

Brief Introduction

This is a simple poll system (functioning just like Poll Everywhere but is a simpler version) that contains two ends: the teacher's end (the server) and the student's end (the client).

On the teacher's end, after the program is started, a window pops out, which will request the user to provide a database address, the name of the table, the username, and the password. If everything is correct, then the answer to the poll will be uploaded to the table in the database. If the information provided by the user has something wrong, or nothing is provided, then the user can still use the program, but the answer will not be uploaded. The database can be set up later.

Then, the next thing the teacher needs to do is set the question; before that, nothing can be done. The teacher must provide the question, choices A to D, and the identifier of the question. After that, the teacher is allowed to start the server (on Port 1919), and the question and its choices will be sent to the connected student's end. Meanwhile, a thread of the message-listener is started to hear from the student's end, and the answer will be added to a thread-safe ArrayList.

When the teacher stops the server, ALL client sockets will be closed, meaning that students' answers cannot be received **(NO LATE SUBMISSION)**. Only after the server is stopped can the teacher edit the question and download the records in text format. The teacher can view the pie chart anytime they want.

On the student's end, after being started, a connection window will pop out. The student needs to fill in the IP address provided by the teacher. Then, a socket will start, and the question and its choices will be received from the teacher's end. The question and its choices will be rendered on the main window. The student [must] fill in their name and make a choice. After submitting, the teacher will receive the answer and the student's name in the following format:

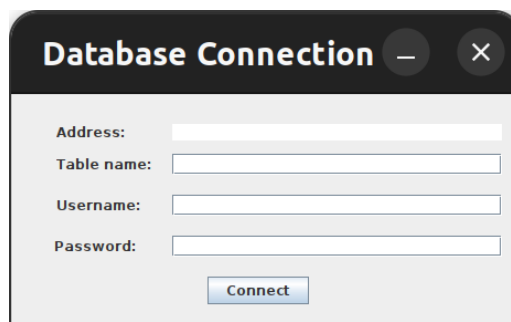
“[ans]\t[name]”

After the answer is submitted, the student will have no chance to answer the question unless they re-run the client. After the teacher closes the server, no answer will be received by the teacher **(NO LATE SUBMISSION)**.

Screenshots

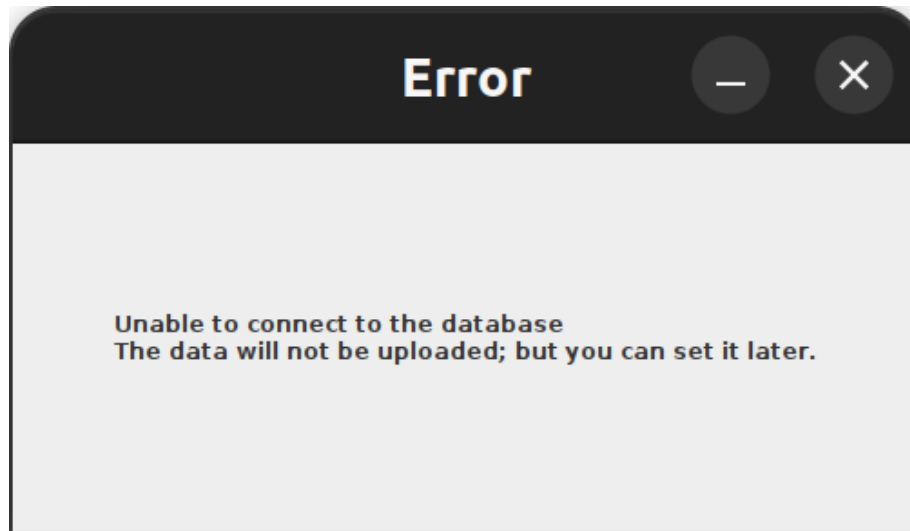
1. Teacher's End

The Database Connection window. Users can fill in these fields to connect to the database.

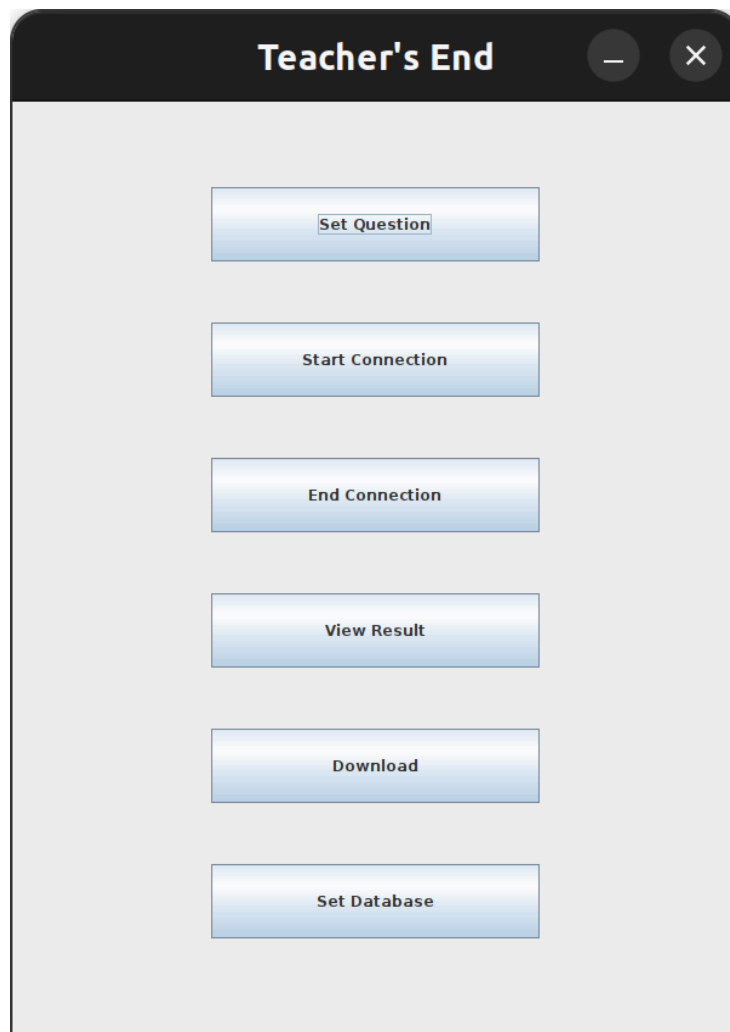


The screenshot shows a window titled "Database Connection" with a dark header bar containing a minus sign and a close button (X). The main area is light gray and contains four labeled input fields: "Address:", "Table name:", "Username:", and "Password:". Below these fields is a blue "Connect" button.

If something is wrong in connecting to the database, then there will be a window like this:



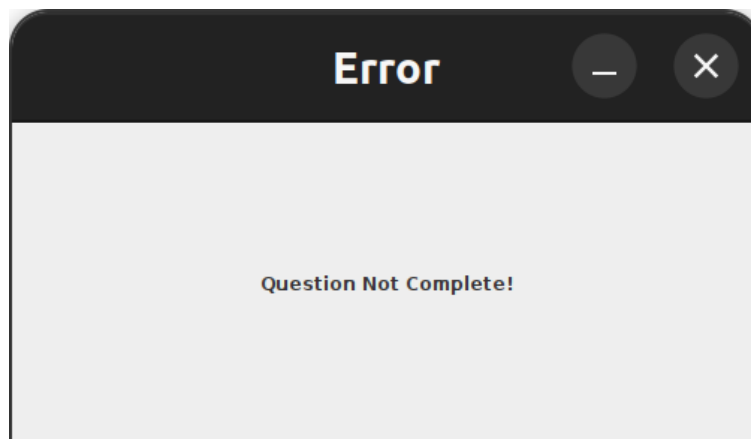
The main window of the teacher's end:



This window will pop out after clicking on the “Set Question” button:

The 'set question' dialog box features a dark header with the title 'set question' and standard window controls (minimize, maximize, close). The main area is a light gray panel with a vertical list of input fields on the right, each preceded by a label on the left. The labels are: 'Write your question here:', 'Write the Option A here:', 'Write the Option B here:', 'Write the Option C here:', 'Write the Option D here:', and 'Short name for this question:'. Below these fields is a small note: '(Make sure that you have filled all fields!)'. At the bottom right of the panel is a blue button labeled 'DONE'.

If any of the fields remain unfilled, then an error message will appear:



After setting the question and answer for the first time, the user will not be allowed to change the short name field when editing thereafter.

set question

Write your question here:

What is the original name of Java?

Write the Option A here:

C

Write the Option B here:

C++

Write the Option C here:

Oak

Write the Option D here:

X86

Short name for this question:

Java==name

(Make sure that you have filled all fields!)

DONE

After the question is set, the teacher can start the server, the window will look like this:

Teacher End--Running

Set Question

Start Connection

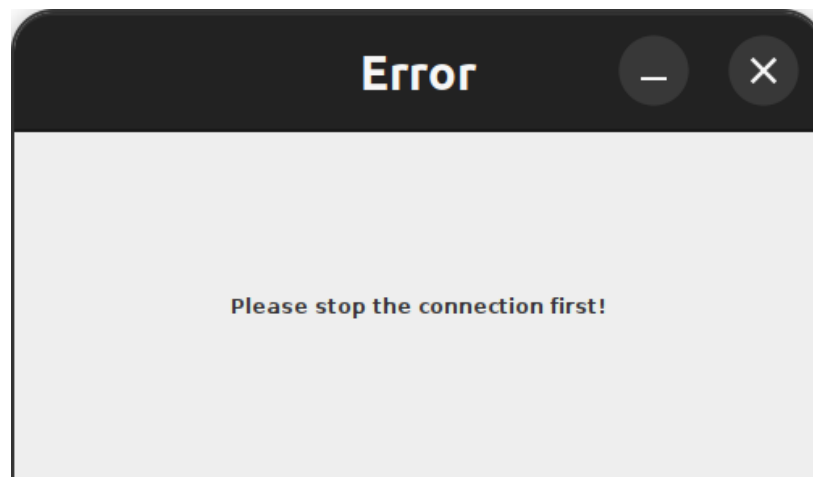
End Connection

View Result

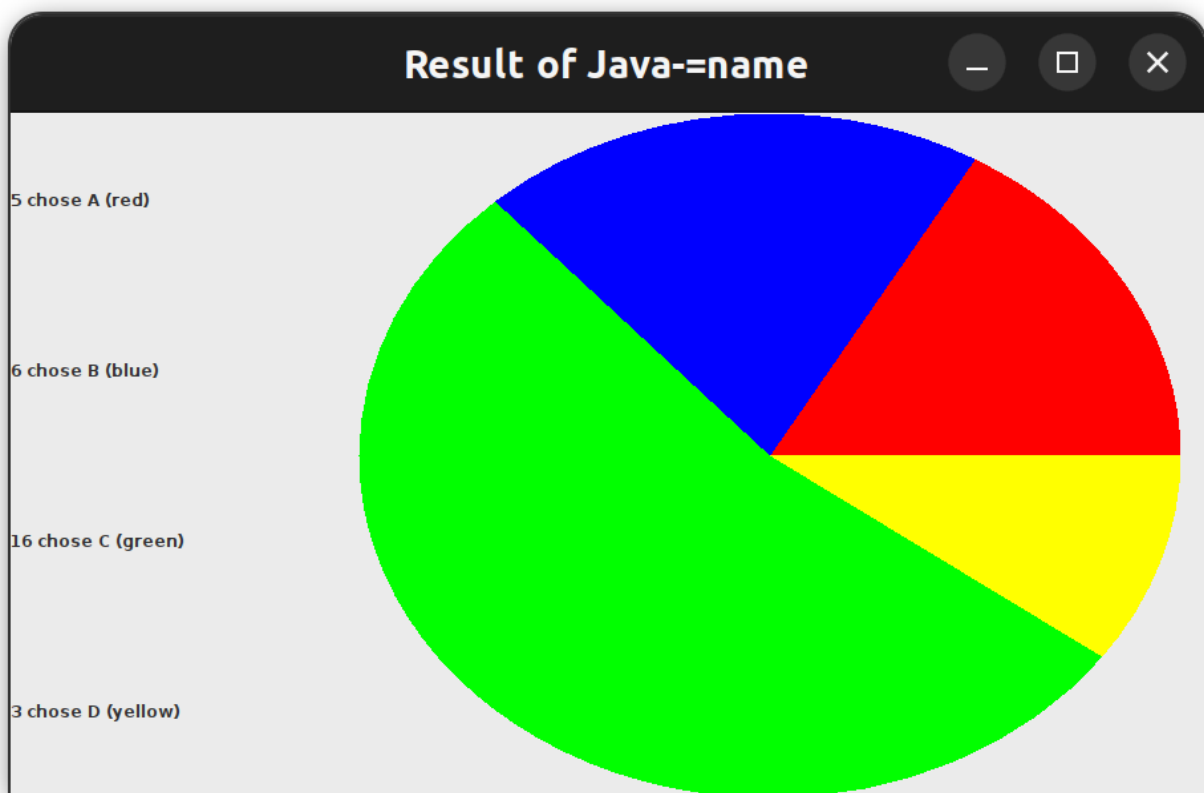
Download

Set Database

At this moment, the teacher can neither change the question nor download the .txt records.



The teacher can view the pie chart, which looks like this:



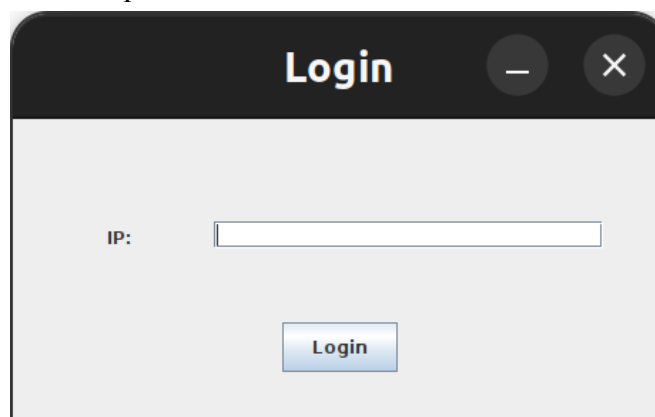
After stopping the connection, the teacher can download the .txt records with the file name in the format “[MM-DD-YYYY] [H:MIN:SEC]_[Question ID].txt” to the current working directory. The records file will look like this:

```
Student.java 05-02-2024 10:55:00_Java-==name.txt Teacher.java
1 05-02-2024 10:55:00
2 Q: What is the original name of Java?
3 A: C
4 B: C++
5 C: Oak
6 D: X86
7 There are 30 student(s) answered the question.
8 5 student(s) chose A
9 6 student(s) chose B
10 16 student(s) chose C
11 3 student(s) chose D
12 -----
13
14 Option | name
15 C Public
16 B Mike
17 A Alice
18 C Ben
19 C Charles
20 C Edward
21 D Darles
22 B Frank
23 C Georegez
24 A Harry
25 C Irk
26 D Jeff
```

Clicking on the button “set database” will let the database GUI appear again, and then the user can add or change the database settings.

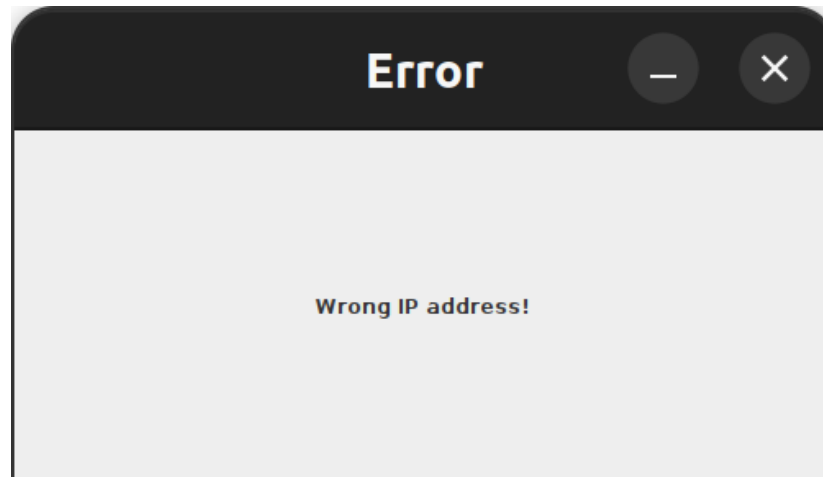
2. Students’ End

Upon the start, the student is required to fill in the IP field.



The image shows a Java Swing window titled "Login". It has a standard title bar with minimize, maximize, and close buttons. The main content area is light gray. On the left, the text "IP:" is followed by a white text input field. Below the input field is a blue button with the text "Login" in white.

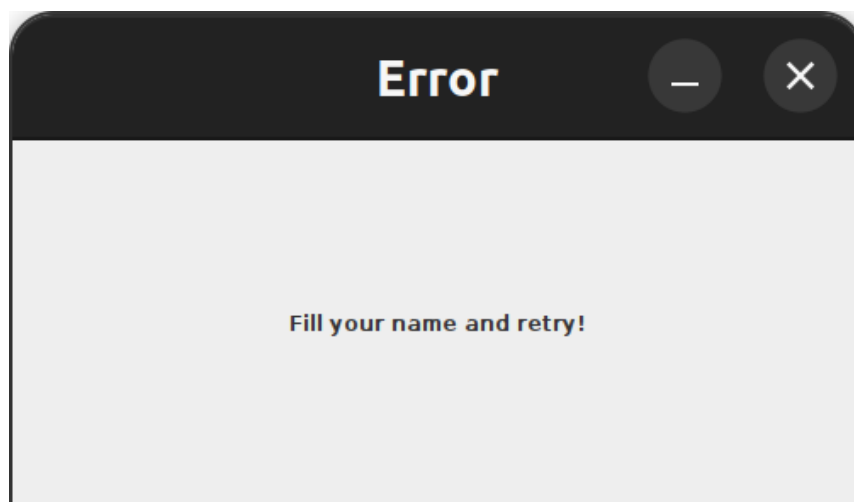
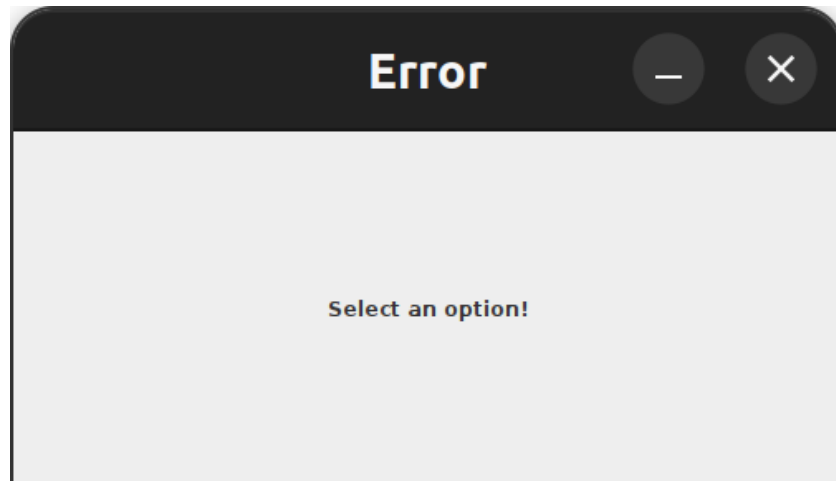
The client will try to connect to the IP for 5 seconds. If the connection spends more than 5 seconds, this situation will be regarded as a timeout (connection failed).



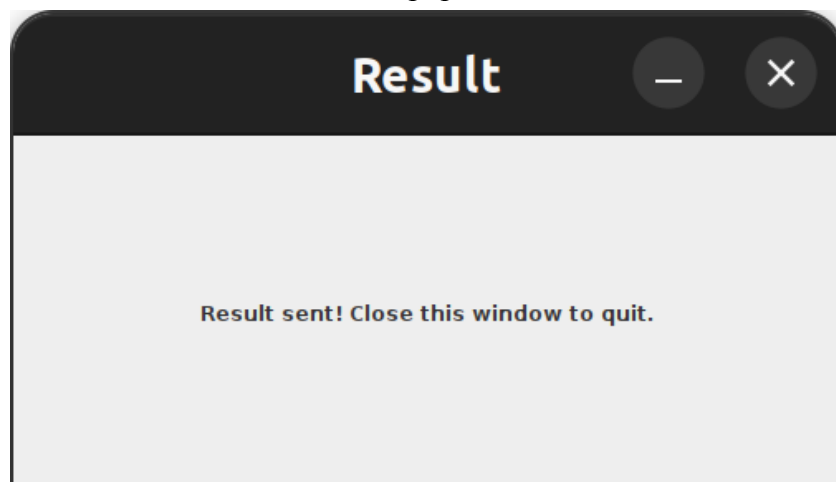
If successful, then the student's end will show the question and its choices:

A dialog box titled "Student End" with a dark gray title bar and standard minimize and close buttons. The main area is light gray. It contains a text box with the question "Q:What is the original name of Java?". Below the text box are four radio button options: "A.C", "B.C++", "C.Oak", and "D.X86". At the bottom, there is a label "Your Name:" followed by a text input field, and a blue "submit" button.

The student must select a choice AND fill in their name. Otherwise:



If everything is filled in, then this window will pop out:



This means that the results have been sent to the server. After closing this window, the student's end will quit.

3. Database

In the database, the table must contain these fields: student, question_id, and answer. Only the records AFTER the database connection will be uploaded to the database. The data stored in the database will look like:

```
mysql> select* from polls.RECORDS;
+-----+-----+-----+
| student | question_id | answer |
+-----+-----+-----+
| Amy     | Java-name   | C      |
+-----+-----+-----+
```