Clevis User Manual

Group 20

1. Introduction

About Command LinE Vector graphIcs Software (Clevis) . Users should be able to create and manipulate vector graphic containing shapes such as line segments, circles, rectangles, and squares using Clevis . Users can operate Clevis through the terminal (CLI mode) or through a GUI viewer.

2. System Requirements

To run the Clevis successfully , please ensure that you have installed the following into your computer:

Download and install Java SE Development Kit 21

https://www.oracle.com/java/technologies/javase/jdk21-archive-downloads.html

Download and install IntelliJ IDEA Community Edition 2024.2

https://www.jetbrains.com/idea/download/other.html

3. Running Instructions

Running in CLI Mode

Open Application.java

To launch Clevis with logging enabled, use the following command format:

Command enter java Application.java -html log.html -txt log.txt -gui

It will contain GUI Mode

A GUI window will appear and update automatically when commands are entered

If you don't want gui

Command enter java Application.java -html log.html -txt log.txt

Then it will run the program

[HTML_File_Path]: The path where the HTML log file will be saved (e.g., d:\log.html).

[TXT_File_Path]: The path where the plain text log file will be saved (e.g., d:\log.txt).

All operations you perform will be recorded in these two files.

4. GUI Mode Usage

When Clevis is executed with:

-gui

A graphical display window will appear.

GUI Behavior

Automatically refreshes when the user enters commands

Draws shapes in order of Z-index

Only visualizes (no mouse drawing)

Supports all shape types and updates dynamically

The GUI acts only as a viewer; all commands must still be typed in the console.

5. Notes

Clevis does not save data to disk

All shapes disappear after quitting

Coordinates follow Swing convention (0,0 at top-left corner)

GUI does not accept mouse input

Console commands remain the main control method

6. Command Line Usage

Below are all the commands supported by Clevis:

Shape Creation Commands

1. Create a Rectangle

Command: rectangle n x y w h

Effect: Creates a new rectangle named n.

Parameters:

n: The unique name for the rectangle.

x, y: The coordinates of the top-left corner.

w: The width of the rectangle.

h: The height of the rectangle.

2. Create a Line

Command: line n x1 y1 x2 y2

Effect: Creates a new line segment named n.

Parameters:

n: The unique name for the line.

x1, y1: The coordinates of the first endpoint.

x2, y2: The coordinates of the second endpoint.

3. Create a Circle

Command: circle n x y r

Effect: Creates a new circle named n.

Parameters:

n: The unique name for the circle.

x, y: The coordinates of the center.

r: The radius of the circle.

4. Create a Square

Command: square n x y l

Effect: Creates a new square named n.

Parameters:

n: The unique name for the square.

x, y: The coordinates of the top-left corner.

l: The side length of the square.

Grouping & Deletion Commands

5. Group Shapes

Command: group n n1 n2 ...

Effect: Creates a new group shape named n from existing shapes n1, n2, etc. After grouping, the individual shapes (n1, n2, ...) cannot be accessed directly until the group is ungrouped.

6. Ungroup a Shape

Command: ungroup n

Effect: Ungroups the group shape n into its component shapes. The group n is deleted, and the individual shapes can now be accessed directly again.

7. Delete a Shape

Command: delete n

Effect: Deletes the shape named n. If n is a group, all shapes within the group are also deleted.

Analysis & Transformation Commands

8. Move a Shape

Command: move n dx dy

Effect: Moves the shape named n horizontally by dx and vertically by dy. If n is a group, all shapes within the group are moved.

9. Find Bounding Box

Command: boundingbox n

Effect: Calculates and displays the minimum bounding box for the shape n.

Output Format: x y w h (the coordinates of the top-left corner, width, and height).

10. Check Intersection

Command: intersect n1 n2

Effect: Reports whether the bounding boxes of shapes n1 and n2 intersect.

11. Find Shape at Point

Command: shapeAt x y

Effect: Returns the name of the top-most shape that covers the point (x, y).

Coverage Rule: A shape covers a point if the minimum distance from the point to the shape's outline is less than 0.05. Shapes created more recently are on top of older shapes.

Information Commands

12. List Shape Information

Command: list n

Effect: Displays the basic information about the shape named n.

For simple shapes (rectangle, line, circle, square), it lists their defining parameters (e.g., center and radius for a circle).

For a group, it lists the group's name and the names of all shapes directly contained within it.

13. List All Shapes

Command: listAll

Effect: Lists the basic information for all shapes in the drawing, ordered from top-most (highest Z-index) to bottom-most. Indentation is used to show the hierarchy of groups and their components.

14. Quit the Tool

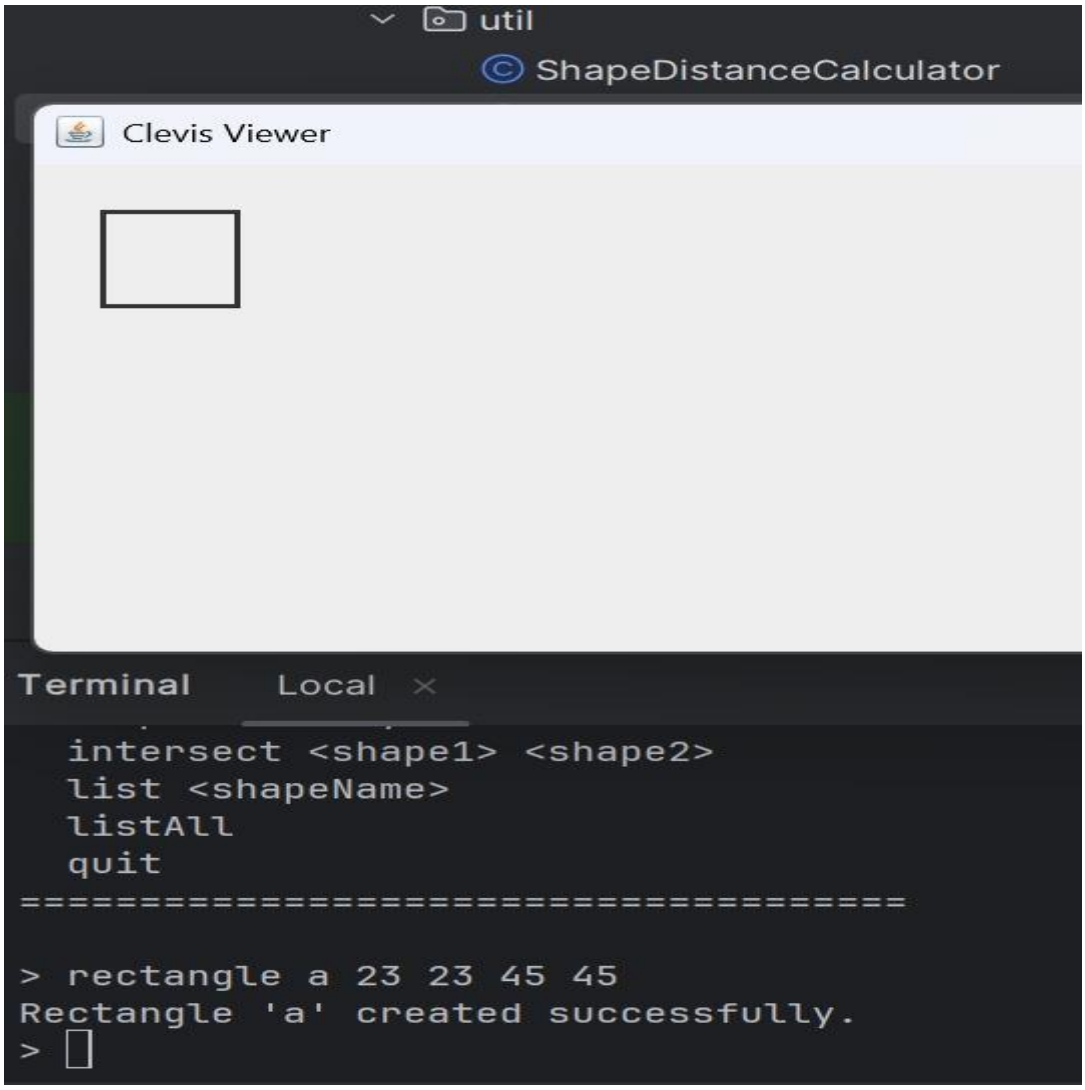
Command: quit

Effect: Terminates the Clevis application.

7. Example Workflow

Enter Rectangle name x y width height

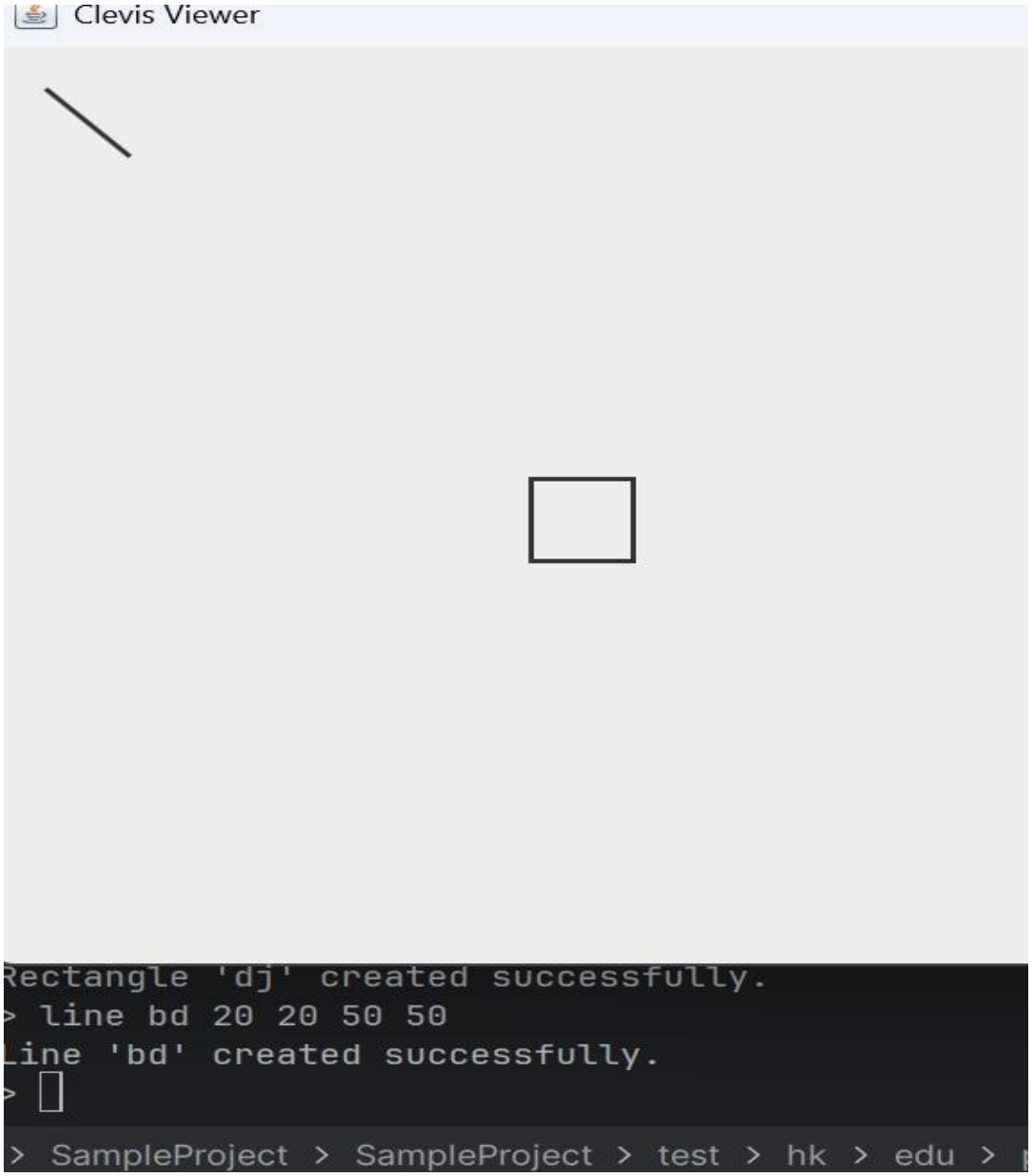
Example: Rectangle a 23 23 45 45



This code will create a rectangle a in the GUI window

Enter line name x1 y1 x2 y2

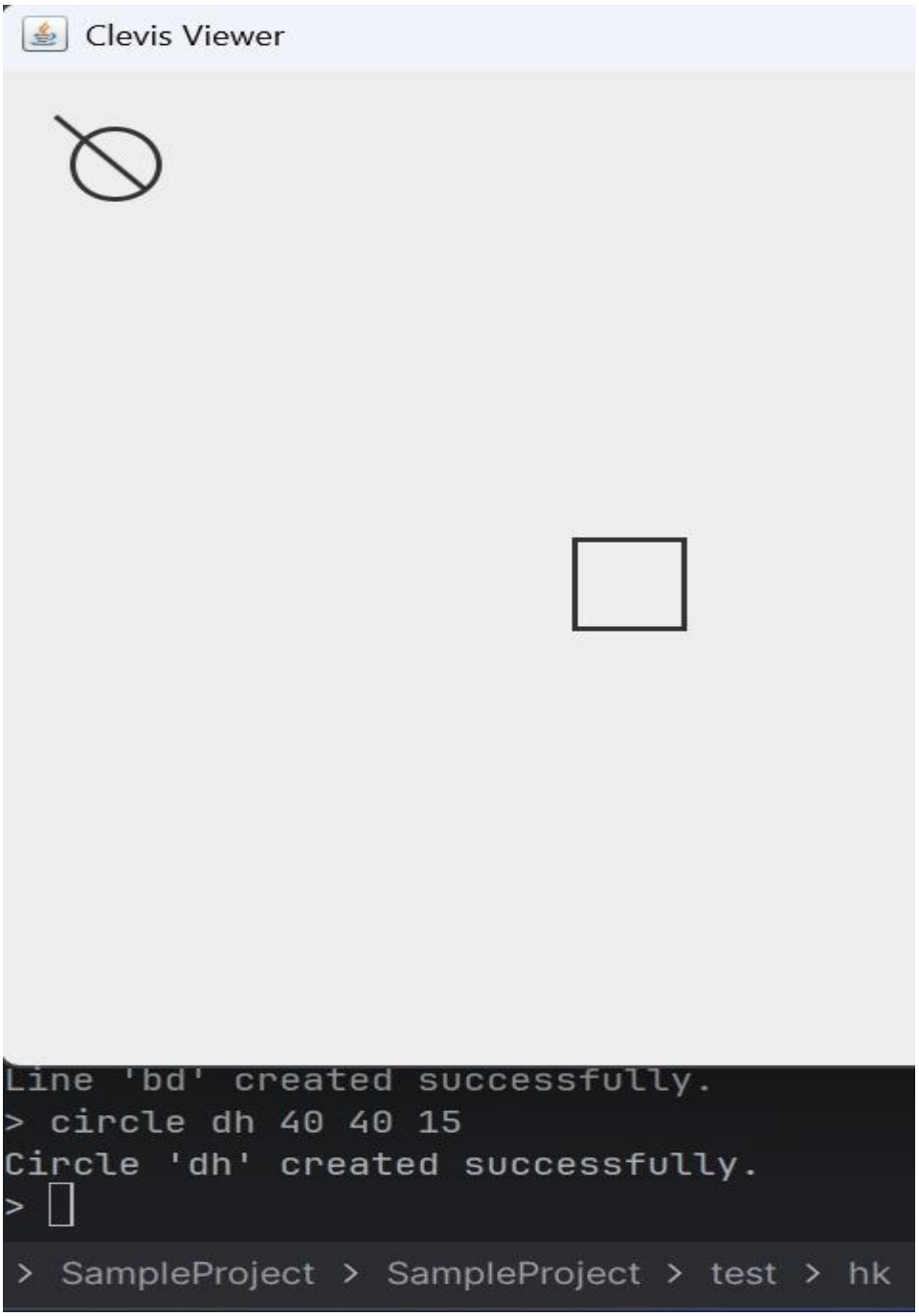
Example: Line bd 20 20 50 50



This code will create a line bd in the GUI window

Enter circle name x y radius

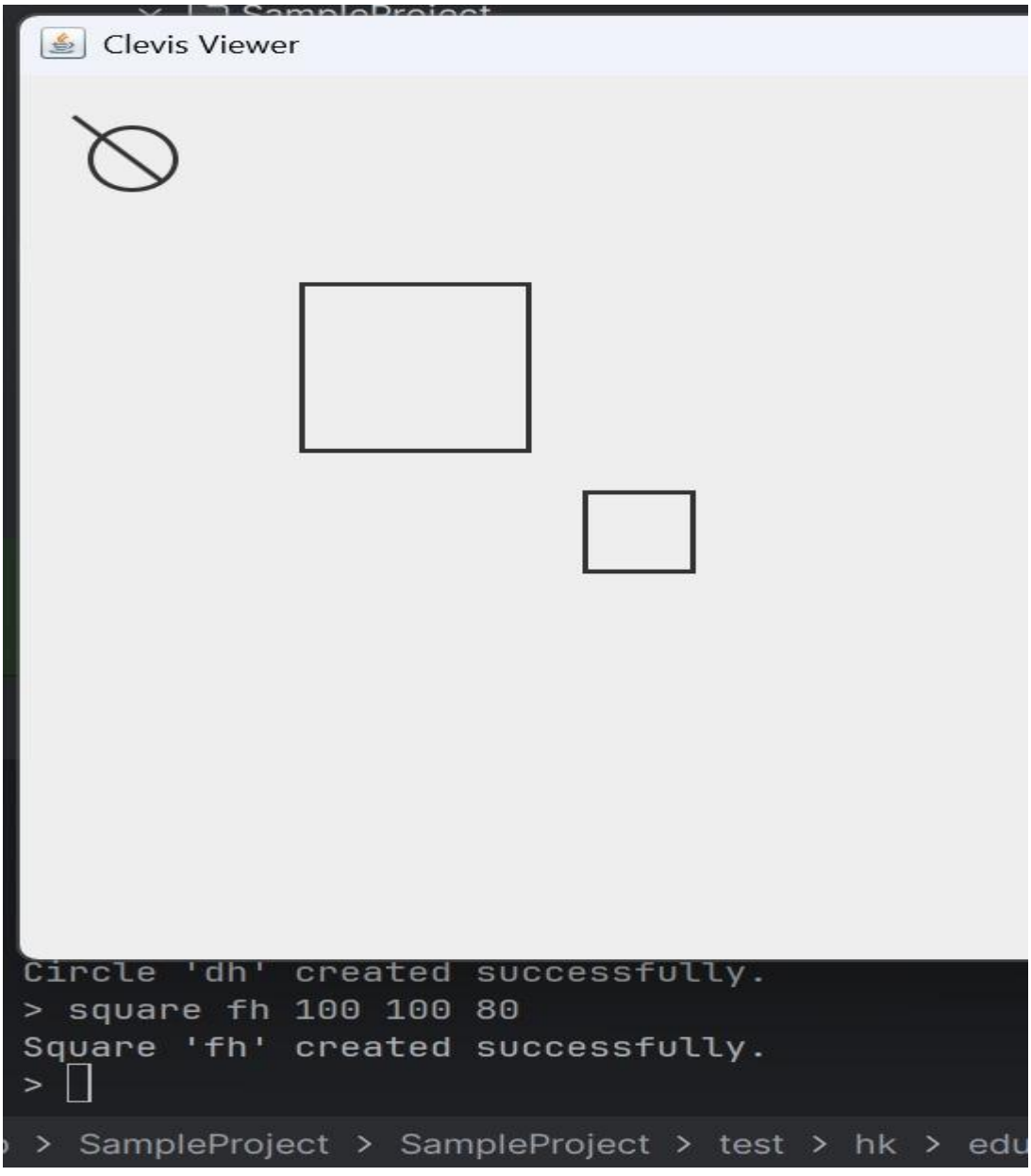
Example: Circle dh 40 40 15



This code will create a Circle dh in the GUI window

Enter Square name x y length

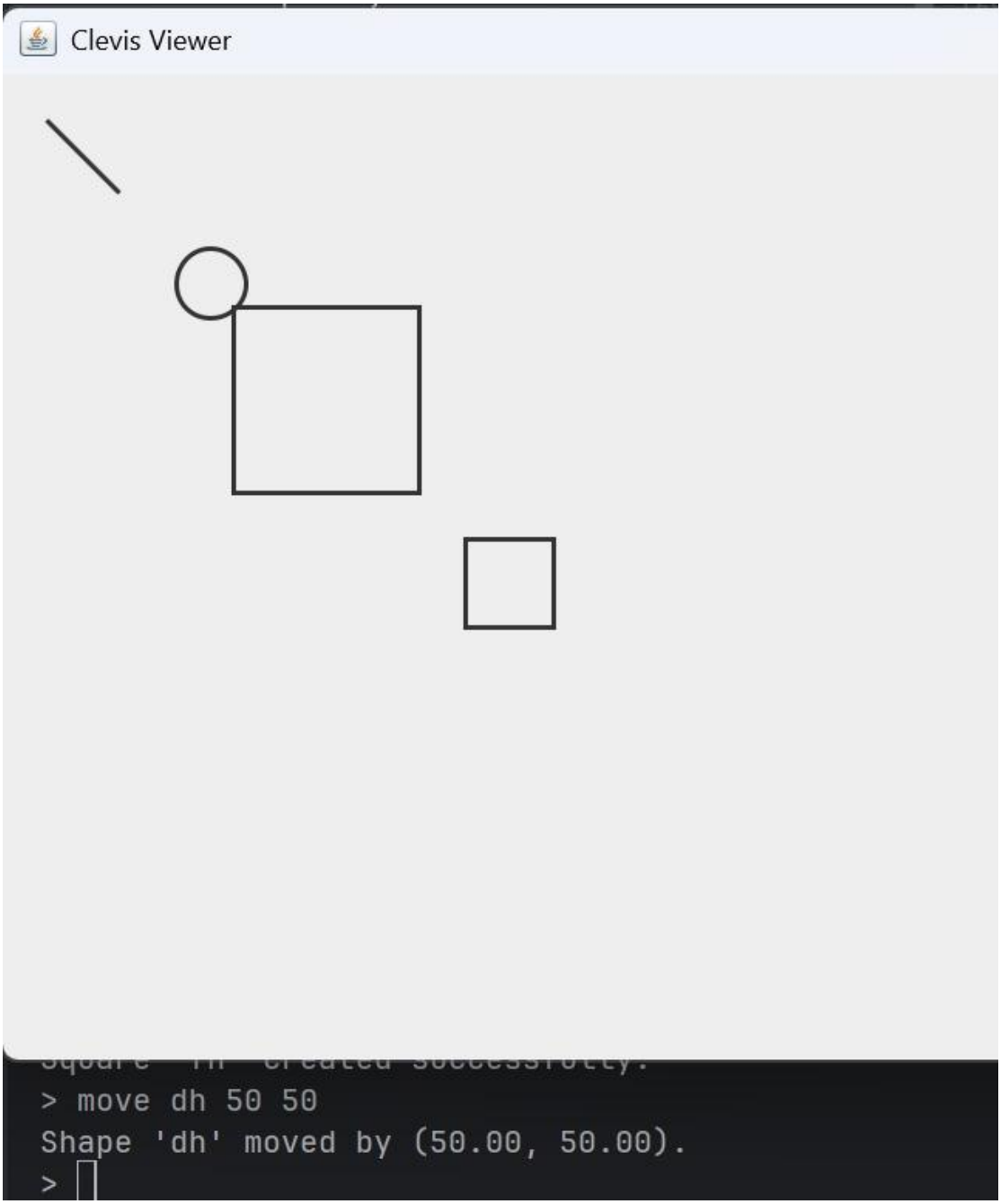
Example: Square fh 100 100 50



This code will create a square fh in the GUI window

Enter move shapename dx dy

Example: Move dh 50 50



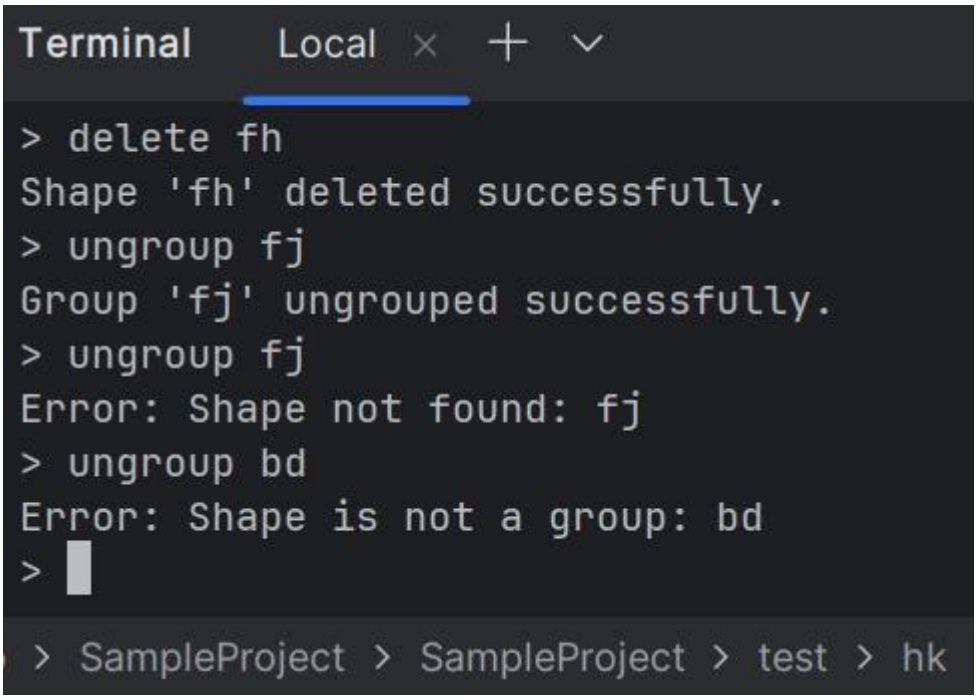
This code will move the Circle dh 50 50 in the GUI window

Enter Group name shapename1 shapename2 ....

Example: Group fj dh fh bd

Enter ungroup name

Example: Ungroup fj



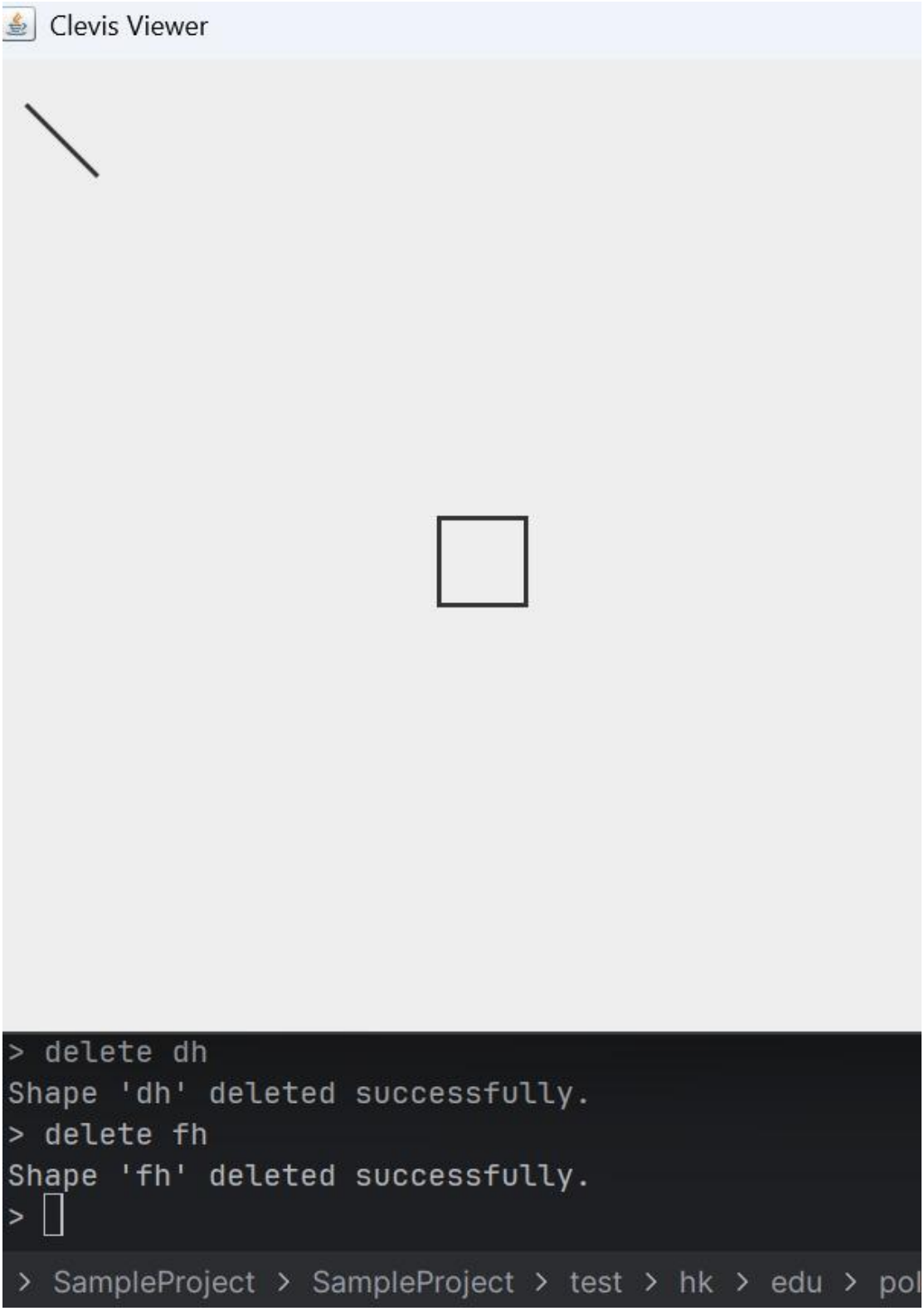
There are two code here

Group will make the shape names which you enter here become to a group

Ungroup will cancel the group you created before

Enter Delete name

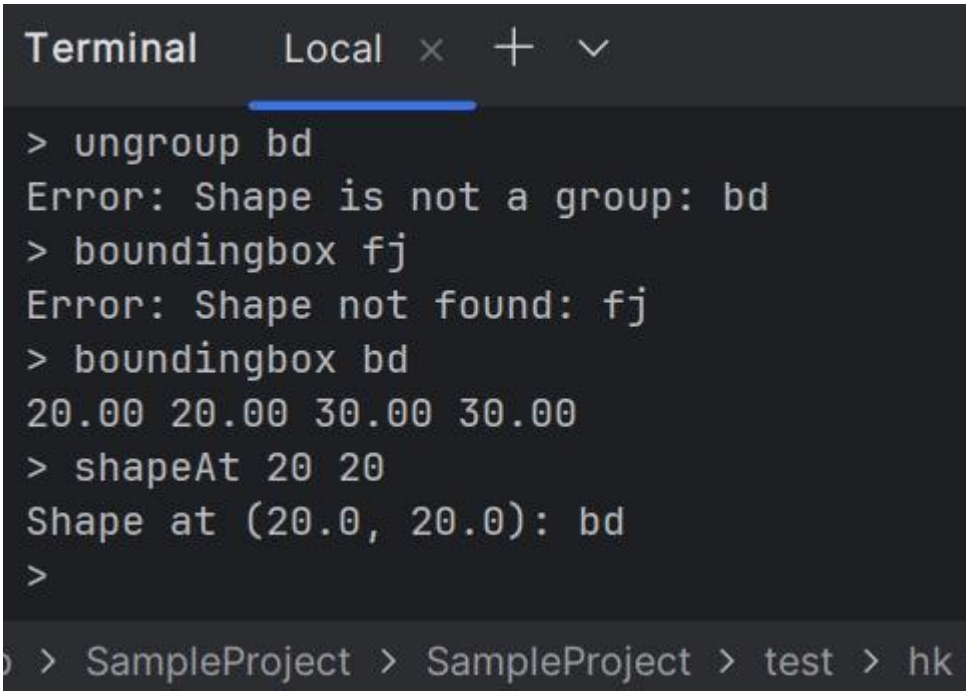
Example: Delete fh, dh



This code will delete the shape you created before

Enter bounding name

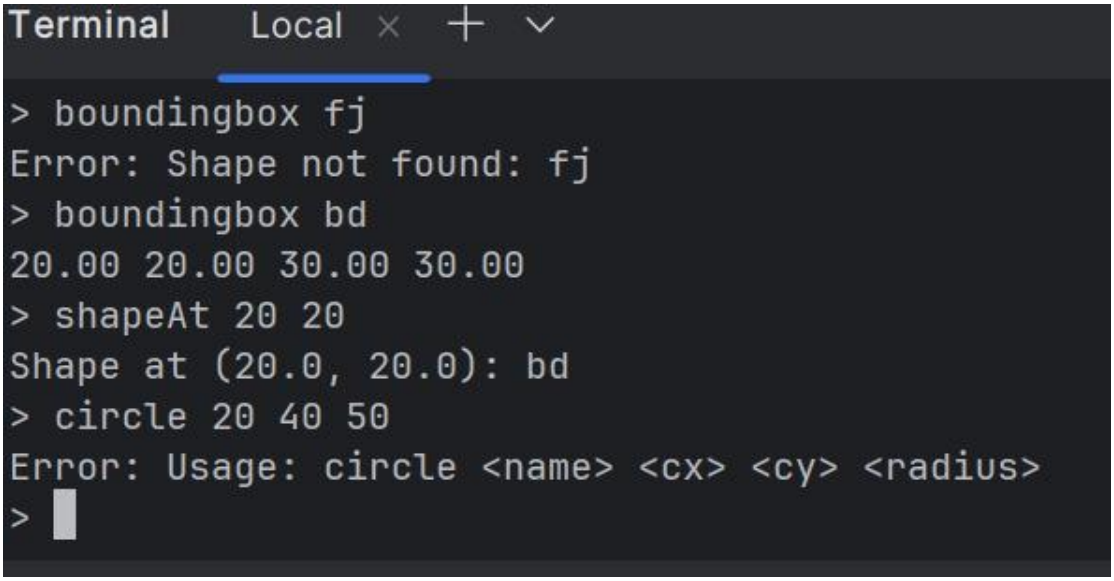
Example: Boundingbox bd



This code will show the two side coordinates of the shape name you enter

Enter shapeAt x y

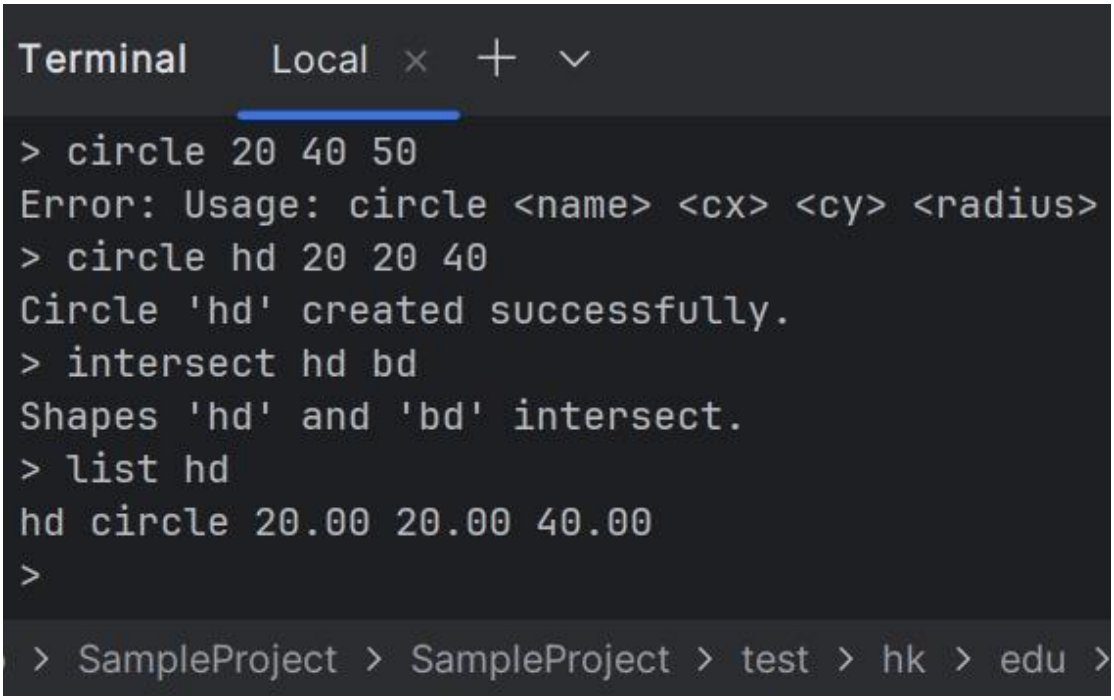
Example: shapeAt 20 20



This code will show the shape which on your enter coordinates

Enter intersect shapename1 shapename2

Example: Intersect hd bd

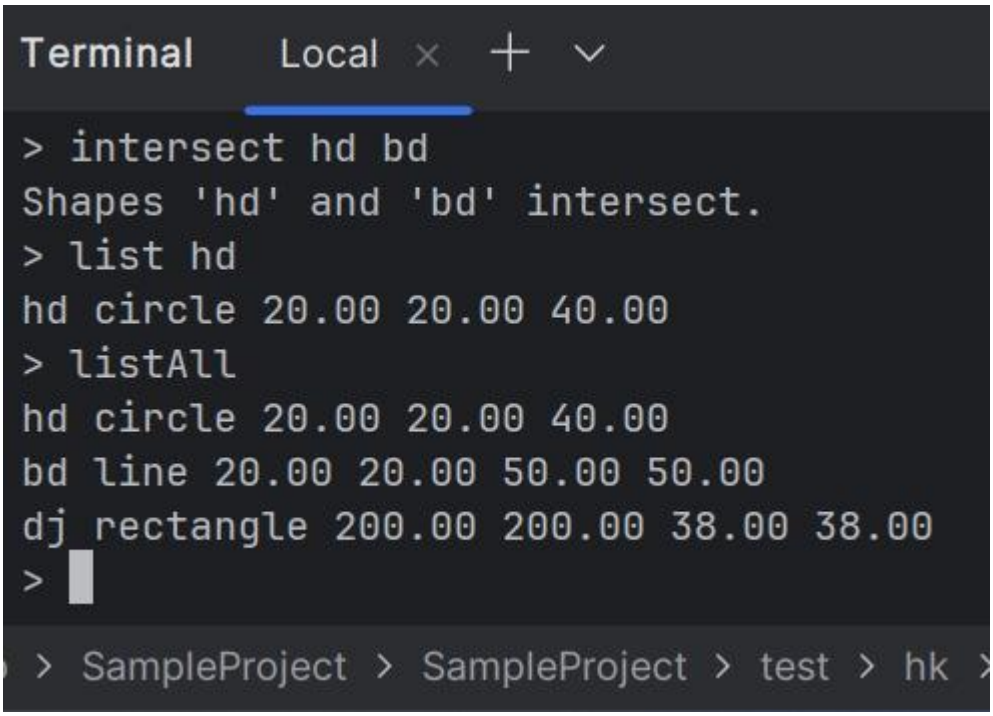


This code will check your enter shape name are they Intersect

Enter List name

Example: list hd

Exampe: listAll



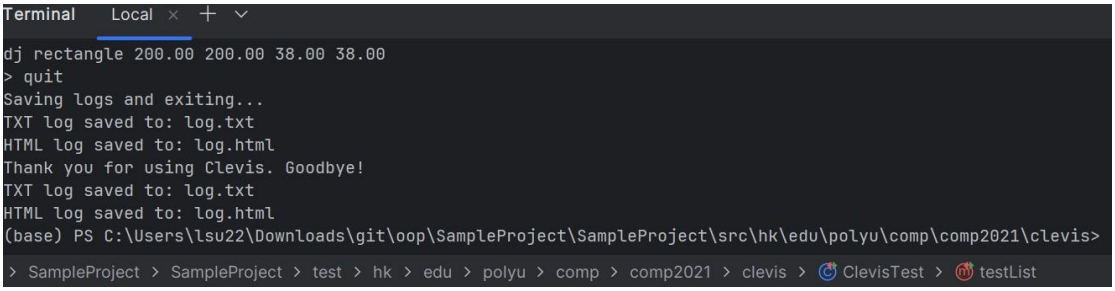
There are two code here

List name will only show one shape information

Listall will show all the shape information which do not be deleted

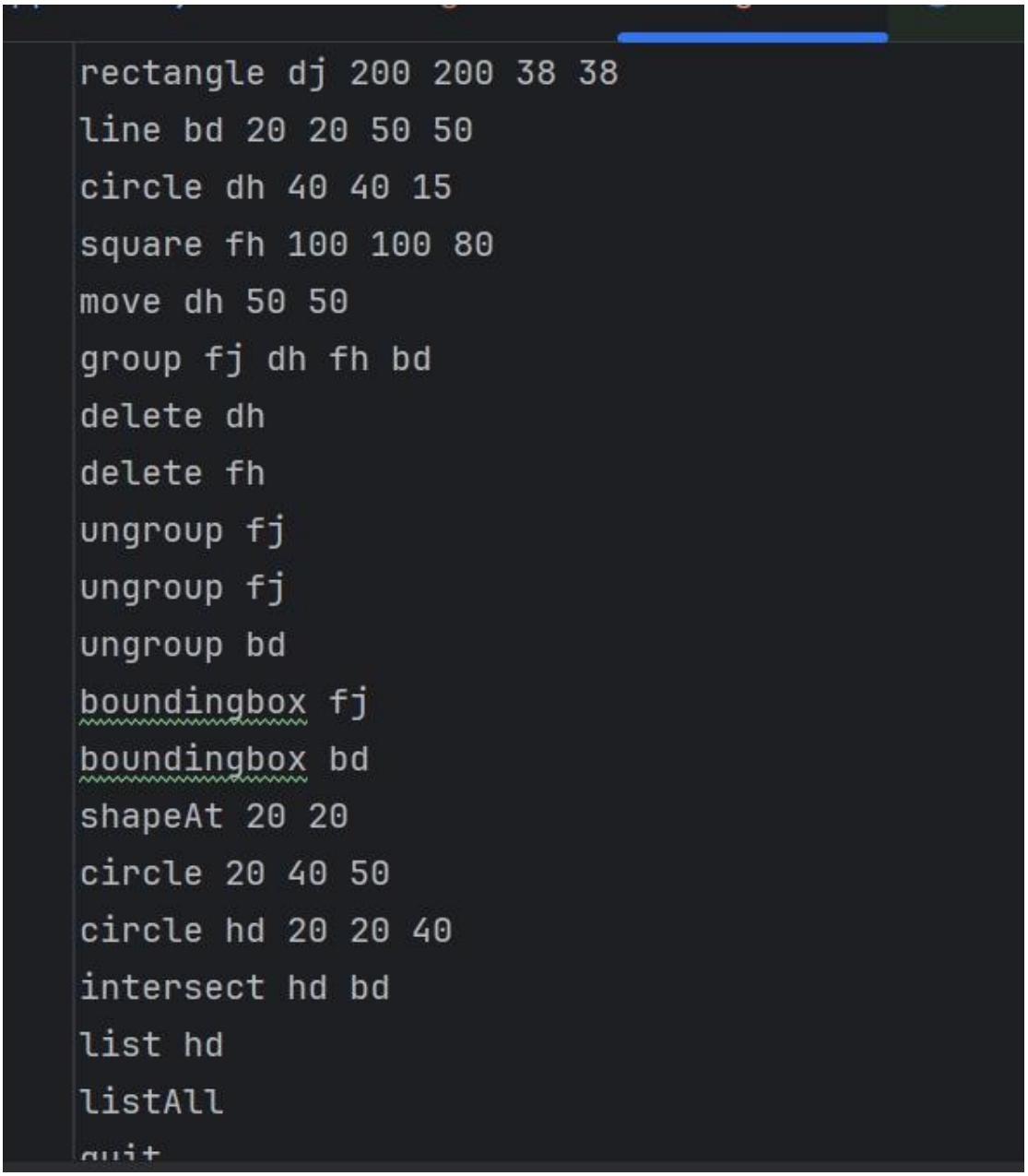
Enter quit

Example: quit



Terminates the execution of Clevis.

Record



It will record the commands

8. Error Handling

Clevis provides user-friendly messages when errors occur:

Possible error situations:

1.Duplicate shape name

2.Invalid numeric values (e.g., negative width and height)

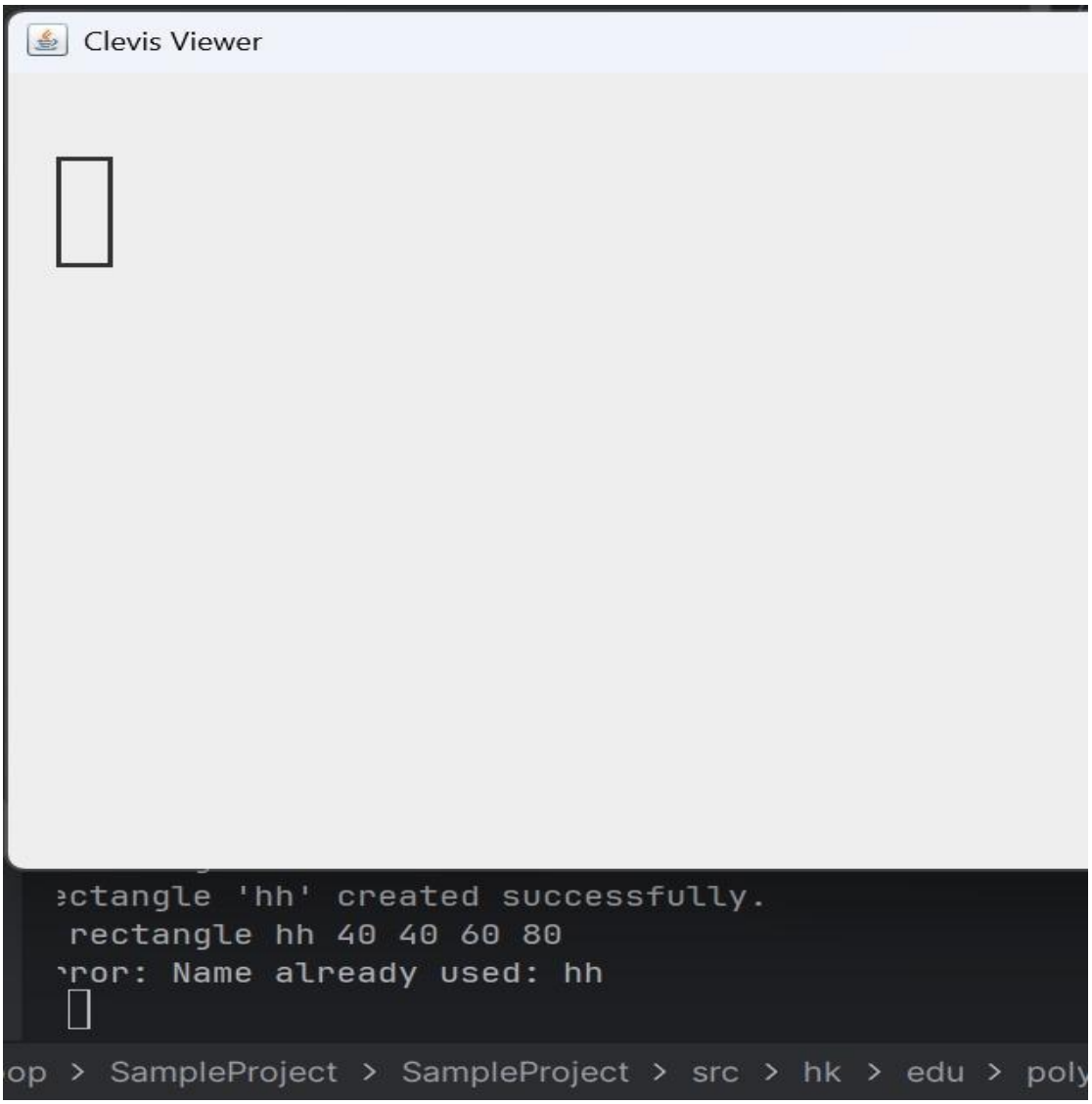
3.Shape without name

Sample messages:

Error: name already used

Example: User enters a shape name which already use

Name hh have been used before



User should change another name for the shape

Error: radius/width and height must be positive

Example: User enters a negative radius/width and height

-45 radius

```
> rectangle a 23 23 45 45
Rectangle 'a' created successfully.
> rectangle c 23 23 -34 34
Error: width and height must be positive!!
> circle c 30 30 -45
Error: radius must be positive!!
>
```

Some value such as radius, width and height must be positive. User should change them

Error: no name

Example: User forget to create a name for the shape

```
Terminal    Local
> boundingbox fj
Error: Shape not found: fj
> boundingbox bd
20.00 20.00 30.00 30.00
> shapeAt 20 20
Shape at (20.0, 20.0): bd
> circle 20 40 50
Error: Usage: circle <name> <cx> <cy> <radius>
>
```

User need to create a name for the shape

just enter the commands again this time don’t forget the name