```java
package tellscopeV4;

/* import libraries */
import java.awt.event.ActionEvent;


/* declare class and implement action listener */
public class SubmitListener implements ActionListener {

    /* declare local attributes  */
    private static PrintWriter out;                                             //create printwriter attribute
    private static Socket s;                                                    //create socket attribute
    private static Thread resultsThread;                                        //create thread attribute
    private boolean isNotNull;                                                  //create boolean for null validation testing
    private boolean isNotChar;                                                  //create boolean for char validation testing


    public void actionPerformed(ActionEvent e) {                                //actionperformed method

        /* run methods to check if user input is null or has invalid characters */
        isNotNull = TellScopeGui04.checkInputForNull();                         //call checkInputForNull method to validate user entry
        isNotChar =  TellScopeGui04.checkInputForChars();                       //call checkInputForNull method to validate user entry

        /* use the results of validation methods to either connect to network and wait for results,
         * or display error messages based on the specific validation error
         */
        if(isNotNull == true)
        {
            if(isNotChar == false)
            {
                JOptionPane.showMessageDialog(TellScopeGui04.txtFocalInput,         //create a new message dialog box
                        "Inputs may not contain charcters or symbols!",             //print message to user
                        "Invalid Entry",                                            //set title of dialog box
                        JOptionPane.ERROR_MESSAGE);                                 //set box type to error
            }

                else
                {                                                                       //if all inputs have been entered

                String lensDiamter = TellScopeGui04.txtLensDiameter.getText();          //get user input for lens diameter and store in local attribute
                String focalRatio = TellScopeGui04.txtFocalRatio.getText();             //get user input for lens focal ratio and store in local attribute
                String eyeFocalLength = TellScopeGui04.txtEyeFocalLength.getText();     //get user input for lens eyepiece focal length and store in local attribute

                TellScopeGui04.resetInputs();                                           //reset the user input text fields

                /* create a new socket using the getSocket() method and the port number  */
                s = TellScopeGui04.getSocket(1234);                                     //create new socket


                try {                                                                   //try catch block for printwriter
                    out = new PrintWriter(s.getOutputStream(), true);
                    resultsThread = new Thread(new ClientListeningThread(s));           //create new results listening thread
                    resultsThread.start();                                              //start results listening thread
                    }
                catch (IOException f)                                                   //catch any exceptions
                    {
                        f.printStackTrace();                                            //print the exception
                    }

                    /* check if the user has selected reflecting or refracting telescope */
                        if(TellScopeGui04.reflect.isSelected())                             //if reflect is selected
                        {
                            /* get the text and write users input to socket */
                            out.println("reflect");                                         //send keyword "reflect" to server
                            out.println(focalRatio);                                        //send focal ratio to server
                            out.println(lensDiamter);                                       //send lens diameter to server
                            out.println(eyeFocalLength);                                    //send eye focal length to server
                            out.println("calculate");                                       //send keyword "calculate" to server
                        }

                            else if(TellScopeGui04.refract.isSelected())                    //else if refract is selected
                            {
                                /* get the text and write users input to socket */
                                out.println("refract");                                     //send keyword "refract" to server
                                out.println(focalRatio);                                    //send focal ratio to server
                                out.println(lensDiamter);                                   //send lens diameter to server
                                out.println(eyeFocalLength);                                //send eye focal length to server
                                out.println("calculate");                                   //send keyword "calculate to
server"

                            }
                    }

            }

            else                                                                    //if(isNotNull == false)
            {
                JOptionPane.showMessageDialog(TellScopeGui04.txtFocalRatio,         //create a new message dialog box
                        "You must enter values for all three fields",               //print message to user
                        "Invalid Entry",                                            //set title of dialog box
                        JOptionPane.ERROR_MESSAGE);                                 //set box type to error
            }


    }//end action performed method



}//end SubmitListner()

```