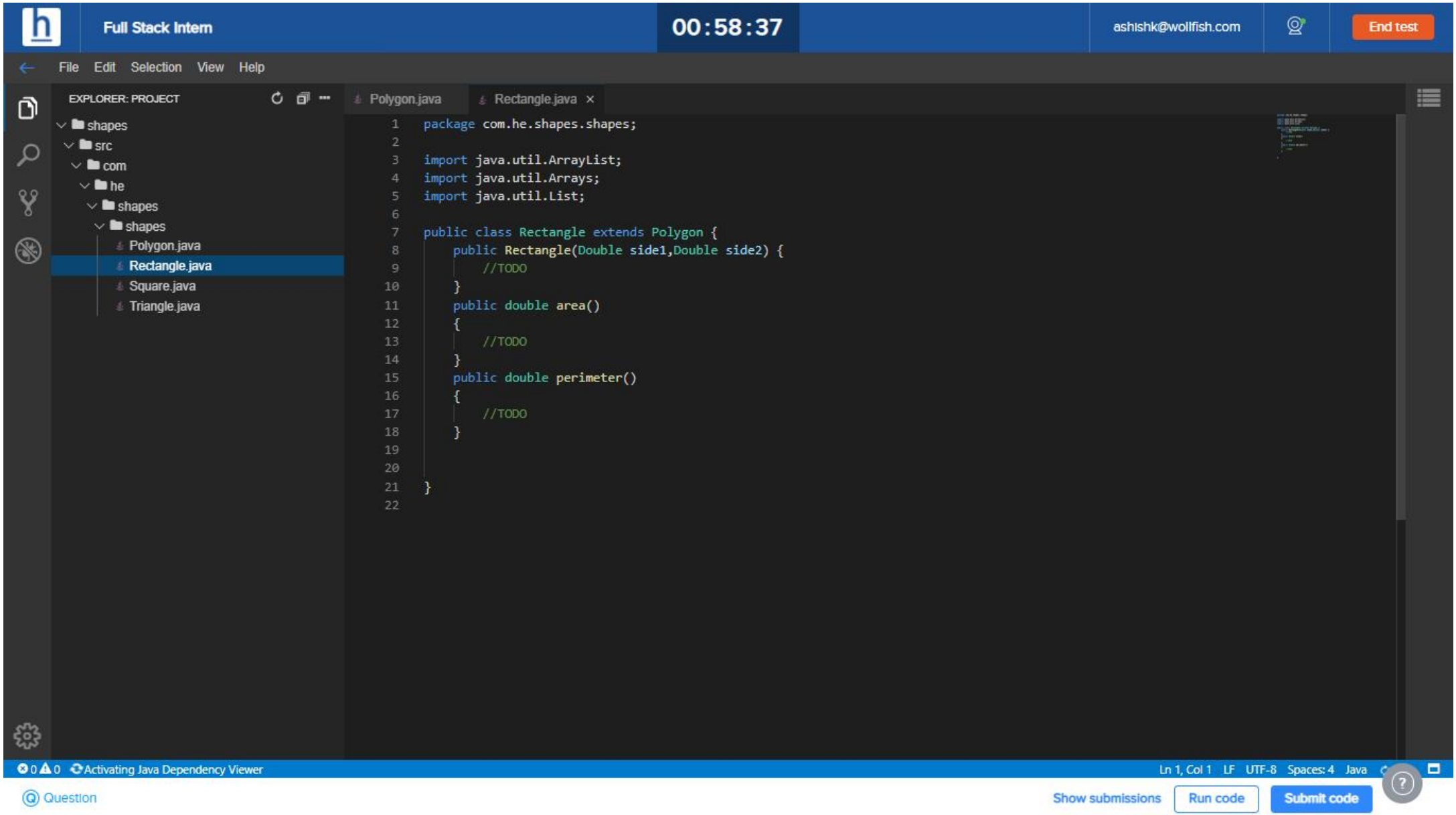
```java
package com.he.shapes.shapes;

import java.util.ArrayList;
import java.util.List;

    public abstract class Polygon {
    private List<Double> sides;

    public List<Double> getSides() {
        return sides;
    }

    public Polygon(List<Double> sides)
    {
        this.sides=sides;
    }

    public abstract double area();

    public abstract double perimeter();

}

```

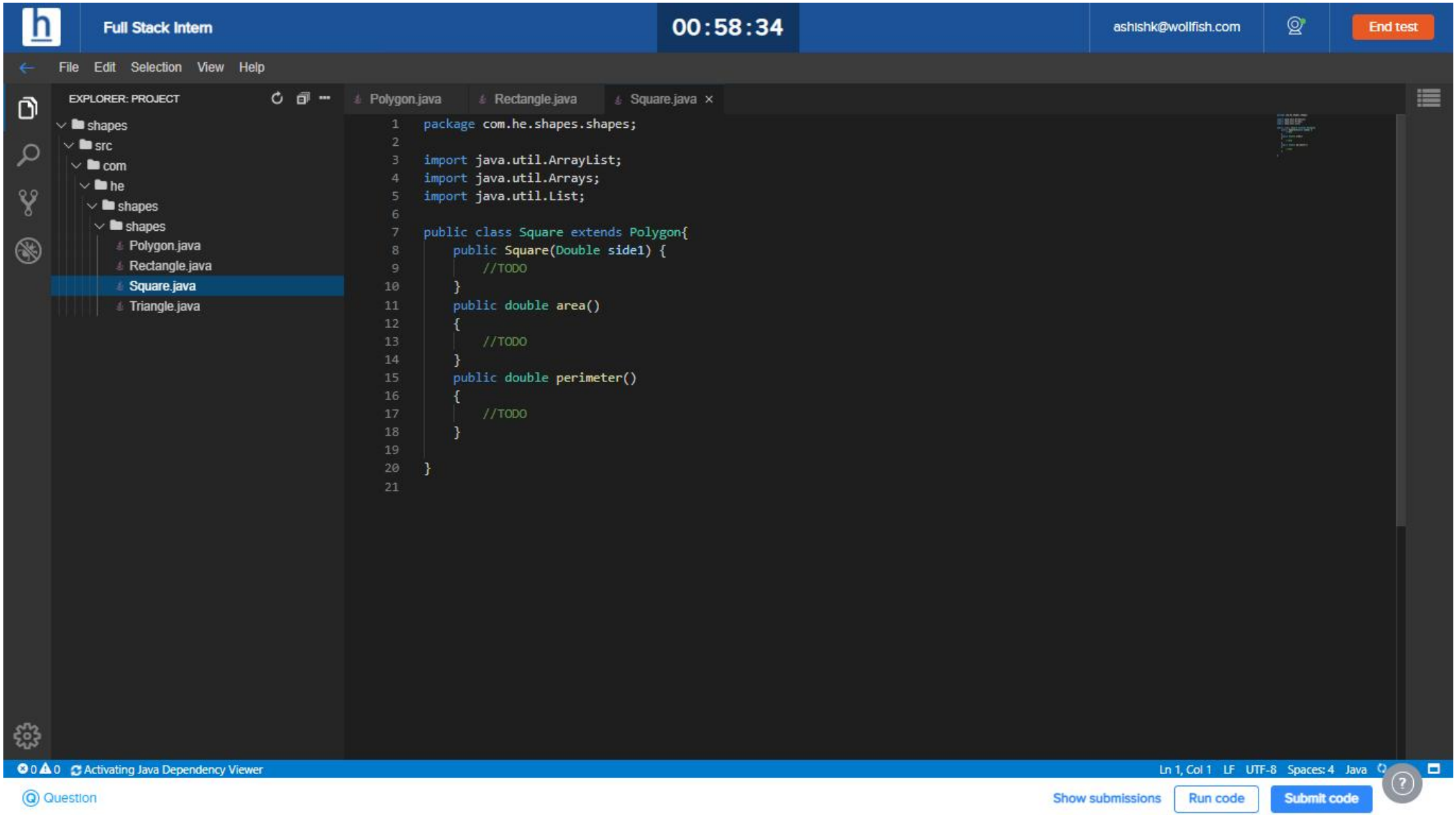
0 0 Activating Java Dependency Viewer

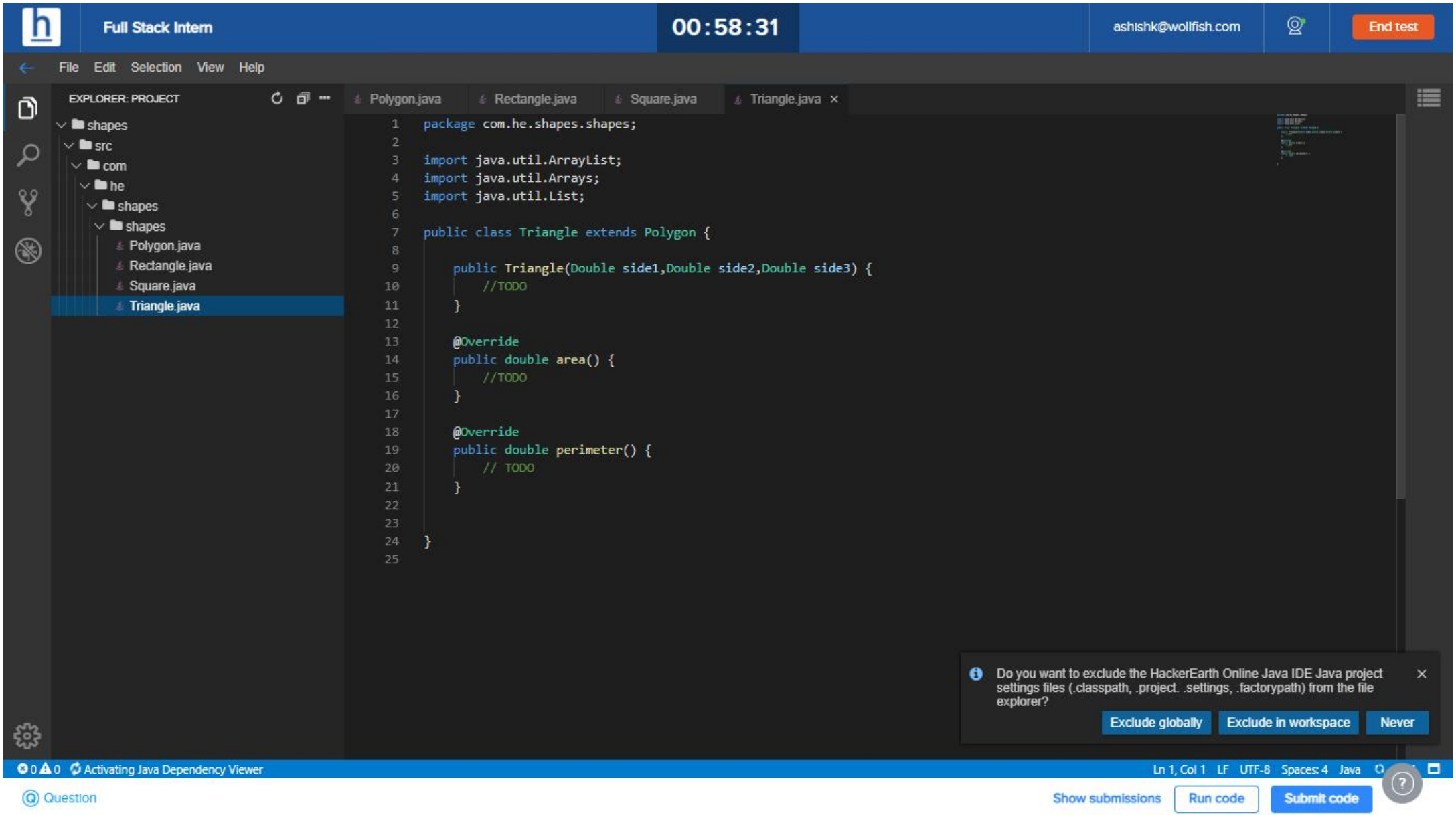
Ln 1, Col 1 LF UTF-8 Spaces: 4 Java

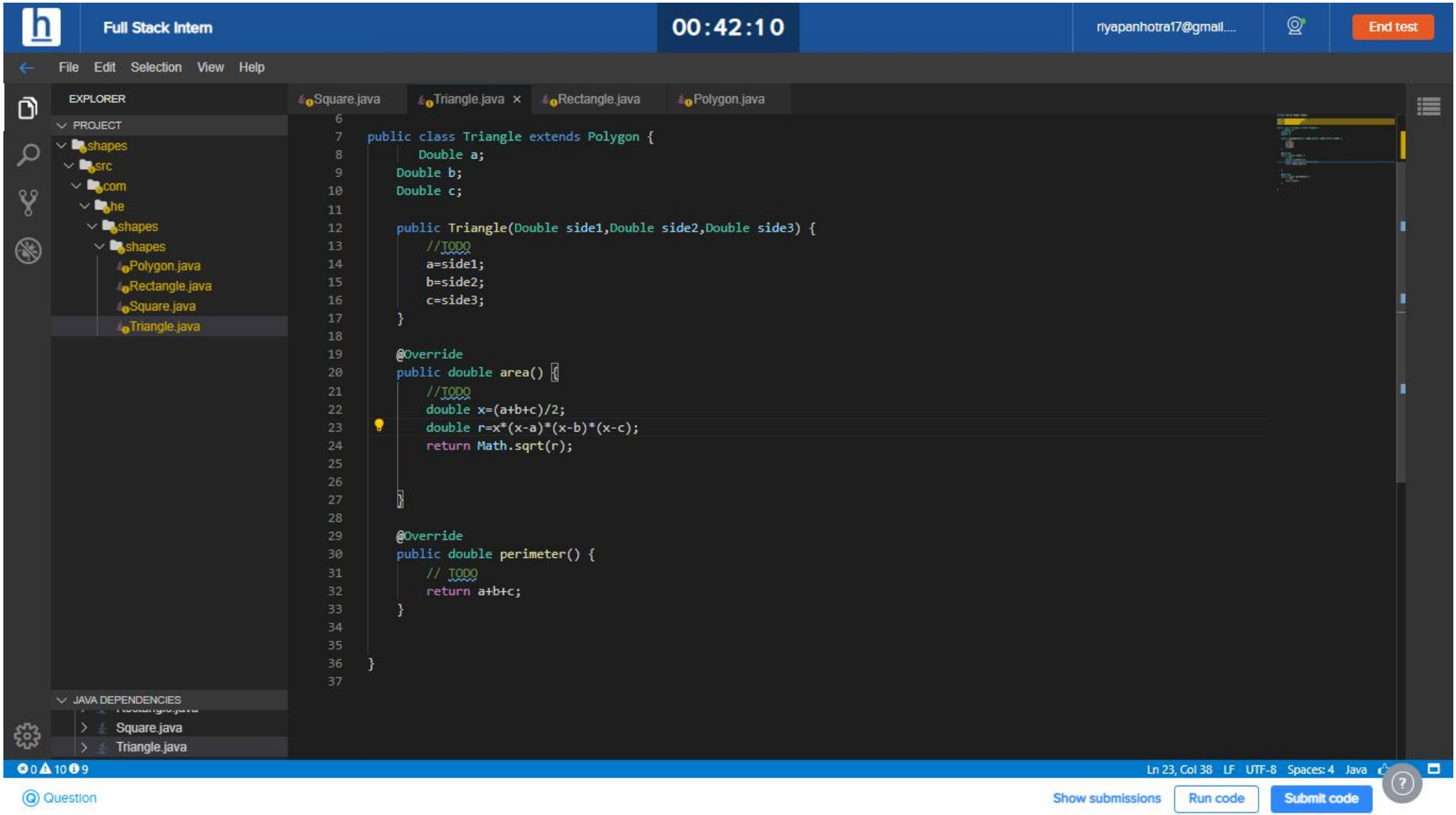
Question

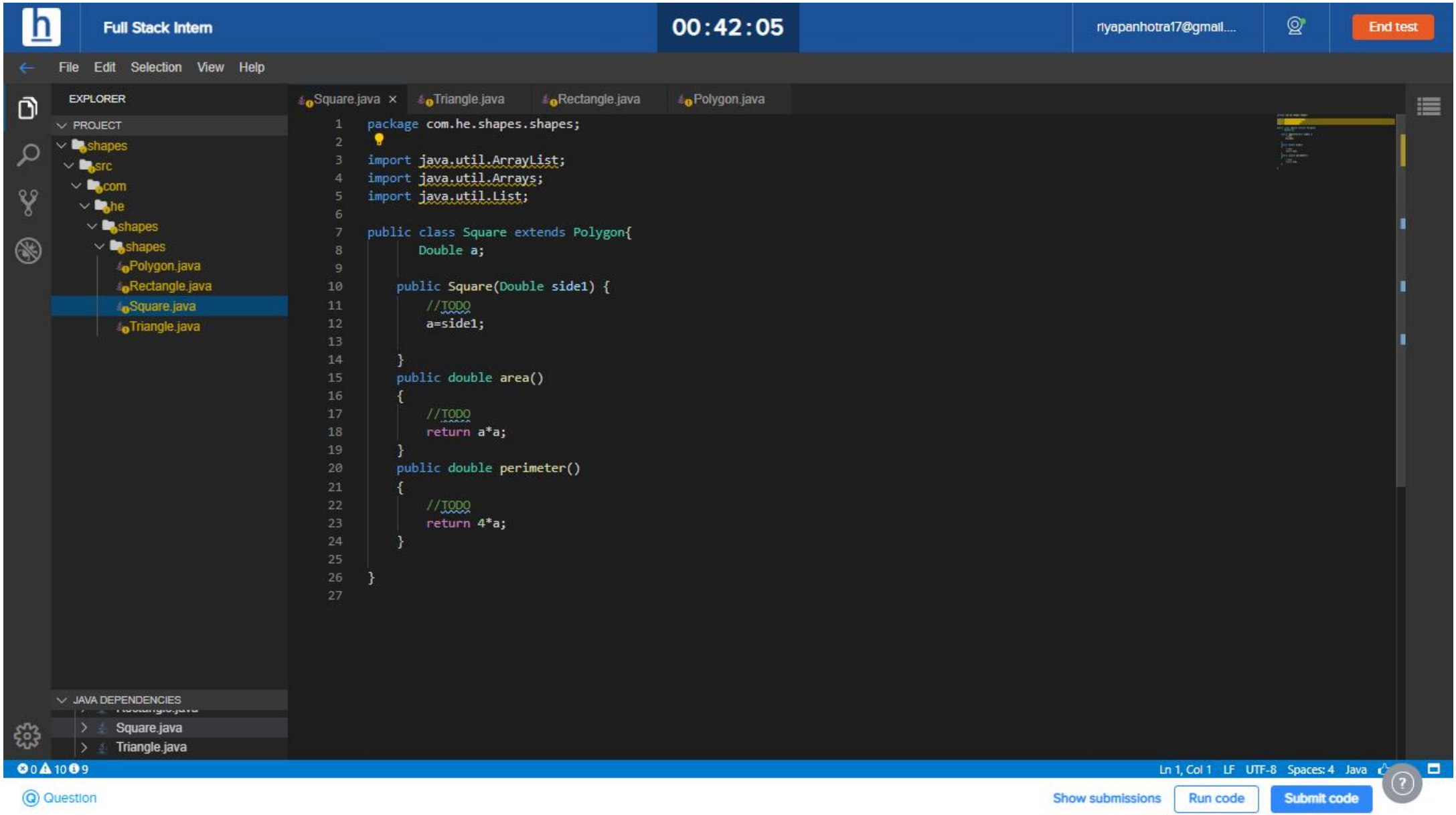
Show submissions Run code Submit code

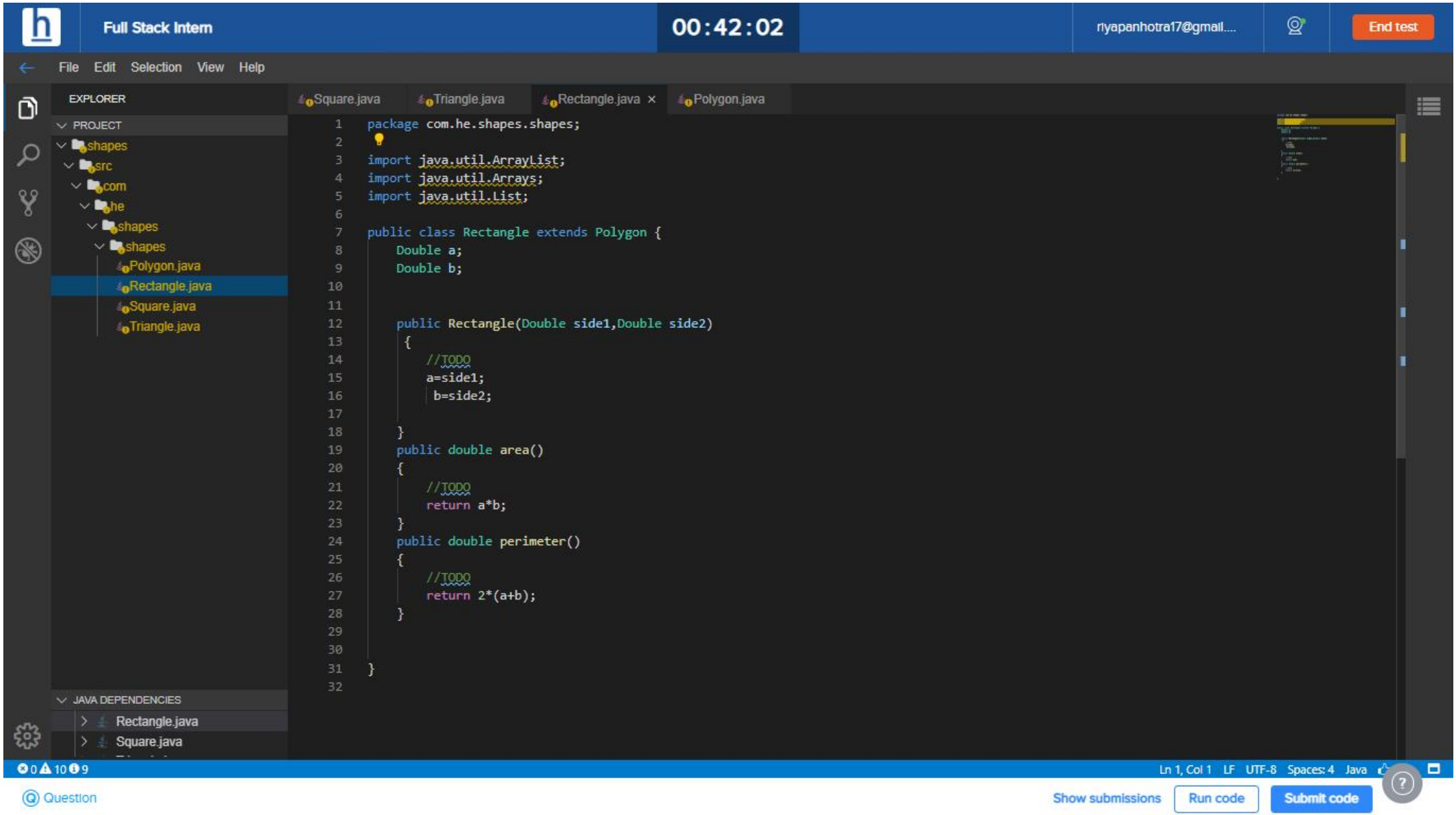

Full Stack Intern
00:58:37
ashishk@wollfish.com
End test
File Edit Selection View Help
EXPLORER: PROJECT
shapes
src
com
he
shapes
shapes
Polygon.java
Rectangle.java
Square.java
Triangle.java
Polygon.java
Rectangle.java
1 package com.he.shapes.shapes;
2
3 import java.util.ArrayList;
4 import java.util.Arrays;
5 import java.util.List;
6
7 public class Rectangle extends Polygon {
8 public Rectangle(Double side1,Double side2) {
9 //TODO
10 }
11 public double area()
12 {
13 //TODO
14 }
15 public double perimeter()
16 {
17 //TODO
18 }
19
20
21 }
22
0 0 Activating Java Dependency Viewer
Ln 1, Col 1 LF UTF-8 Spaces: 4 Java
Question
Show submissions
Run code
Submit code

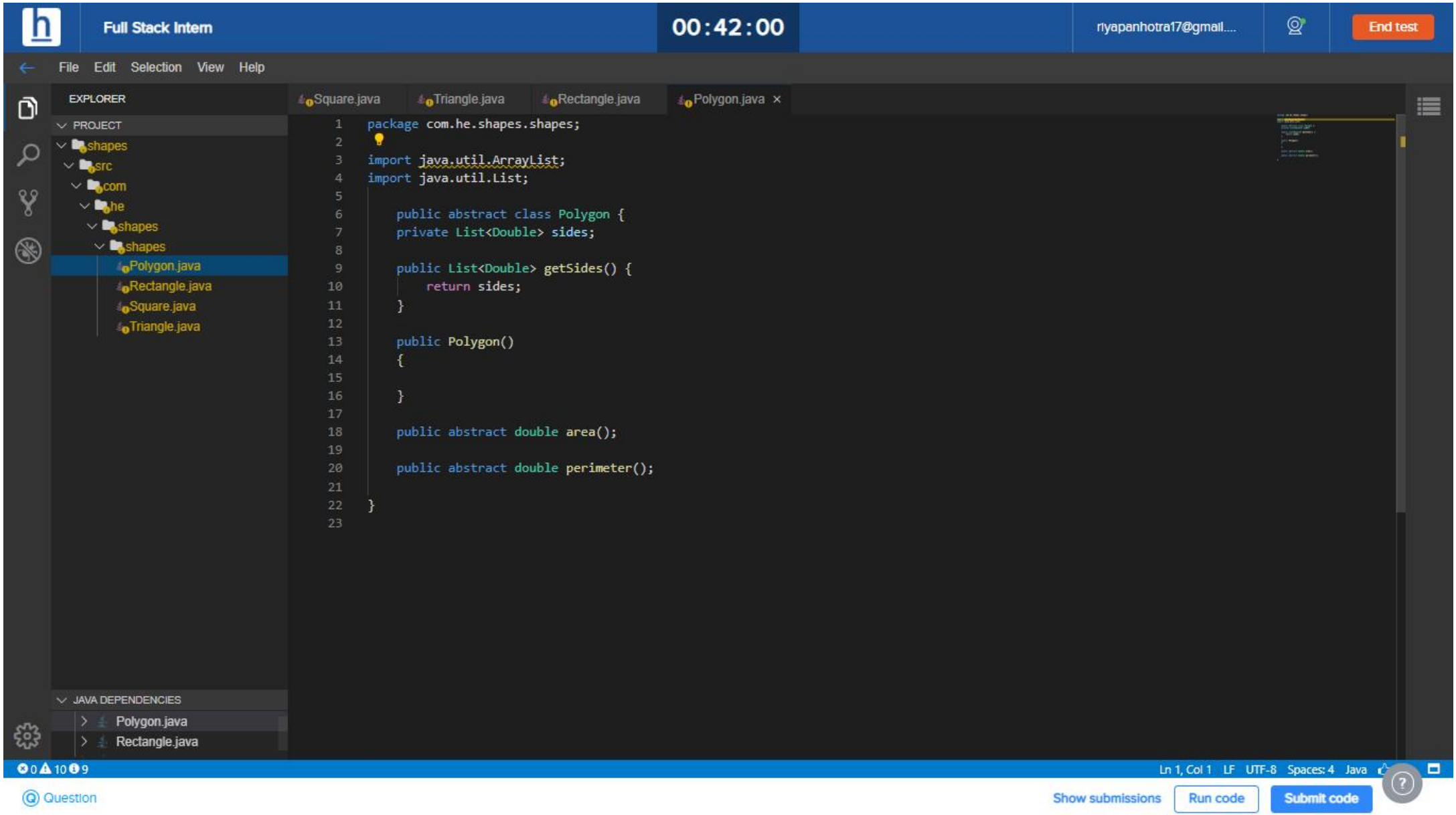

Full Stack Intern
00:58:34
ashishk@wollfish.com
End test
File Edit Selection View Help
EXPLORER: PROJECT
shapes
src
com
he
shapes
shapes
Polygon.java
Rectangle.java
Square.java
Triangle.java
Polygon.java
Rectangle.java
Square.java
1 package com.he.shapes.shapes;
2
3 import java.util.ArrayList;
4 import java.util.Arrays;
5 import java.util.List;
6
7 public class Square extends Polygon{
8 public Square(Double side1) {
9 //TODO
10 }
11 public double area()
12 {
13 //TODO
14 }
15 public double perimeter()
16 {
17 //TODO
18 }
19
20 }
21
0 0 Activating Java Dependency Viewer
Ln 1, Col 1 LF UTF-8 Spaces: 4 Java
Question
Show submissions
Run code
Submit code

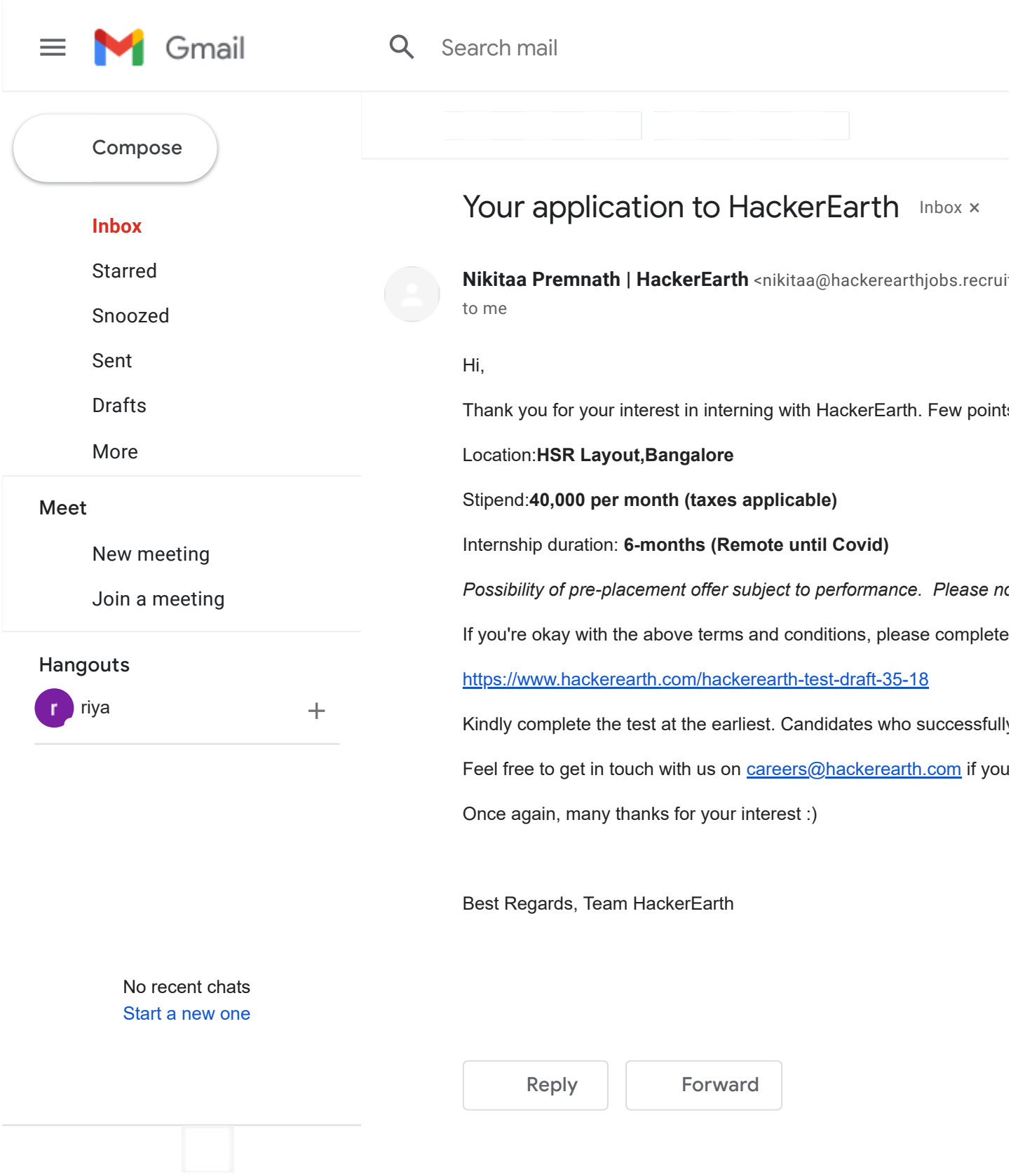

Full Stack Intern
00:58:31
ashishk@wollfish.com
End test
File Edit Selection View Help
EXPLORER: PROJECT
shapes
src
com
he
shapes
shapes
Polygon.java
Rectangle.java
Square.java
Triangle.java
Polygon.java
Rectangle.java
Square.java
Triangle.java
package com.he.shapes.shapes;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;
public class Triangle extends Polygon {
public Triangle(Double side1,Double side2,Double side3) {
//TODO
}
@Override
public double area() {
//TODO
}
@Override
public double perimeter() {
// TODO
}
}
Do you want to exclude the HackerEarth Online Java IDE Java project settings files (.classpath, .project. .settings, .factorypath) from the file explorer?
Exclude globally
Exclude in workspace
Never
0 0 Activating Java Dependency Viewer
Ln 1, Col 1 LF UTF-8 Spaces: 4 Java
Question
Show submissions
Run code
Submit code


Full Stack Intern
00:42:10
riyapanhotra17@gmail....
End test
File Edit Selection View Help
EXPLORER
PROJECT
shapes
src
com
he
shapes
shapes
Polygon.java
Rectangle.java
Square.java
Triangle.java
JAVA DEPENDENCIES
Square.java
Triangle.java
Square.java
Triangle.java
Rectangle.java
Polygon.java
public class Triangle extends Polygon {
Double a;
Double b;
Double c;
public Triangle(Double side1,Double side2,Double side3) {
//TODO
a=side1;
b=side2;
c=side3;
}
@Override
public double area() {
//TODO
double x=(a+b+c)/2;
double r=x*(x-a)*(x-b)*(x-c);
return Math.sqrt(r);
}
@Override
public double perimeter() {
// TODO
return a+b+c;
}
}
0 10 9
Ln 23, Col 38 LF UTF-8 Spaces: 4 Java
Question
Show submissions
Run code
Submit code


Full Stack Intern
00:42:05
riyapanhotra17@gmail....
End test
File Edit Selection View Help
EXPLORER
PROJECT
shapes
src
com
he
shapes
shapes
Polygon.java
Rectangle.java
Square.java
Triangle.java
JAVA DEPENDENCIES
Square.java
Triangle.java
Square.java
Triangle.java
Rectangle.java
Polygon.java
1 package com.he.shapes.shapes;
2
3 import java.util.ArrayList;
4 import java.util.Arrays;
5 import java.util.List;
6
7 public class Square extends Polygon{
8 Double a;
9
10 public Square(Double side1) {
11 //TODO
12 a=side1;
13
14 }
15 public double area()
16 {
17 //TODO
18 return a*a;
19 }
20 public double perimeter()
21 {
22 //TODO
23 return 4*a;
24 }
25
26 }
27
0 10 9
Ln 1, Col 1 LF UTF-8 Spaces: 4 Java
Question
Show submissions
Run code
Submit code


Full Stack Intern
00:42:02
riyapanhotra17@gmail....
End test
File Edit Selection View Help
EXPLORER
PROJECT
shapes
src
com
he
shapes
shapes
Polygon.java
Rectangle.java
Square.java
Triangle.java
JAVA DEPENDENCIES
Rectangle.java
Square.java
Square.java
Triangle.java
Rectangle.java
Polygon.java
package com.he.shapes.shapes;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;
public class Rectangle extends Polygon {
Double a;
Double b;
public Rectangle(Double side1,Double side2)
{
//TODO
a=side1;
b=side2;
}
public double area()
{
//TODO
return a*b;
}
public double perimeter()
{
//TODO
return 2*(a+b);
}
}
Ln 1, Col 1 LF UTF-8 Spaces: 4 Java
Question
Show submissions
Run code
Submit code


Full Stack Intern
00:42:00
riyapanhotra17@gmail....
End test
File Edit Selection View Help
EXPLORER
Square.java
Triangle.java
Rectangle.java
Polygon.java
PROJECT
shapes
src
com
he
shapes
shapes
Polygon.java
Rectangle.java
Square.java
Triangle.java
JAVA DEPENDENCIES
Polygon.java
Rectangle.java
1 package com.he.shapes.shapes;
3 import java.util.ArrayList;
4 import java.util.List;
6 public abstract class Polygon {
7 private List<Double> sides;
9 public List<Double> getSides() {
10 return sides;
11 }
13 public Polygon()
14 {
16 }
18 public abstract double area();
20 public abstract double perimeter();
22 }
Ln 1, Col 1 LF UTF-8 Spaces: 4 Java
Question
Show submissions
Run code
Submit code



Gmail

Search mail

Compose

Inbox

Starred

Snoozed

Sent

Drafts

More

Meet

New meeting

Join a meeting

Hangouts

riya

No recent chats

Start a new one

Your application to HackerEarth

Inbox ×

Nikitaa Premnath | HackerEarth <nikitaa@hackerearthjobs.recruit

to me

Hi,

Thank you for your interest in interning with HackerEarth. Few points

Location:**HSR Layout,Bangalore**

Stipend:**40,000 per month (taxes applicable)**

Internship duration: **6-months (Remote until Covid)**

Possibility of pre-placement offer subject to performance. Please no

If you're okay with the above terms and conditions, please complete

https://www.hackerearth.com/hackerearth-test-draft-35-18

Kindly complete the test at the earliest. Candidates who successfully

Feel free to get in touch with us on careers@hackerearth.com if you

Once again, many thanks for your interest :)

Best Regards, Team HackerEarth

Reply Forward

Full Stack Intern 01 : 00 : 00 sk2980039@gmail.com

Enter the te Fill your det s 3 Read Instru ons

4 Solve Questions

Use this time to read the instructions.

Test Instructions

1. Ensure that you are attempting the test using the correct email ID & kinldy take the test in JAVA as we would like to access the JAVA & SpringBoot skills
2. You must click **Submit** after you answer each question.
3. If you need assistance during the test, click the question mark (?) in the lower-right corner of the page to raise a ticket.
4. Once the test has started, the timer cannot be paused. You have to complete the test in one attempt.
5. Do not close the browser window or tab of the test interface before you submit your final answers.
6. It is recommended that you ensure that your system meets HackerEarth's compatibility requirements and check your Internet connection before starting the test.
7. It is recommended that you attempt the test in an incognito or private window so that any extensions installed do not interfere with the test environment.
8. We recommend that you close all other windows and tabs to ensure that there are no distractions.

You can start solving problems now

Start Test

Important: Proctoring Instructions

1. You can only copy and paste code within the code editor. However, nothing can be copied and pasted from any external sources including code from the code editor of another question, external websites, etc.
2. This test is being monitored remotely. Your webcam must be turned on at all times.
3. This test can be taken in full-screen mode only.

Question-specific Instructions

Programming

- Select a programming language from the list before attempting each question.

- For all programming problems, the inputs are taken from STDIN and output to STDOUT.
- Click **COMPILE & TEST** to run your solution against relevant sample test cases before submitting your answer. This is applicable for programming questions only.

```java
            str[i_str] = br.readLine();
        }
        int Q = Integer.parseInt(br.readLine().trim());
        int[][] query = new int[Q][3];
        for(int i_query = 0; i_query < Q; i_query++)
        {
            String[] arr_query = br.readLine().split(" ");
            for(int j_query = 0; j_query < arr_query.length; j_query++)
            {
                query[i_query][j_query] = Integer.parseInt(arr_query[j_query]);
            }
        }

        char[] out_ = char_at_K(N, str, Q, query);
        System.out.print(out_[0]);
        for(int i_out_ = 1; i_out_ < out_.length; i_out_++)
        {
            System.out.print("\n" + out_[i_out_]);
        }

        wr.close();
        br.close();
    }
    static char[] char_at_K(int N, String[] str, int Q, int[][] query){
       char x[]=new char[Q];
       int index=0;
          ArrayList<String> l=new ArrayList<>();
       for(int i=0;i<N;i++)
       {
           l.add(str[i]);
       }
       for(int i=0;i<Q;i++)
       {
          int L=query[i][0];
           int R=query[i][1];
            int K=query[i][2];
              List<String> alphfiltered = l.subList(L-1, R);
        alphfiltered.sort((o1, o2) -> o1.compareTo(o2));
        StringBuffer sb = new StringBuffer();
        alphfiltered.forEach(s -> sb.append(s));
        char tempArray[] = sb.toString().toCharArray();
        Arrays.sort(tempArray);
        String substring=new String(tempArray);
        char  h=substring.toCharArray()[K-1];
       x[index++]=h;

       }

    return x;
    }
}
```