

Access to this work was provided by the University of Maryland, Baltimore County (UMBC) ScholarWorks@UMBC digital repository on the Maryland Shared Open Access (MD-SOAR) platform.

Please provide feedback

Please support the ScholarWorks@UMBC repository by emailing [scholarworks-group@umbc.edu](mailto:scholarworks-group@umbc.edu) and telling us what having access to this work means to you and why it's important to you. Thank you.

# *A Programmed Instruction Tutor for Java: Evidence of Rule-Governed Learning*

Henry H. Emurian

Information Systems Department  
College of Engineering and Information Technology  
UMBC

Baltimore, Maryland 21250  
emurian@umbc.edu

<http://nasa1.ifsm.umbc.edu/learnJava/tutorLinks/TutorLinks.html>

# January 1, 2003

## When Everybody Knew a Poet

To the Editor:

Re "A Lost Eloquence," by Carol Muske-Dukes (Op-Ed, Dec. 29):

The balking of students to putting verse to heart by rote memorization is not limited to poetry. There is almost a pedagogical malaise that decries rote learning in disciplines like science, mathematics and engineering. And critical analysis and scholarship are being replaced by searching the Web.

There is a growing contempt for the hard work of achieving mastery.

But the beauty of a poem, once learned, is not in the recitation of words. The poem, committed to memory, becomes a vehicle of communion for the self and the soul. Rote learning of the tools of thought has similar benefits in all fields.

HENRY H. EMURIAN

Baltimore, Dec. 29, 2002

# Overview

1. Objectives of training
2. Challenges of teaching computer programming to IS majors
3. A programmed instruction tutoring system
  - From rote memorization to “meaningful learning”
4. Evidence of effectiveness
5. Evidence of students’ acceptance and appreciation of this training approach

# Objectives: Near

# ...and Far

1. import java.applet.Applet;
2. import java.awt.Label;
3. public class MyProgram extends Applet {
4.   Label myLabel;
5.   public void init() {
6.     myLabel=new Label("This is my first program.");
7.     add(myLabel);
8.     myLabel.setVisible(true);
9.   }
10. }

# Challenges

1. IS students do **not** like to write computer programs.
2. IS students have **minimal coursework** in computer programming and programming languages.
3. IS students **need** a fundamental mastery programming.
4. IS students are often **demoralized** by taking courses with computer science majors that are taught by computer science faculty.
5. How can we **help** IS students to achieve the objectives?

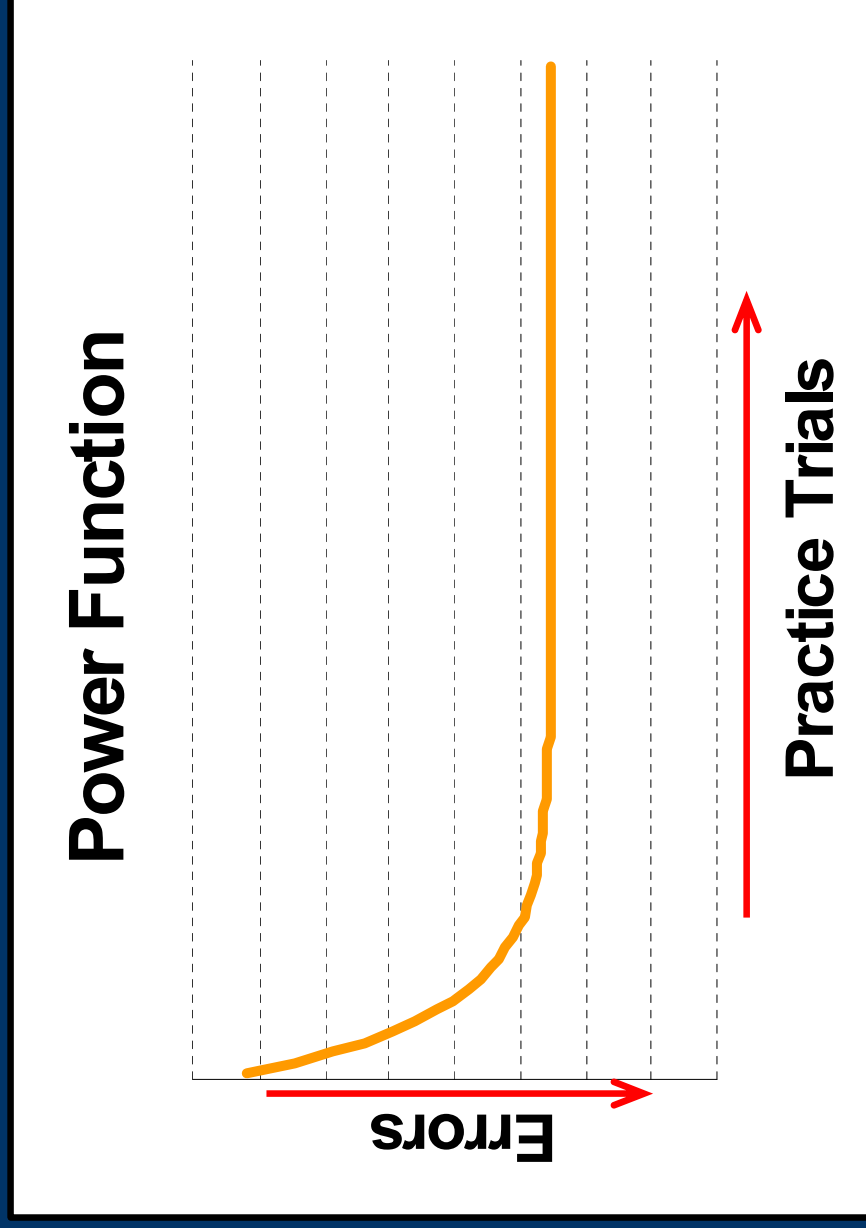
# *Underlying Assistive Principles*

1. Rote Memorization is Good
  - Fundamental to meaningful learning
    - Near and far transfer
  - Constructivism comes later (much, much later...).
2. Disciplined Study Behavior is Good
  - It is essential to mastery.
  - Most students don't know how to study. They don't know how to monitor their acquisition of competence.
3. Repetition and Overlearning are Good
  - Contribute to retention
4. Feeling Good About Yourself is Good
  - Sustains hard work and encourages future learning.

# Programmed Instruction

1. A set of **structured interactions** between a learner and a tutor.
2. Occasions disciplined study behavior that is focused on the **individual learner**.
3. Manages the **moment-by-moment interactions** between a learner and a tutor.
4. **Step-wise progression** from elementary knowledge units (*learn units*) or facts to the achievement of a complex repertoire (*meaningful learning*) that is the outcome of completing this instructional system.

# Assumption



# Marketing Challenges

*An entrenched and cultural elitism exists within academe that views competency-based education with suspicion. If all students achieve an “A,” something is considered to be wrong. My view is that my job is to overcome individual differences, not just to document them.*

*What I have to deal with from computer scientists, professional Java instructors, cognitive psychologists, ...and just about everybody else, to include my colleagues:*

# *Marketing Challenges*

***Rage***

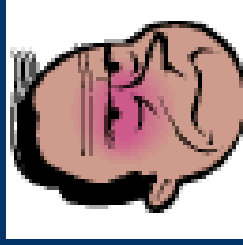


# *Marketing Challenges*

*Rage*



*Contempt*

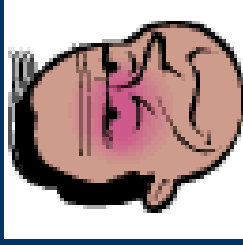


# *Marketing Challenges*

*Rage*



*Contempt*



*Ridicule*

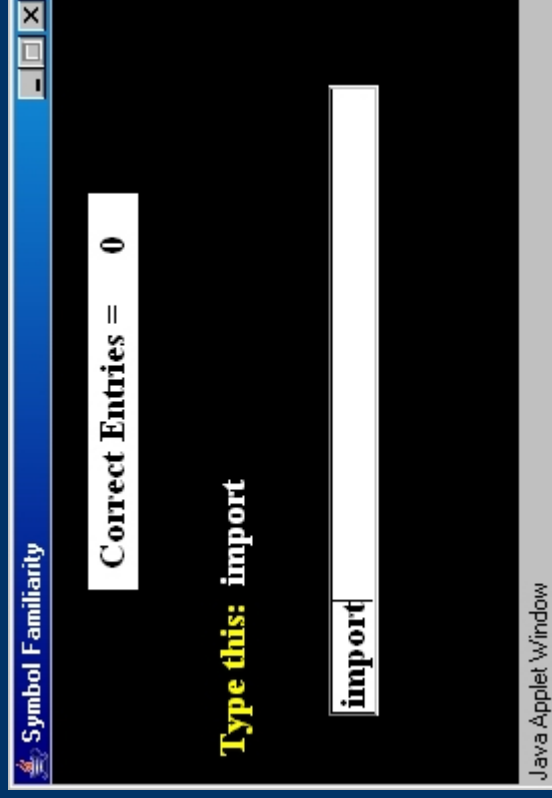


# *My Response?*

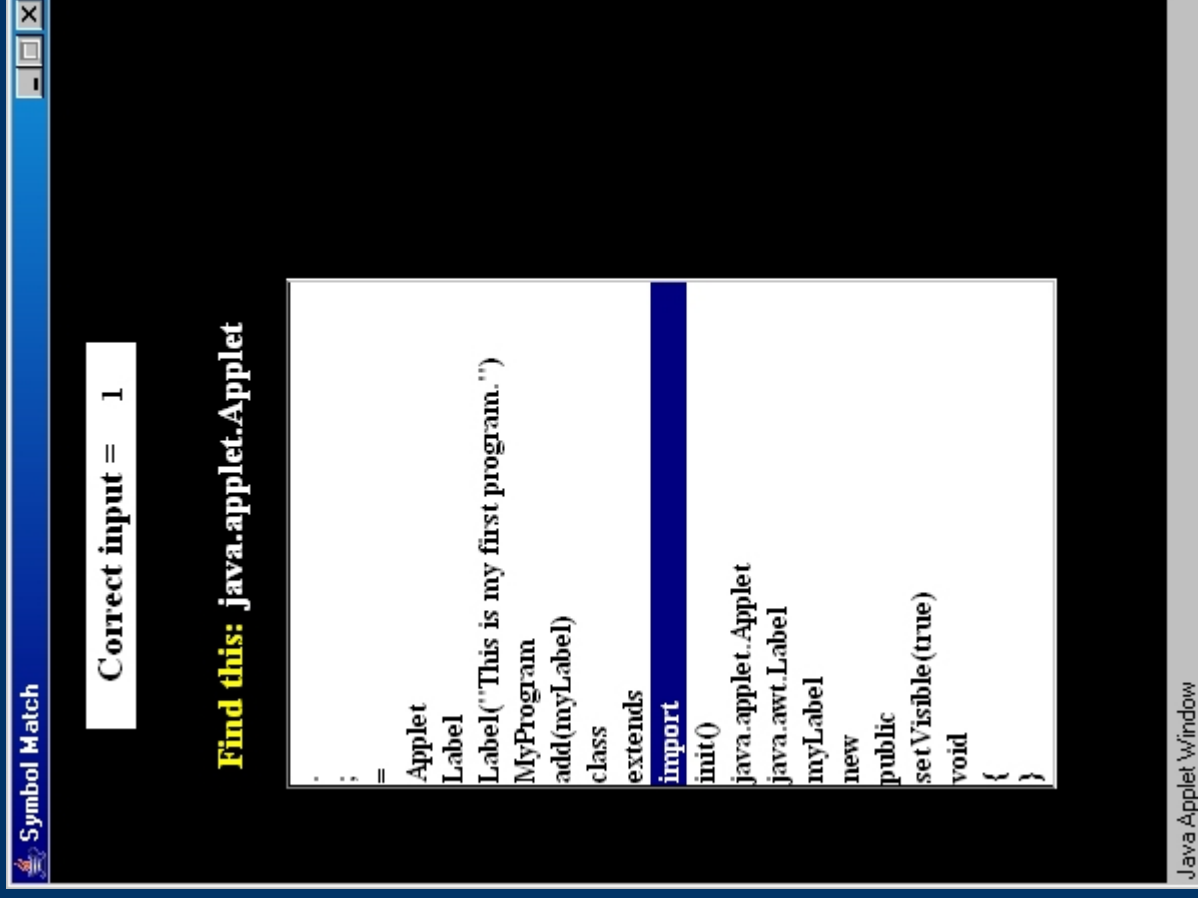
*I don't care.*



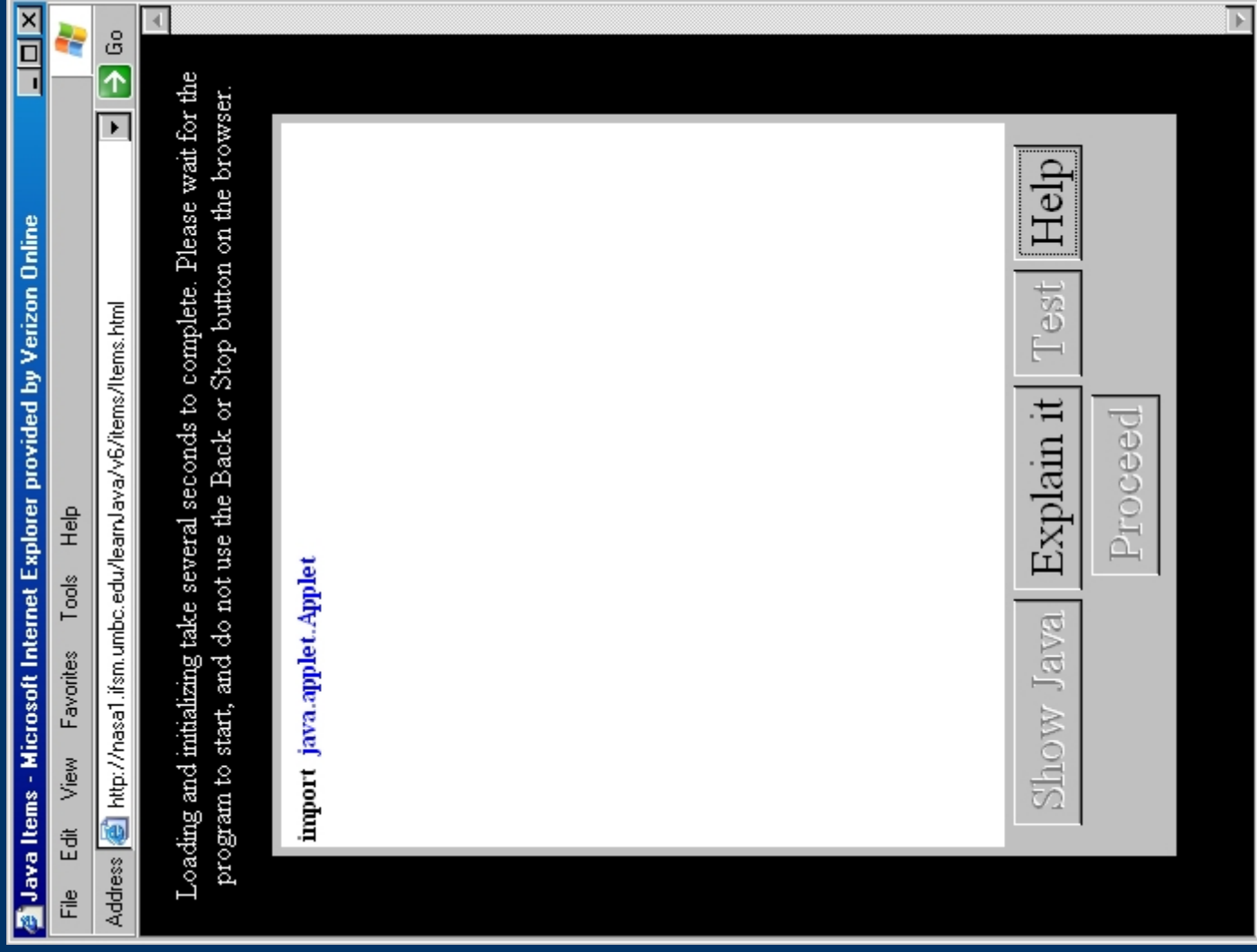
# Symbol Familiarity



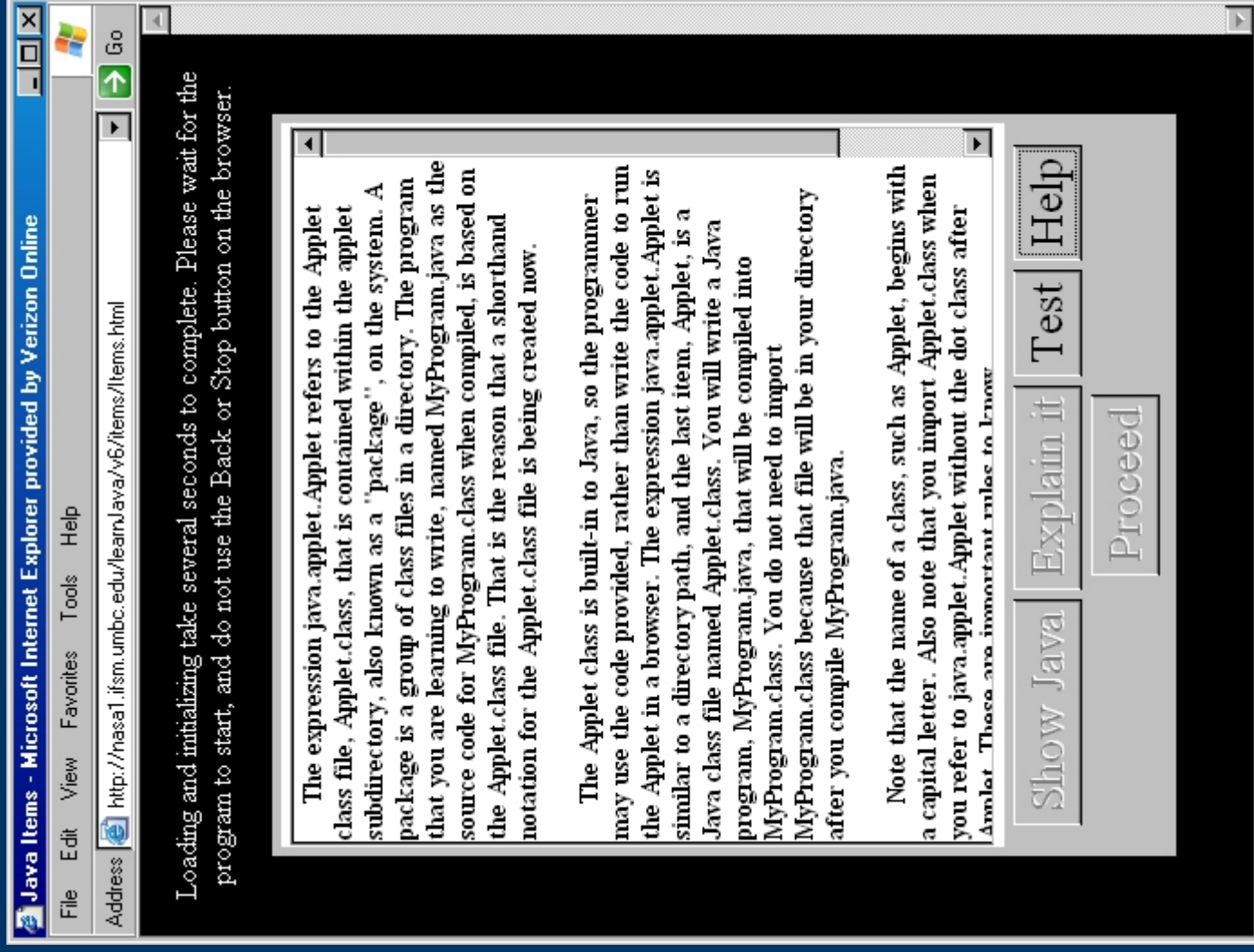
# Symbol Identification



# Show Item



# Explain Item



# *Example of a Rule*

Inside the Applet class, the `init()` method has no statements in it. When the programmer uses the `init()` method in a program and adds statements to it, that is called overriding the `init()` method. That is an important rule to know.

The general form of the `init()` method is as follows:

```
public void init() {  
    a line of Java code;  
    a line of Java code;  
}
```

# Test Item

Java Items - Microsoft Internet Explorer provided by Verizon Online

File Edit View Favorites Tools Help

Address <http://nasal1.ism.umbc.edu/learn/java/v6/items/items.html> Go

Loading and initializing take several seconds to complete. Please wait for the program to start, and do not use the Back or Stop button on the browser.

**What is the purpose of java.applet.Applet?**

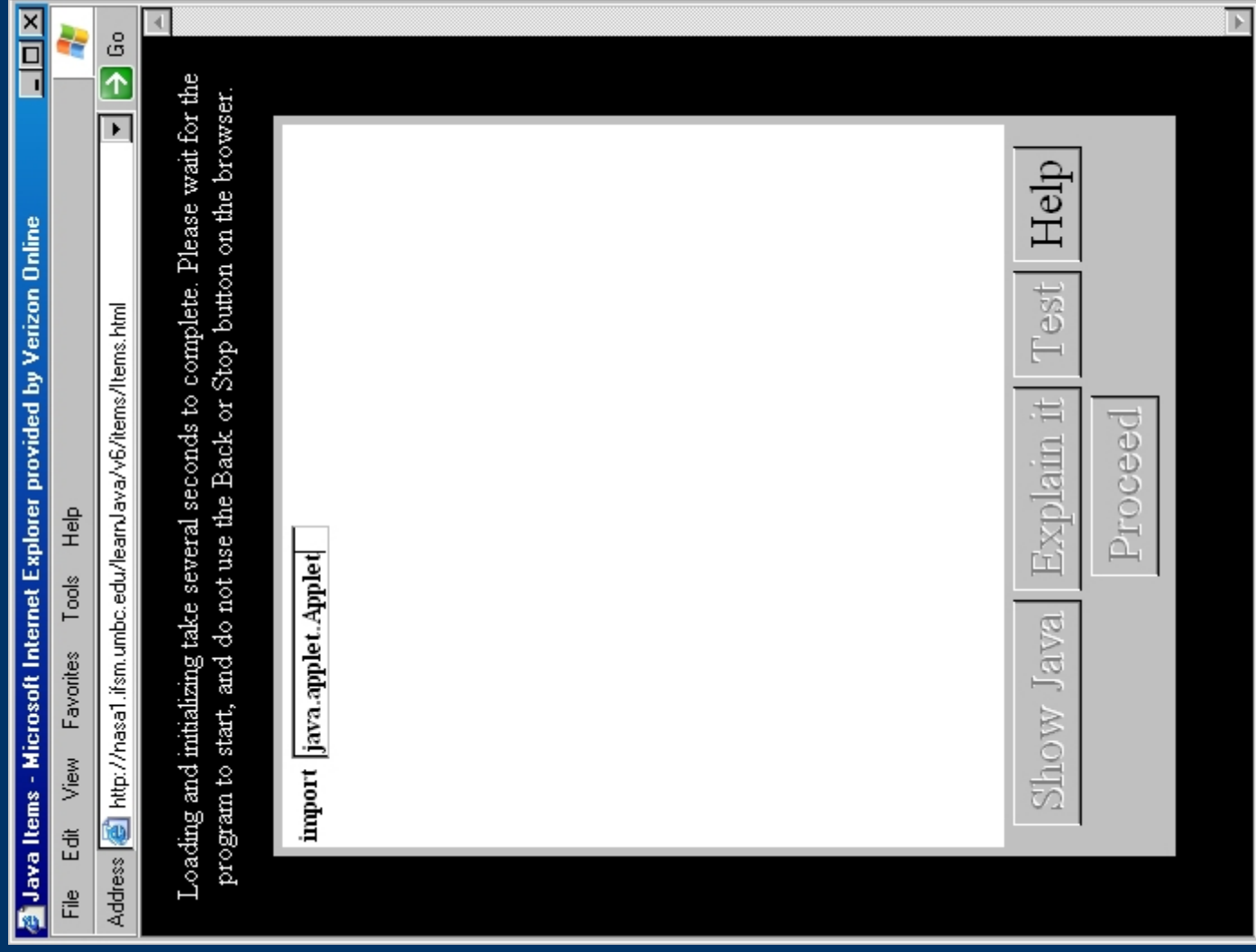
**Select the best answer.**

|                                |                                                                                                          |
|--------------------------------|----------------------------------------------------------------------------------------------------------|
| <input type="radio"/> Choice 1 | This is a reference to the appletclass file on the system.                                               |
| <input type="radio"/> Choice 2 | Compile the java program named applet.Applet.                                                            |
| <input type="radio"/> Choice 3 | This is a reference to the Applet class file, Applet.class, located in the applet package on the system. |

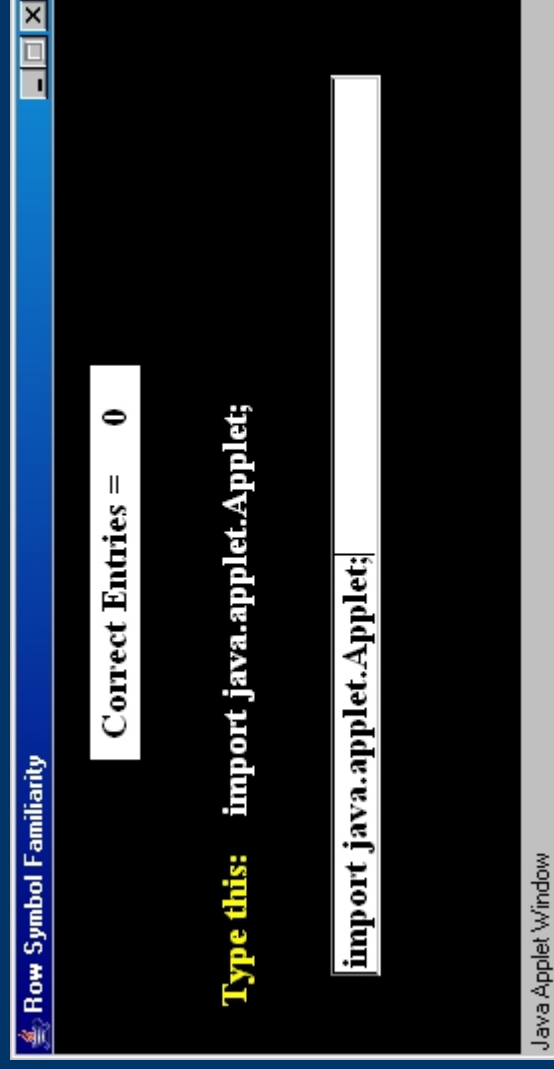
Show Java Explain it Test Help

Proceed

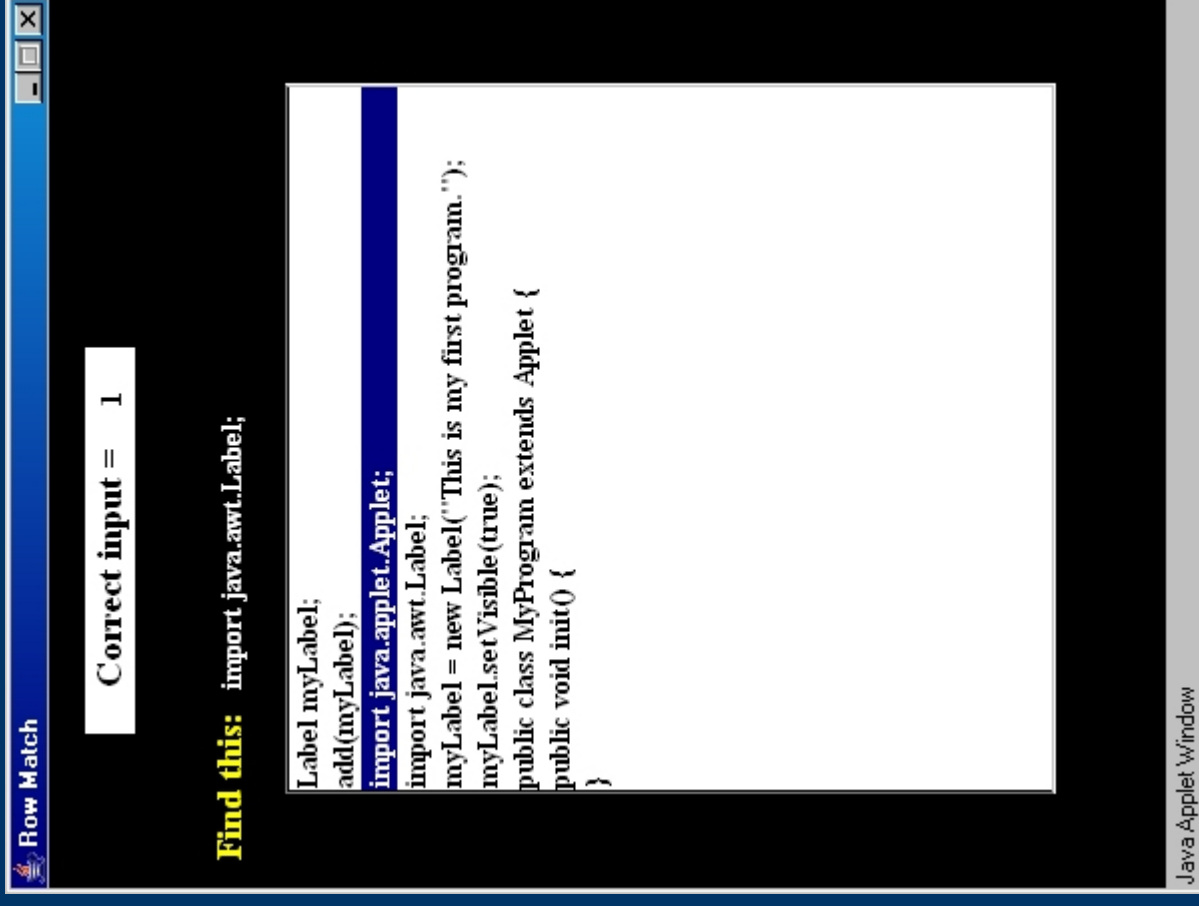
# Enter Item



# Row Familiarity



# Row Identification



# Row Learning (Two passes)

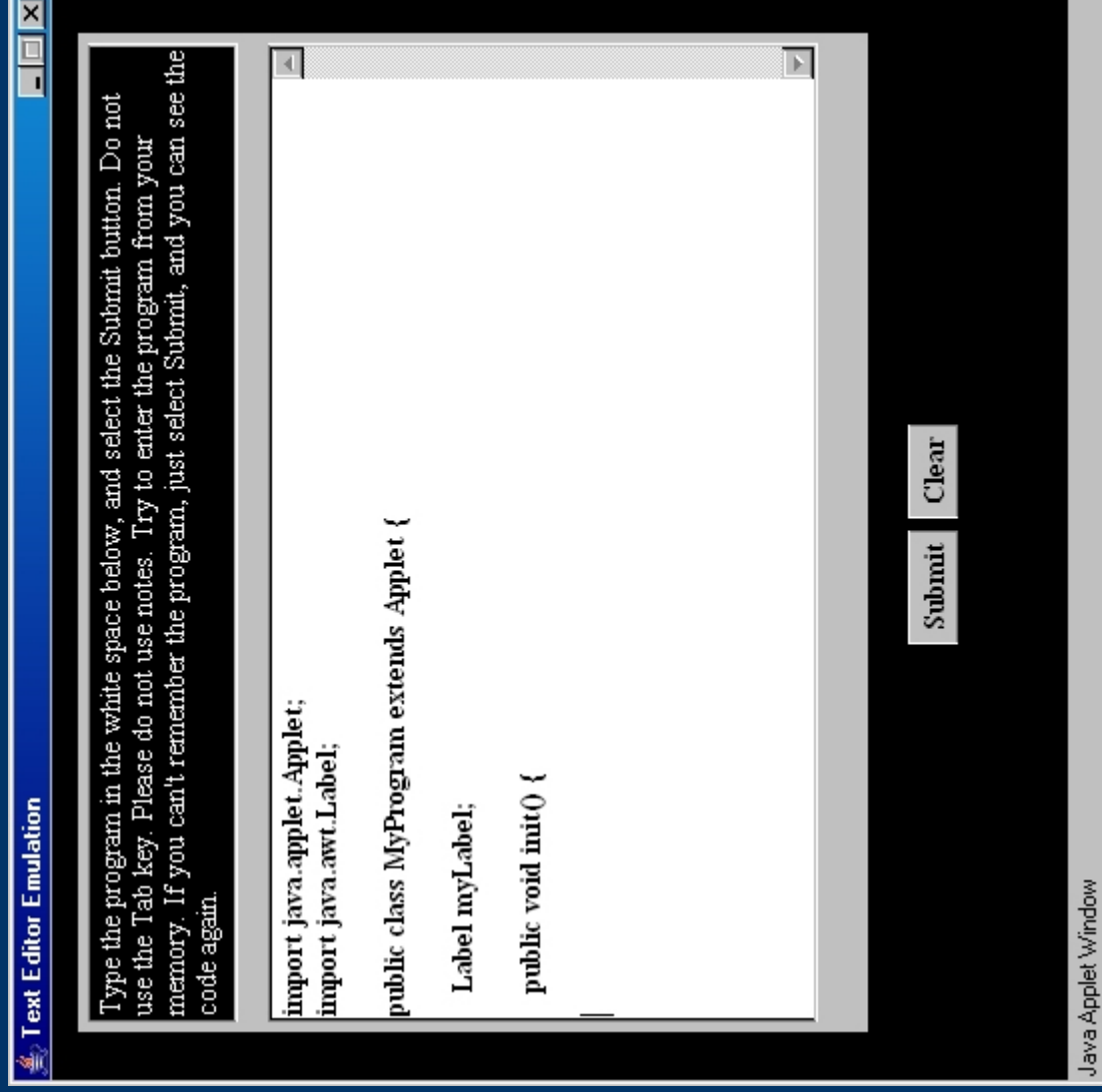
Java Tutoring System: Pass 1 of 2

|        |                                         |
|--------|-----------------------------------------|
| Row 1  | <code>import java.applet.Applet;</code> |
| Row 2  | <code>import java.awt.Label;</code>     |
| Row 3  |                                         |
| Row 4  |                                         |
| Row 5  |                                         |
| Row 6  |                                         |
| Row 7  |                                         |
| Row 8  |                                         |
| Row 9  |                                         |
| Row 10 |                                         |

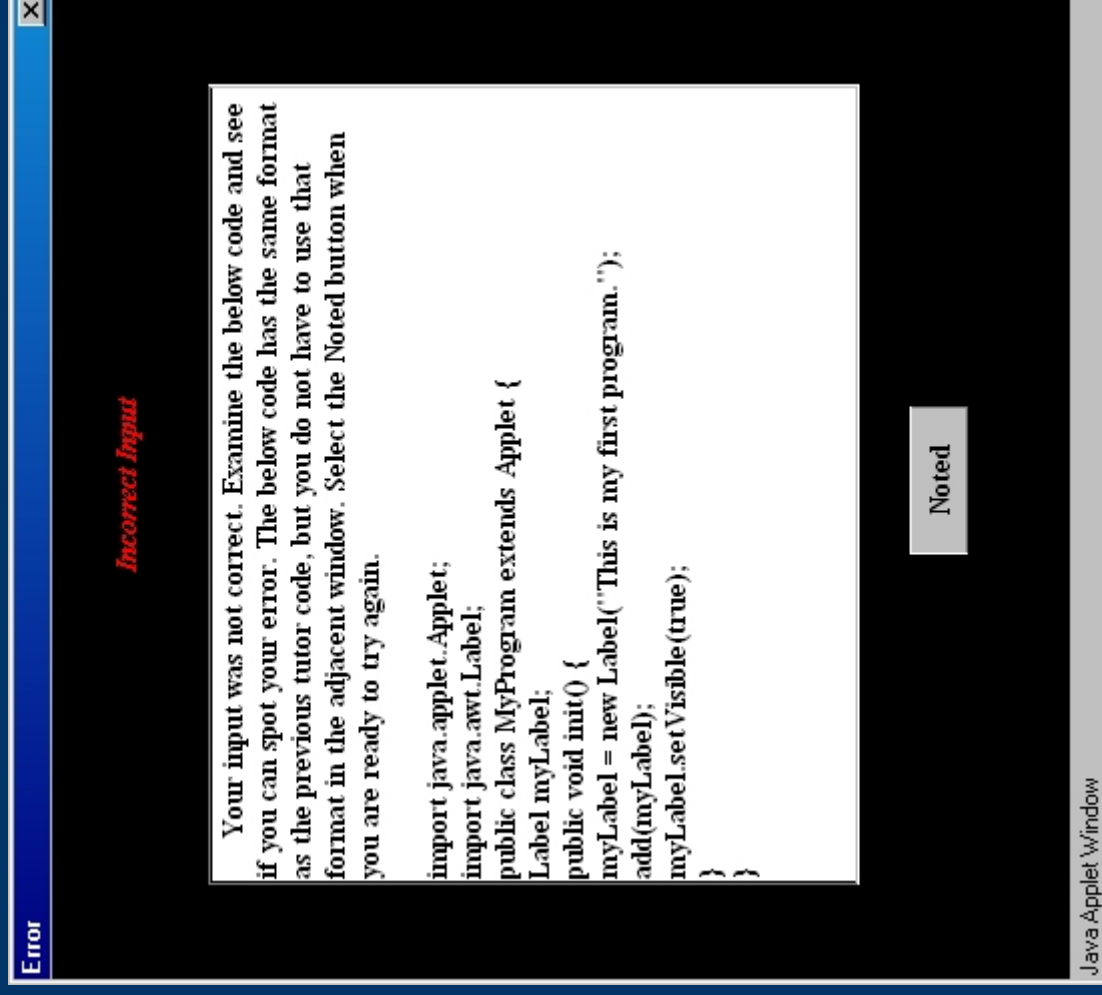
Explain the code?

Java Applet Window

# Text Window



# Text Error

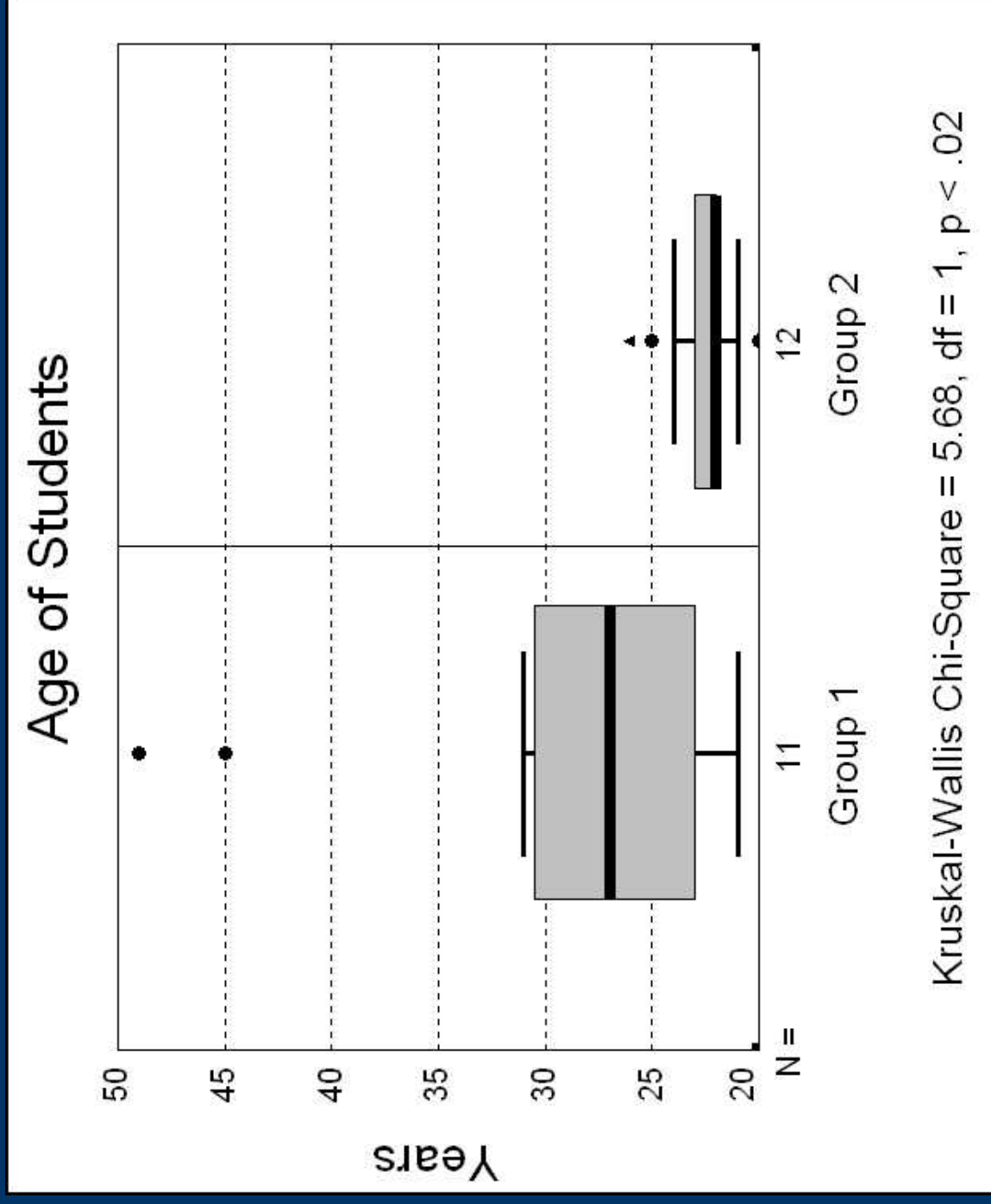


# Subjects

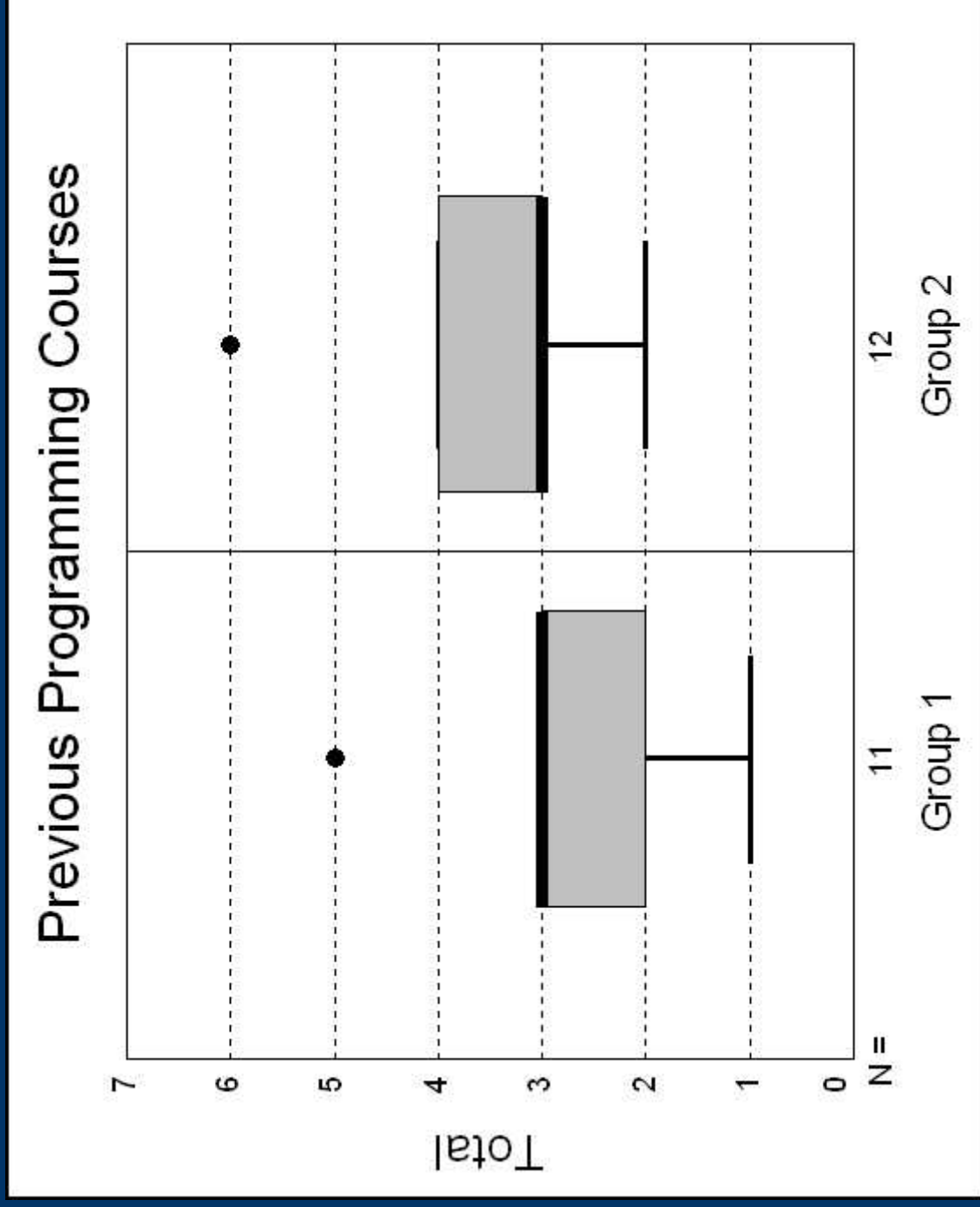
- **Group 1**
  - **Summer 2003**
    - Graduate Course
    - $N = 12$
    - 8 females, 4 males

- **Group 2**
  - **Spring 2004**
    - Undergraduate Course
    - $N = 12$
    - 2 females, 10 males

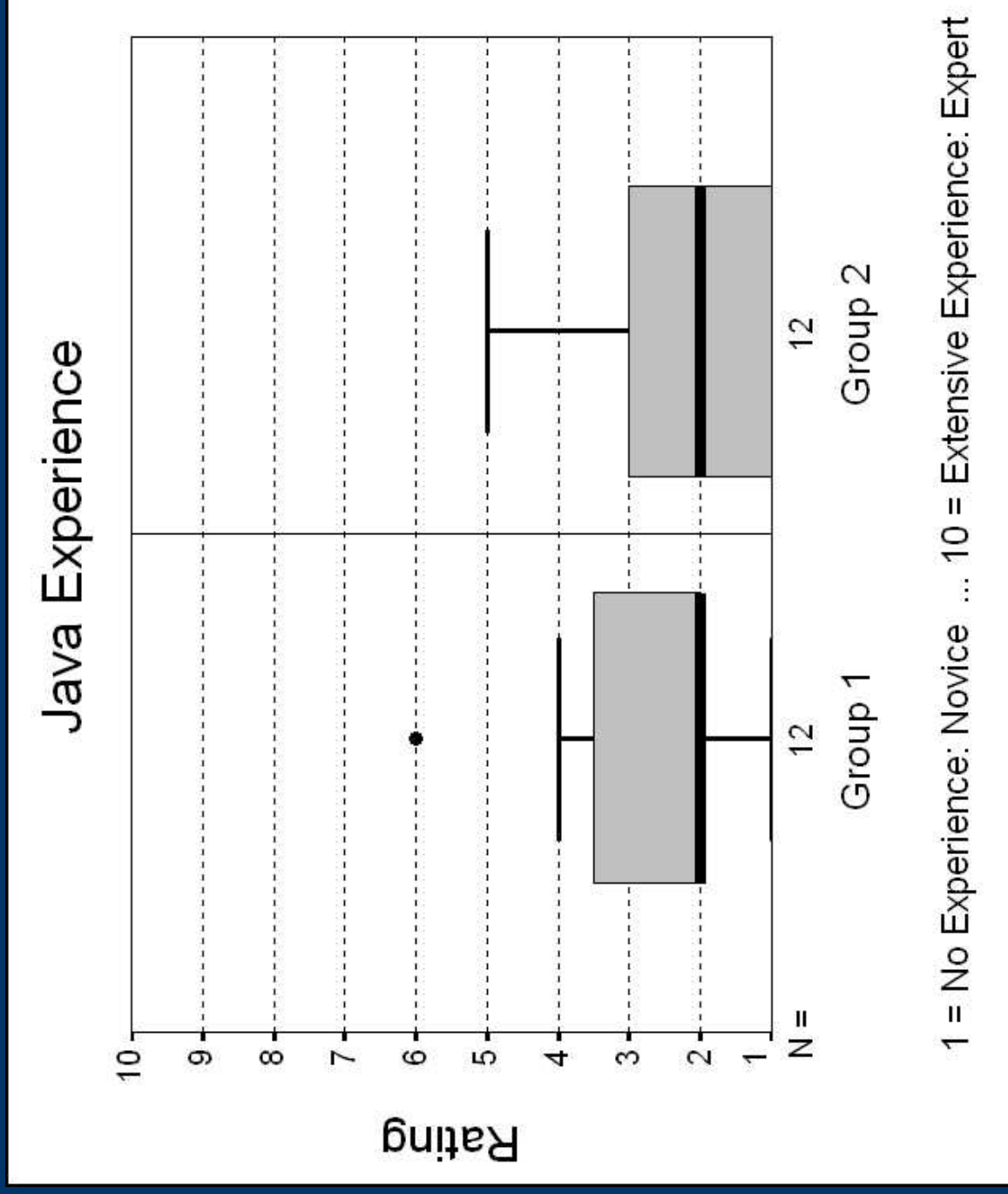
# Background: Age



# Background: Courses



# Background: Java Experience



# Procedure

- **First Session: 3 Hours**
  - **Pre-Tutor** Questionnaires
    - Background
    - SSE
    - Rules test: 5 or 10 multiple-choice questions
    - Rules efficacy
  - Run the Tutor
  - **Post-Tutor** Questionnaires
    - SSE
    - Rules test
    - Rules efficacy

- **Second Session: 3 Hours**
  - Lecture
  - Run the Applet
  - **Post-Lecture** Questionnaires
    - SSE
    - Rules test
    - Rules efficacy

# Software Self-Efficacy Assessment

## 21 Items

How confident are you that you can use the following symbol now to write a Java applet?

import

Not at all confident   1   2   3   4   5   6   7   8   9   10   Totally confident

# Rules Assessment

## 5 or 10 Items

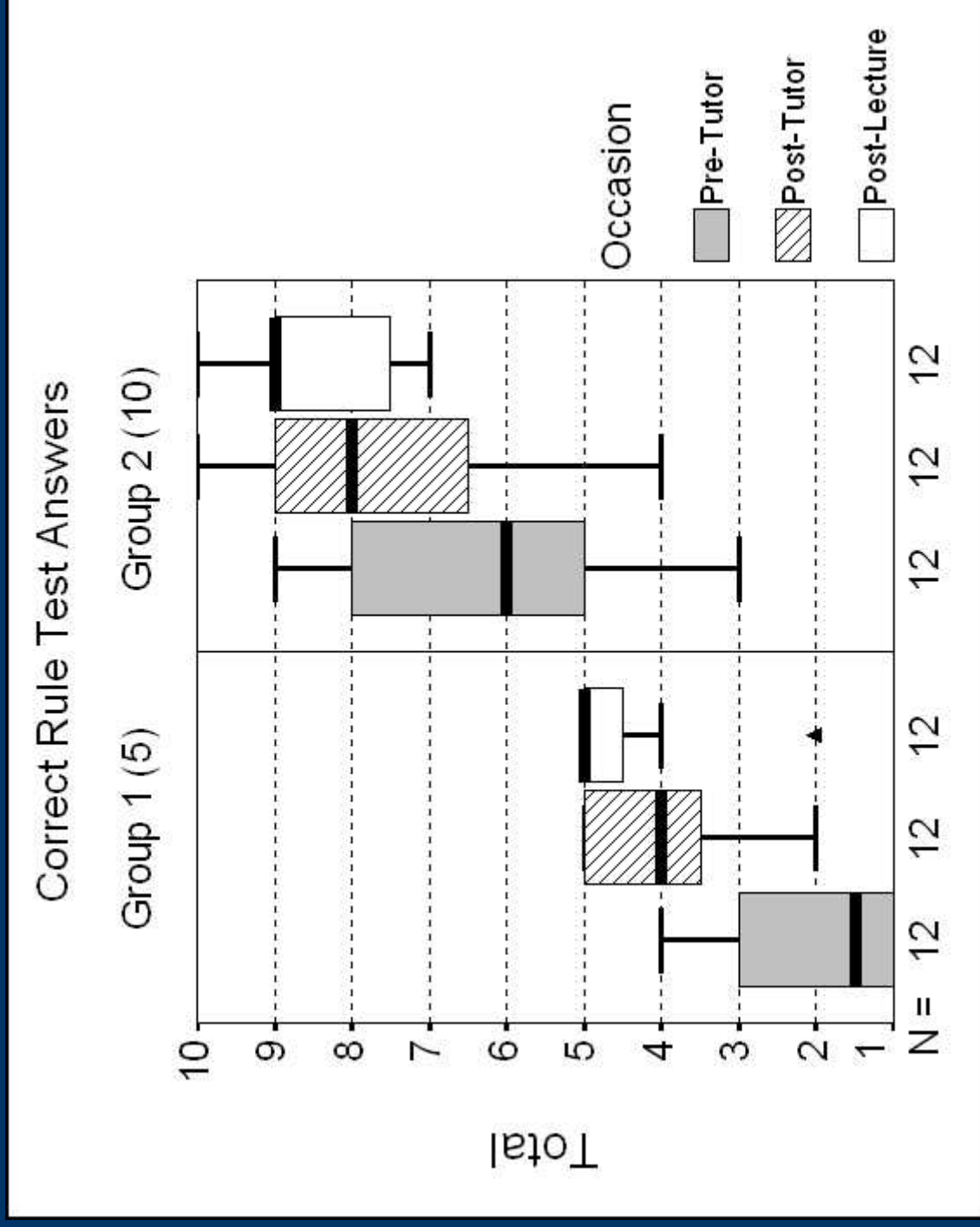
Which one of the below lines declares myJFrame as a potential instance of the JFrame class?

1. myJFrame extends JFrame.
2. JFrame myJFrame:
3. myJFrame JFrame;
4. JFrame myJFrame;

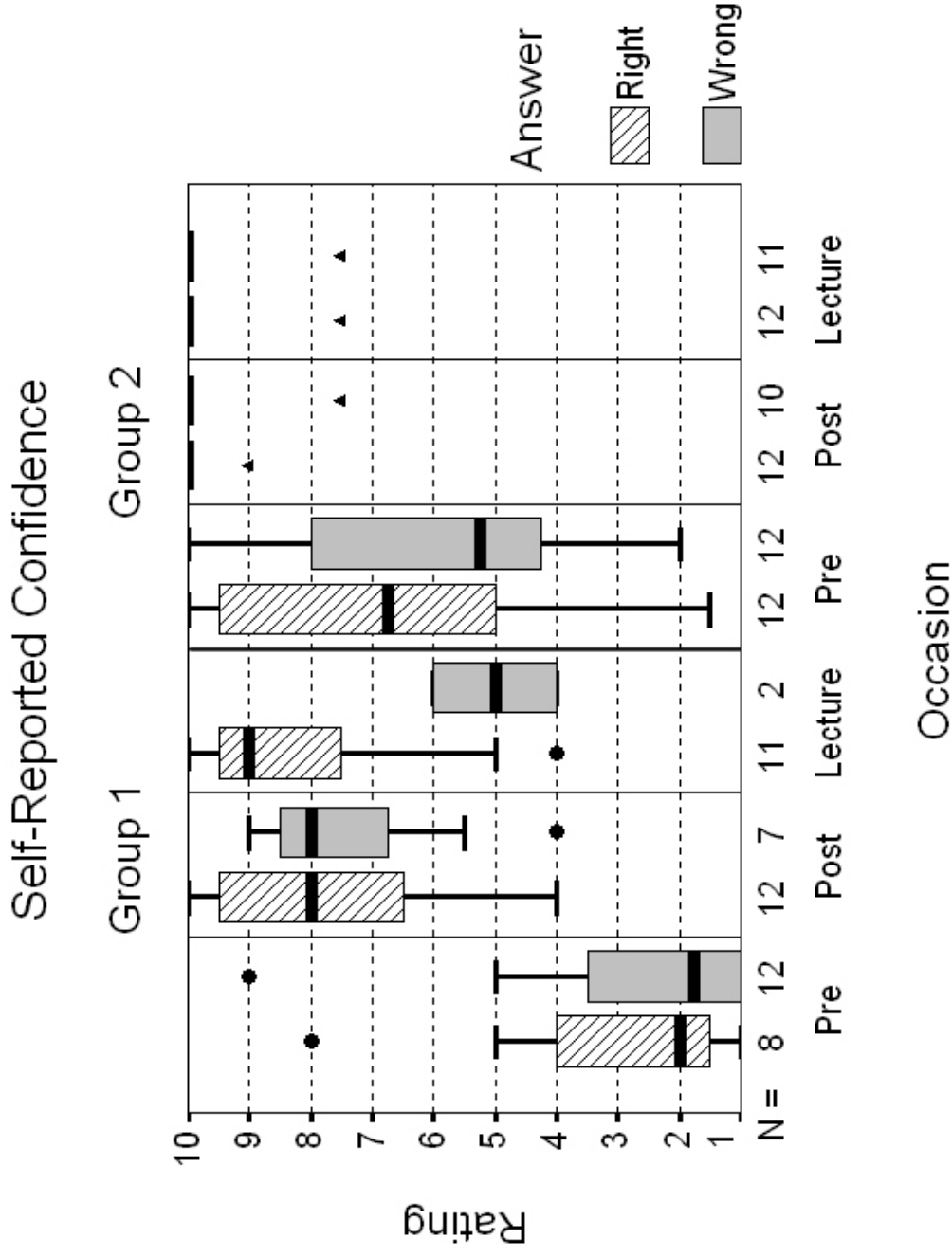
How confident are you that you selected the correct answer?

Not at all confident. 1 2 3 4 5 6 7 8 9 10 Totally confident.

# Correct Rule Test Answers

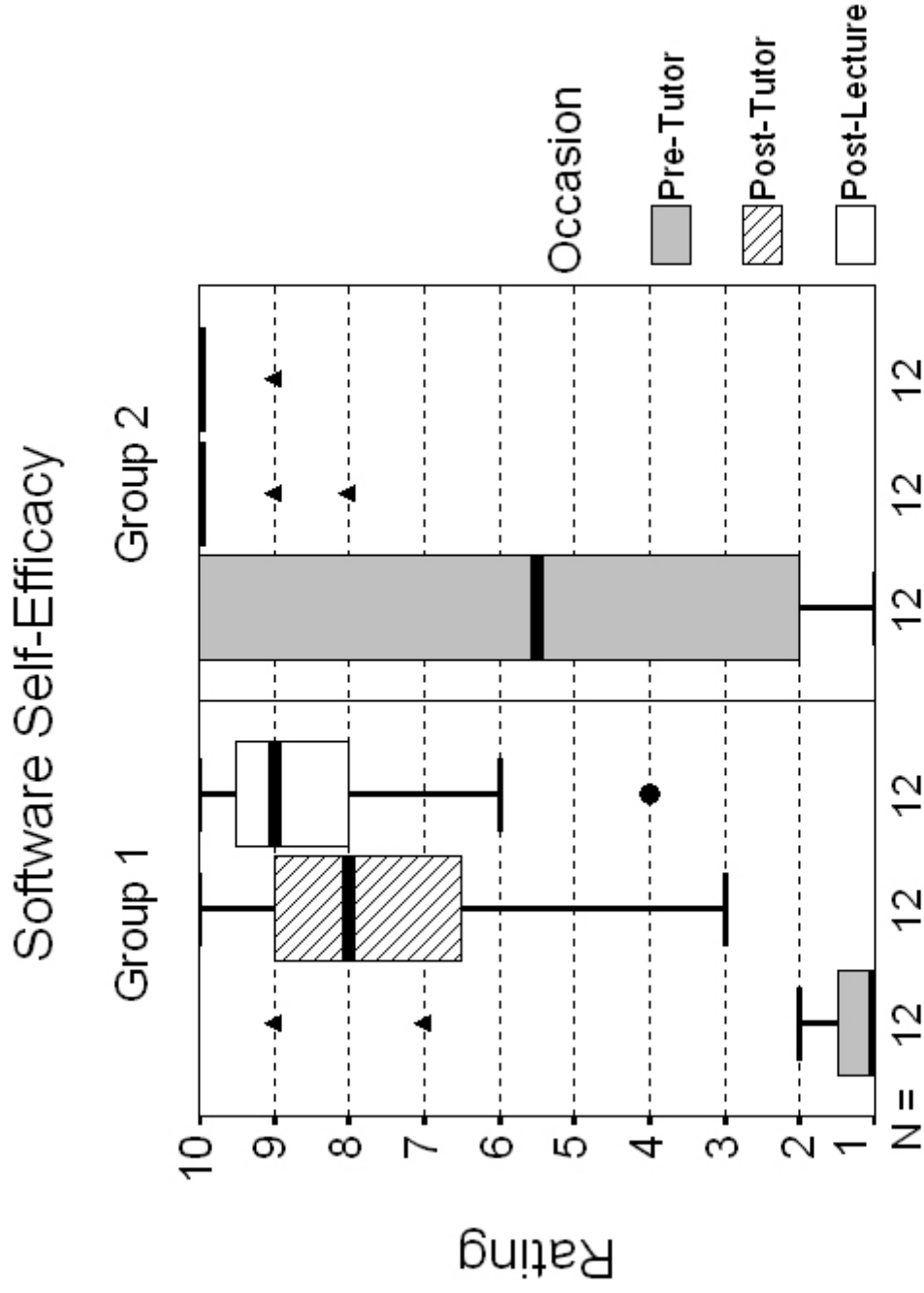


# Confidence in Rule Answers



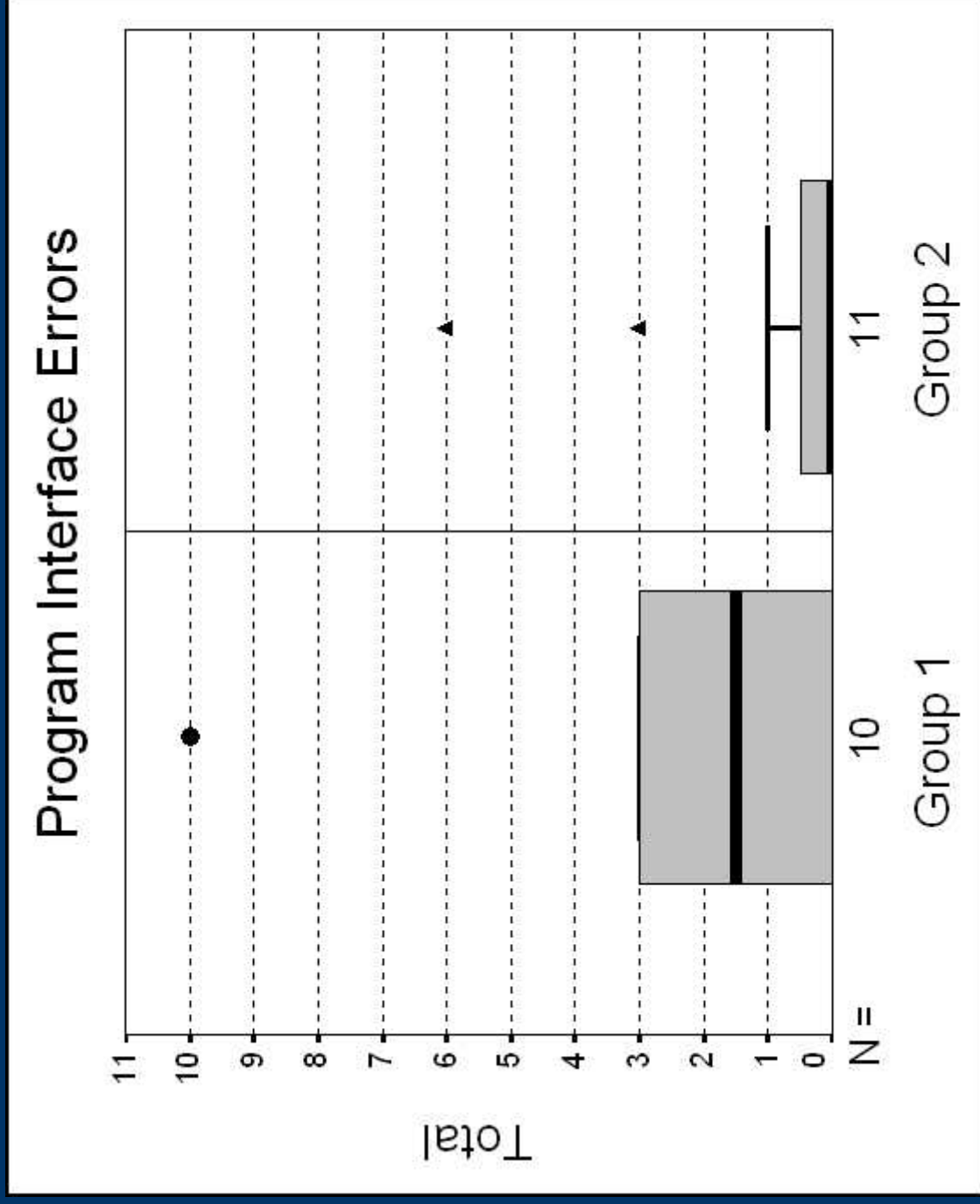
1 = Not Confident ... 10 = Totally Confident

# Software Self-Efficacy

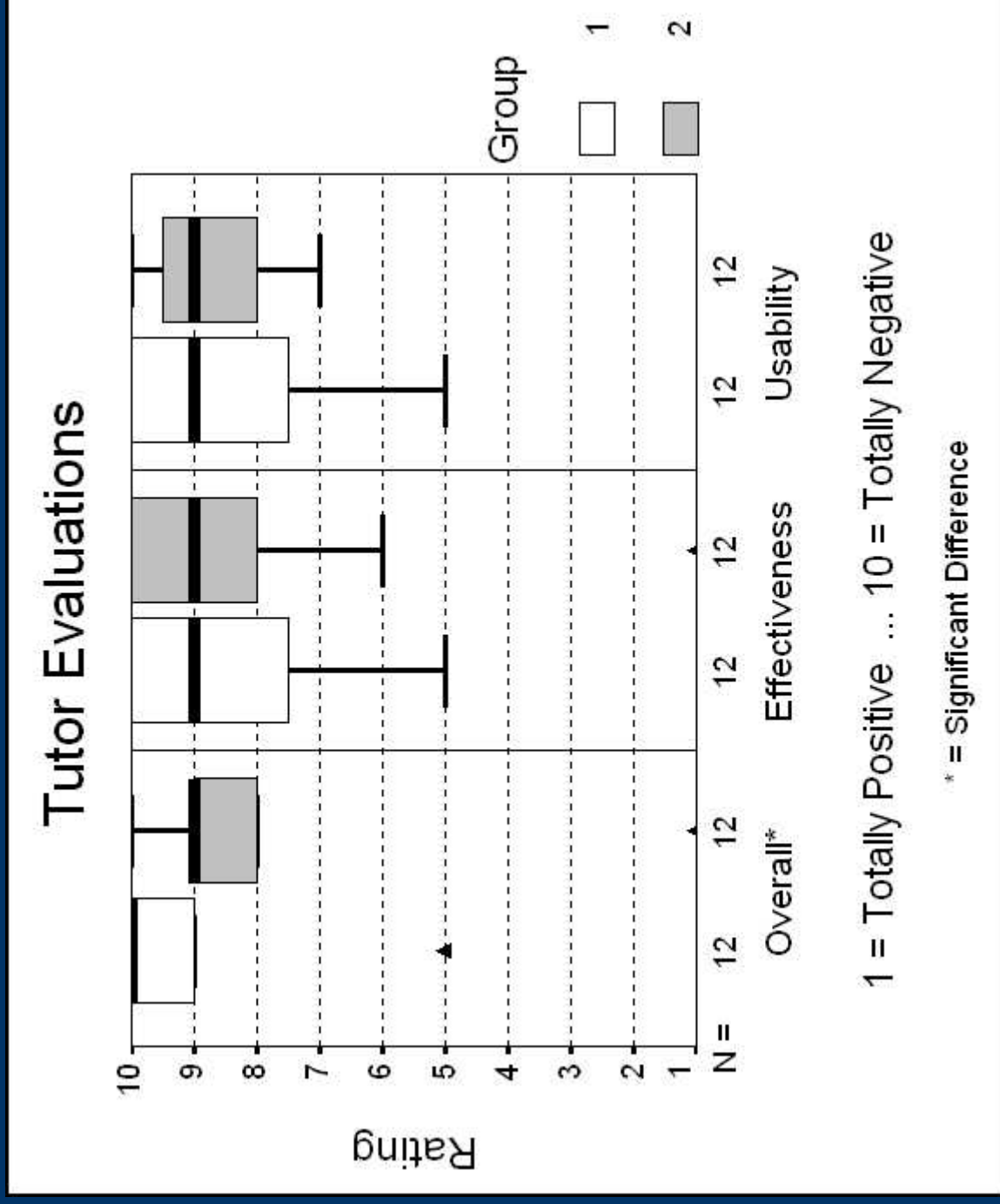


1 = Not Confident ... 10 = Totally Confident

# Program Errors



# Evaluations of the Tutor



# Conclusions

1. Programmed instruction is an effective tool in technology education.
2. It meets the needs of the individual learner.
3. The instructional design can promote meaningful learning and self-confidence.
4. The tutoring system is well-received by novitiate learners.
5. The competency attained sets the occasion for advanced learning with enthusiasm.
6. Students like the tutor, and so do I.

***Thank you!***

**Questions?**