



---

# JABBERPOINT MANUAL

---

Thu Tran – Bocheng Peng



02 APRIL 2025

NHL STENDEN UNIVERSITY OF APPLIED SCIENCE  
SOFTWARE QUALITY

## Table of Contents

1. Introduction .....	2
2. Project Initialisation .....	2
2.1. DTAP .....	2
3. Application Navigation .....	4
4. File Control .....	5
4.1. Open.....	5
4.2. Save .....	6
4.3. New.....	8
4.4. Exit .....	9
5. Navigation Bar.....	10
5.1. Go to .....	10
5.2. Prev.....	11
5.3. Next .....	11
6. About Application .....	12
7. Testing.....	13

# 1. Introduction

JabberPoint is a primitive slide-show program in Java. Copyright (c) 1995-1997 by Ian F. Darwin, [ian@darwinsys.com](mailto:ian@darwinsys.com). Adapted by Gert Florijn (version 1.1) and Sylvia Stuurman (version 1.2 and higher) for the Open University of the Netherlands, 2002 -- now. Author's version available from <http://www.darwinsys.com/>

The software quality course aims to refactor the application to implement design patterns correctly and efficiently. This refactored version has the composite, command and factory method design patterns. Aside from that, a single is also applied to SlideItemFactory.

The project is available at <https://github.com/traanf-thuw/jabberpoint-refactor>

The README.md file provides a more technical description of the application. This manual aims to provide a working instruction on how to run the application correctly. If you encounter any errors while running, please contact [thu.tran1@student.nhlstenden.com](mailto:thu.tran1@student.nhlstenden.com) or [bocheng.peng@student.nhlstenden.com](mailto:bocheng.peng@student.nhlstenden.com)

## 2. Project Initialisation

When you open the folder with IntelliJ, please make sure that you enabled the maven project and that the pom.xml file is built to initialise all the dependencies.

### 2.1. DTAP

Open the Maven tab on the right side of IntelliJ to start running the lifecycle

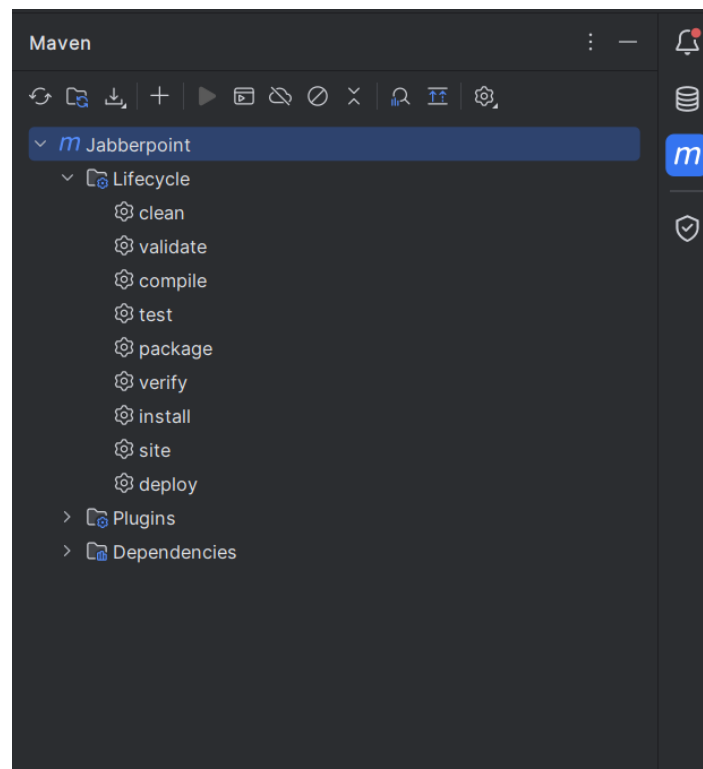
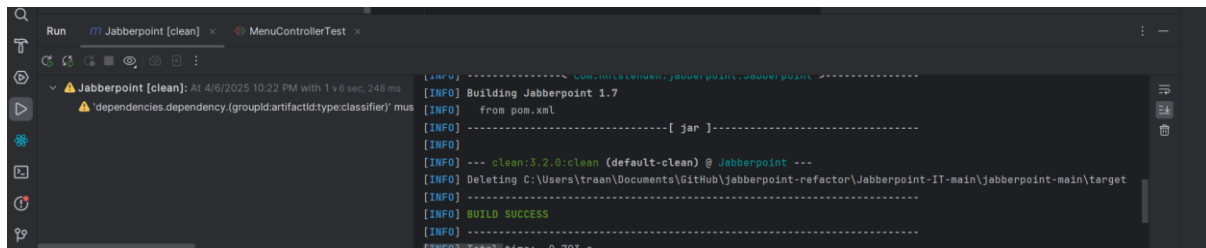


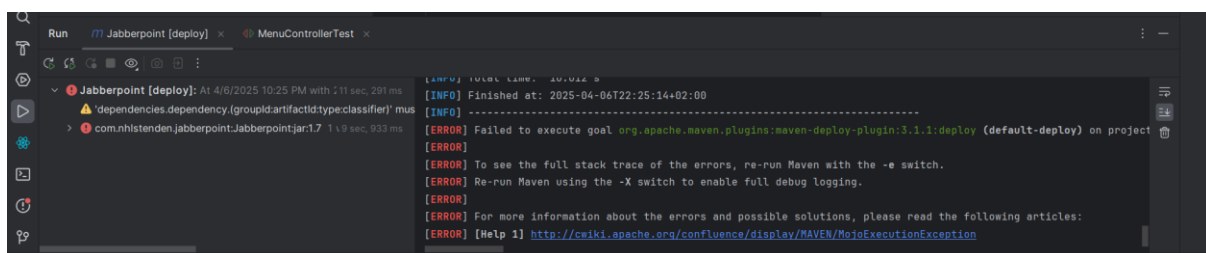
Fig 2.1: Jabberpoint lifecycle

Double click on the action (for example “clean”), the “run window” will popped up and show the build status. For example, the image below shows a successful build.



*Fig 2.2: Successful build for “clean”*

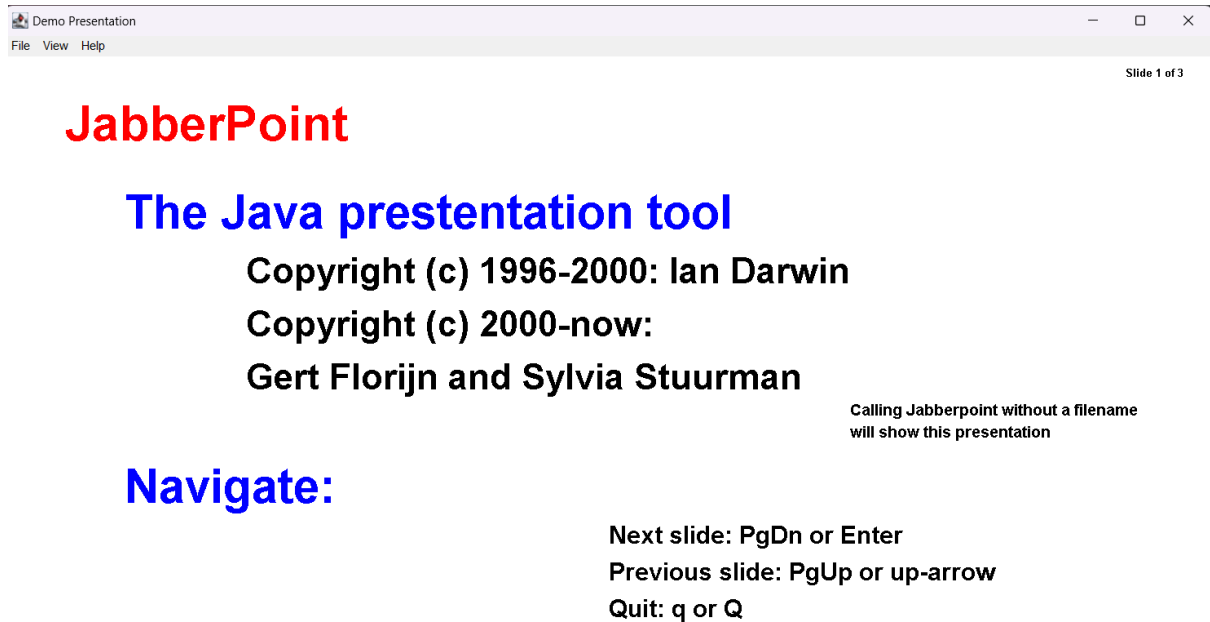
Continue to run the remaining actions. However, the deploy action cannot be run because we do not host the application anywhere, so it will show errors like this while running.



*Fig 2.3: Failed build for “deploy”*

### 3. Application Navigation

You can go to class Jabberpoint to run the application. When running successfully, the UI will look like this.



*Fig 3.1: Application UI*

On the top left corner of the application, there are 3 menu bar which are file, view and edit. The application has different titles and images.

## 4. File Control

The file menu bar contains 4 buttons to open presentation, save presentation, create new presentation and exit the application.

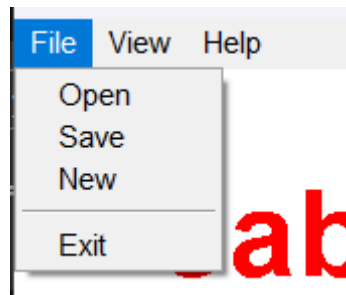


Fig 4.1: File menubar

### 4.1. Open

When you click on the open button, a dialog box will appear to browse files on your device.

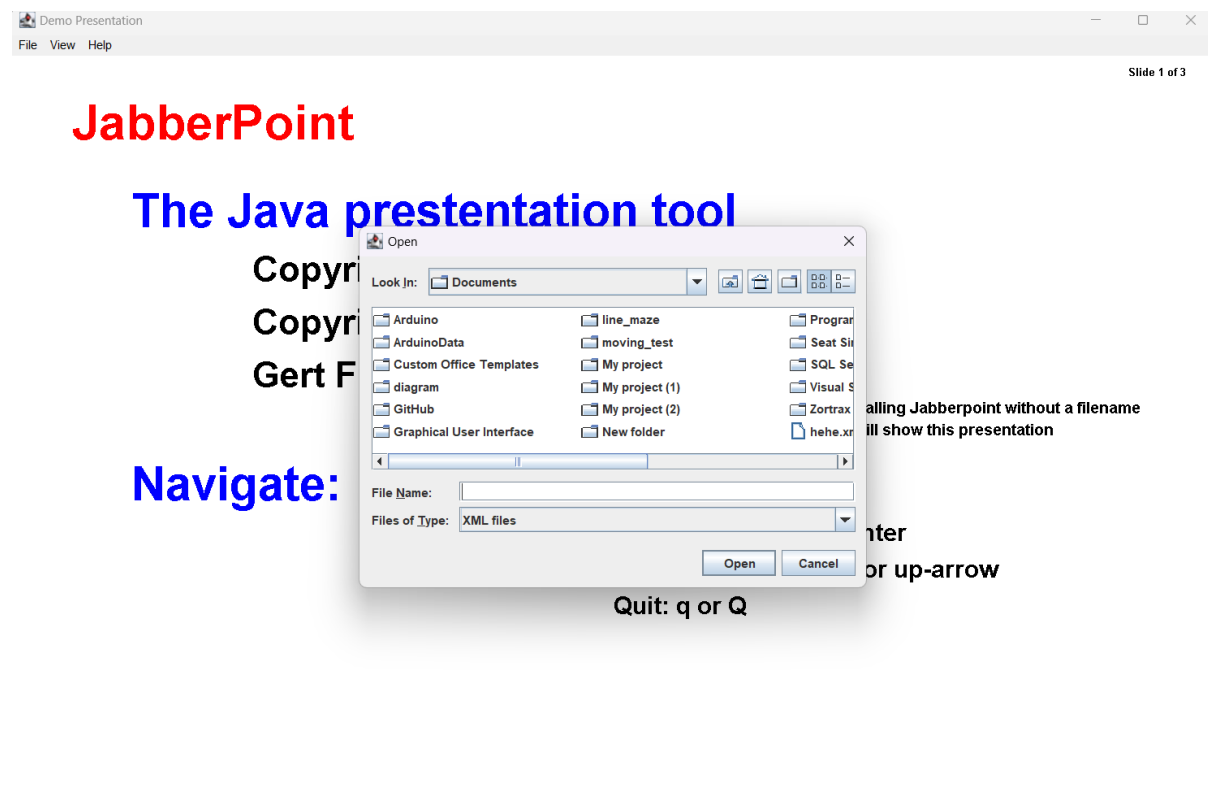


Fig 4.1: Browse files on your device

Then you can choose the test.xml file in the Jabberpoint-refactor folder to open a presentation or choose any other presentation in .xml format you have. The chosen presentation will clear the frame and display on the entire screen.

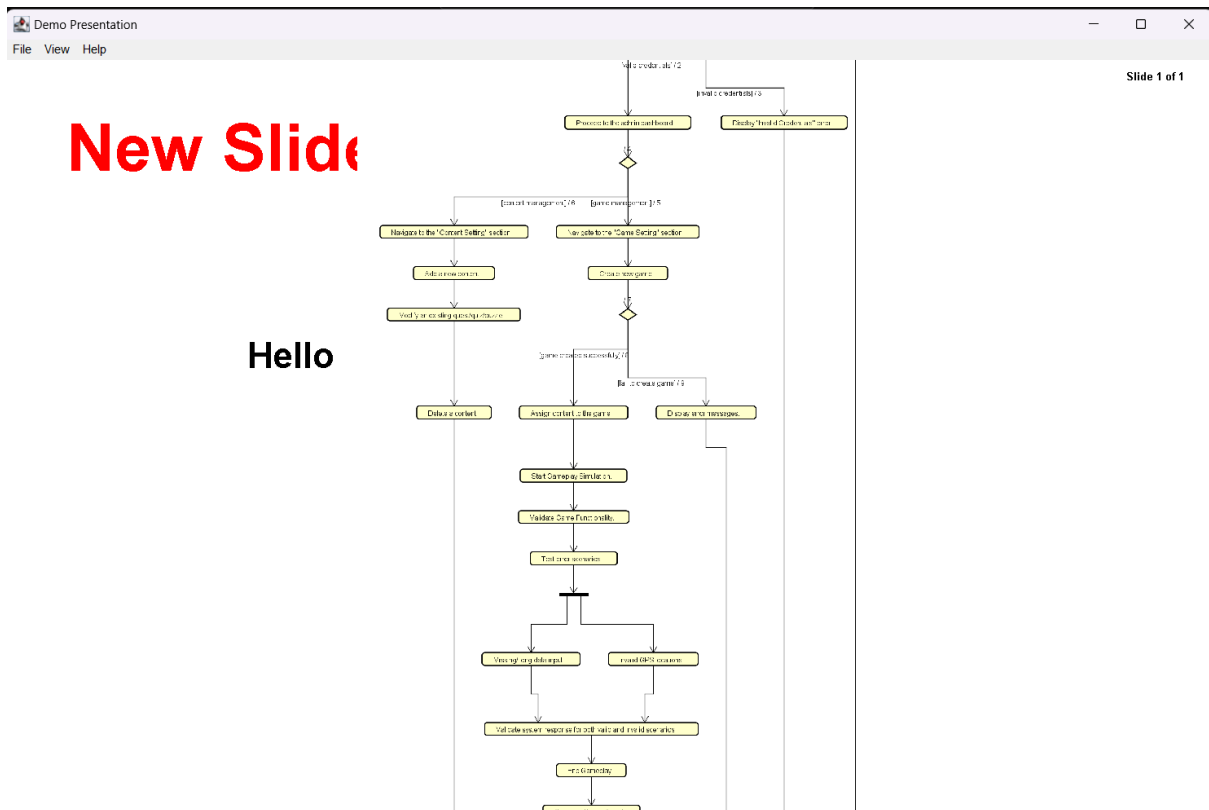


Fig 4.2: Open new presentation

## 4.2. Save

To save the current presentation, you can click on the button save and choose the folder to save it. Then after saving you can open to see if the file is saved correctly.



JabberPoint

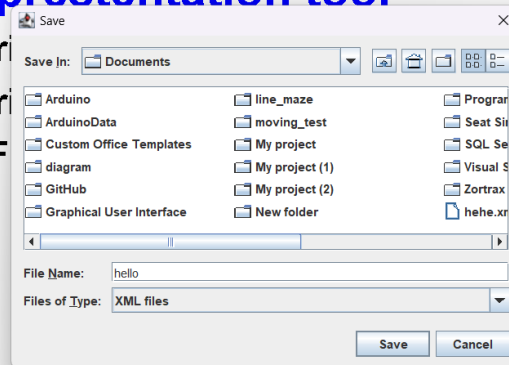
The Java presentation tool

Copyr

Copyr

Gert F

Navigate:



calling Jabberpoint without a filename  
will show this presentation

Enter  
or up-arrow

Quit: q or Q

Fig 4.3: Name the file "hello"



JabberPoint

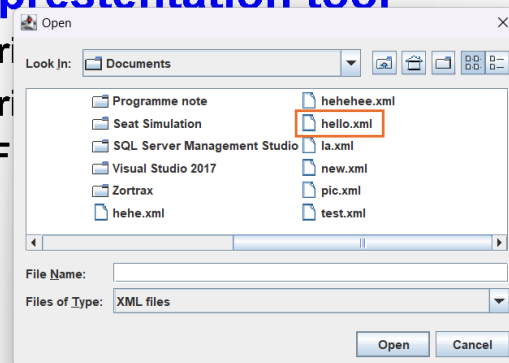
The Java presentation tool

Copyr

Copyr

Gert F

Navigate:



calling Jabberpoint without a filename  
will show this presentation

Enter  
or up-arrow

Quit: q or Q

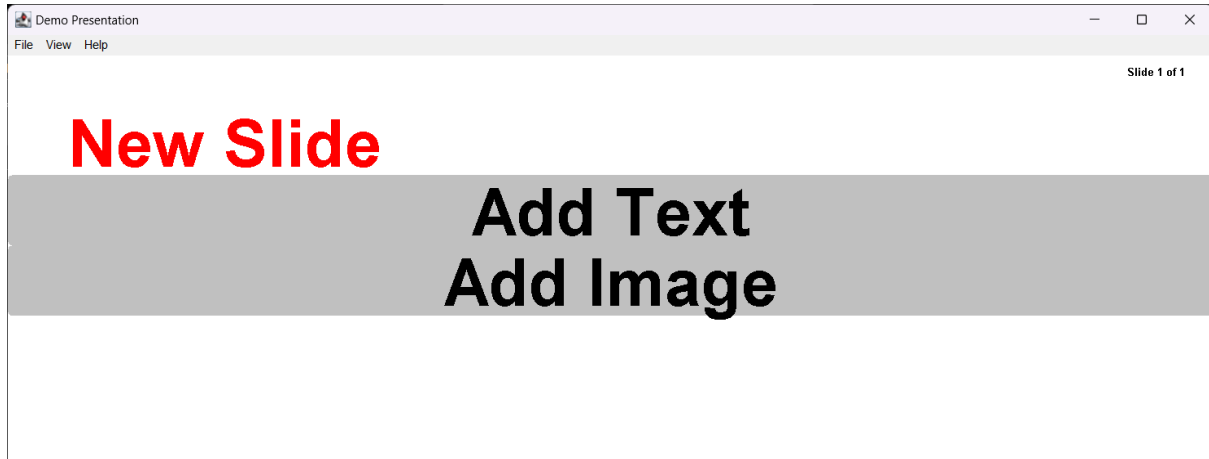
Fig 4.4: Find the file "hello" to open

The presentation should open now.

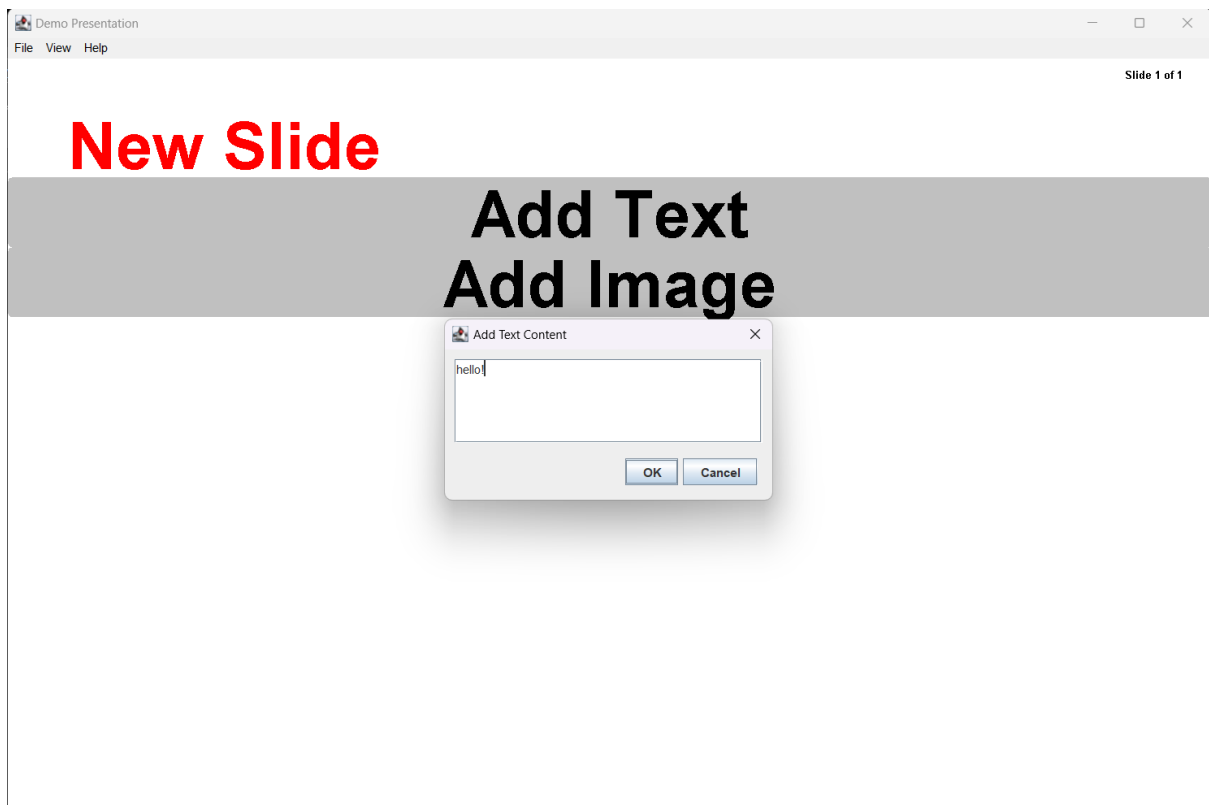


### 4.3. New

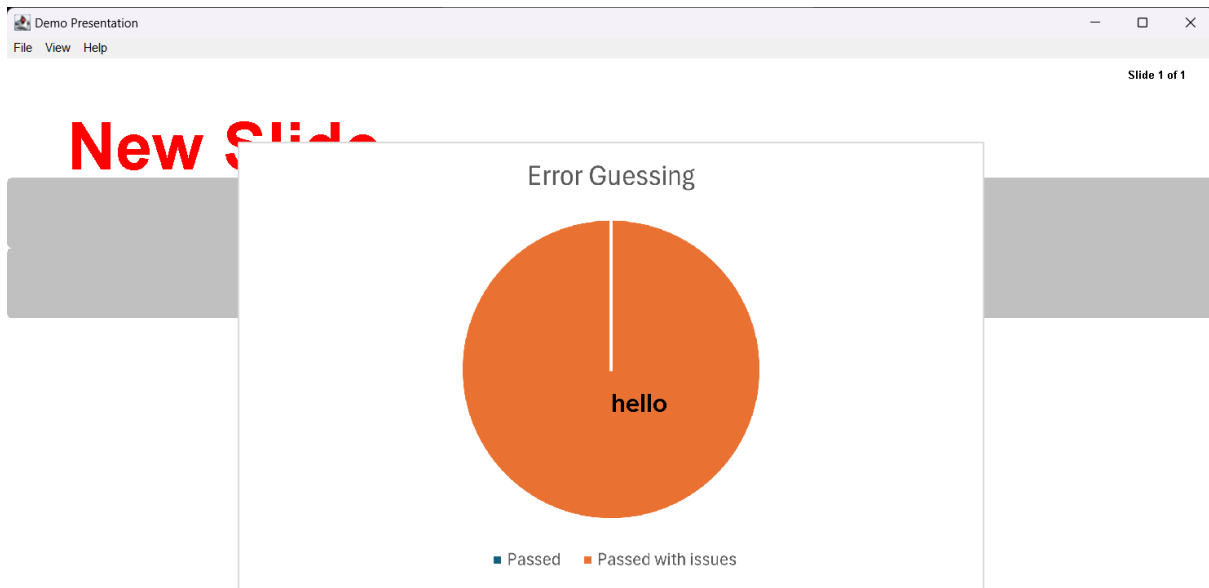
When you click the new button, it will allow you to create a new presentation. Because this function is extra work, I did not make it create a new slide but just one slide and contain a lot of slide components.



*Fig 4.5: Add text and image UI*

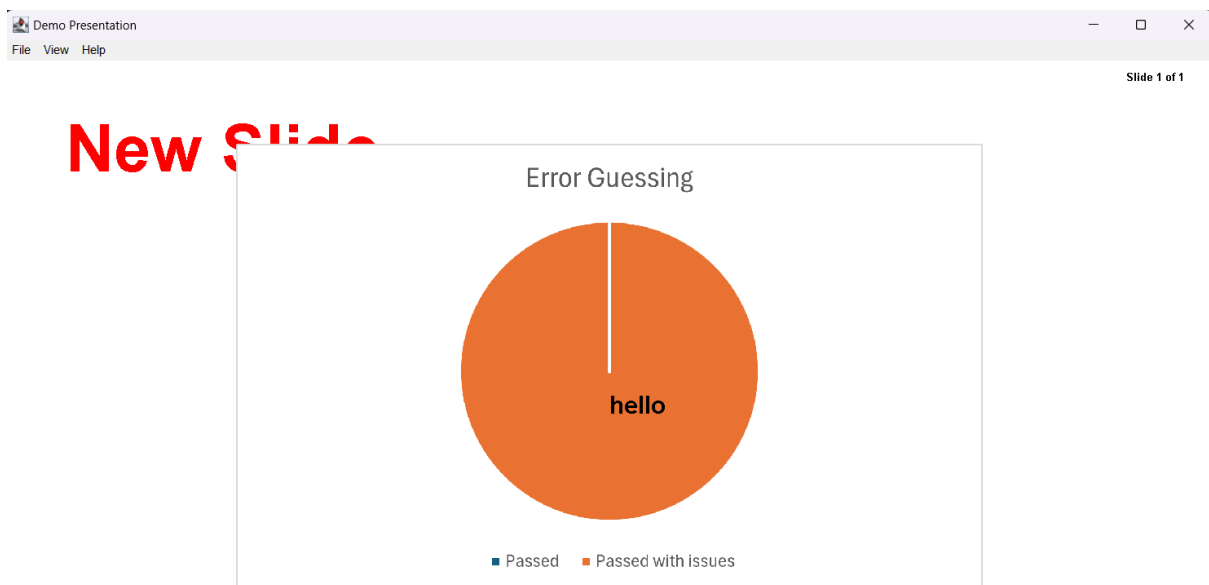


*Fig 4.6: Add text item*



*Fig 4.7: Add bitmap item*

You can save the presentation and try to open it. When you open the presentation, the buttons to add text and images will disappear.



*Fig 4.8: Open the presentation*

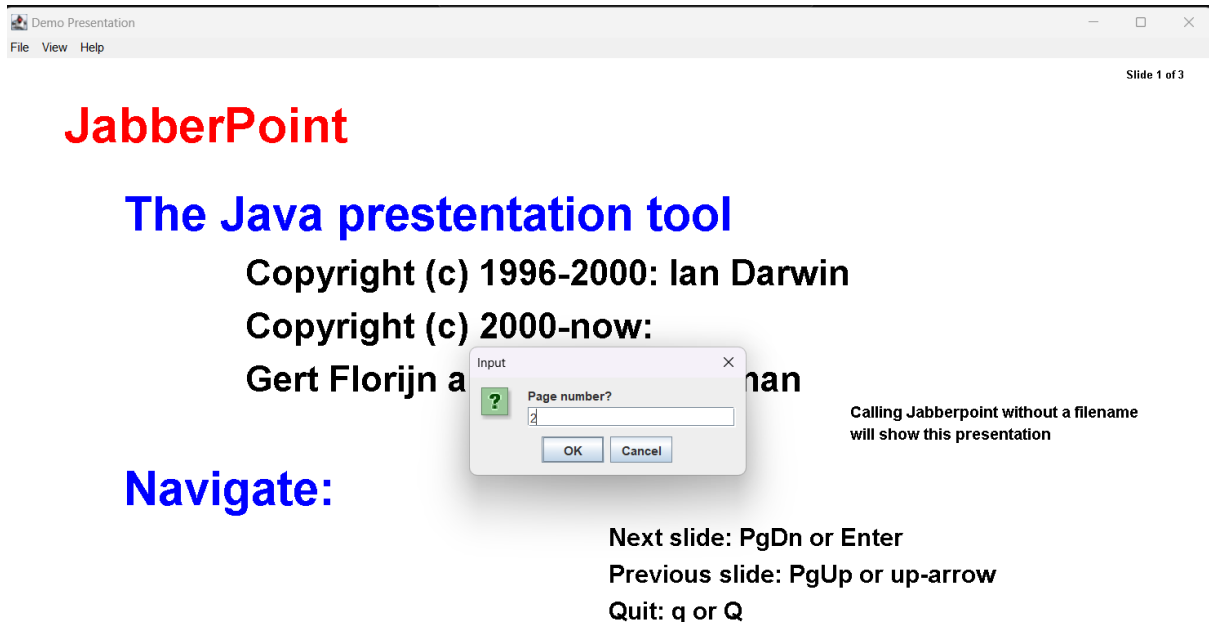
#### 4.4. Exit

When you click “Exit” button or press q or Q in your keyboard, the application will close.

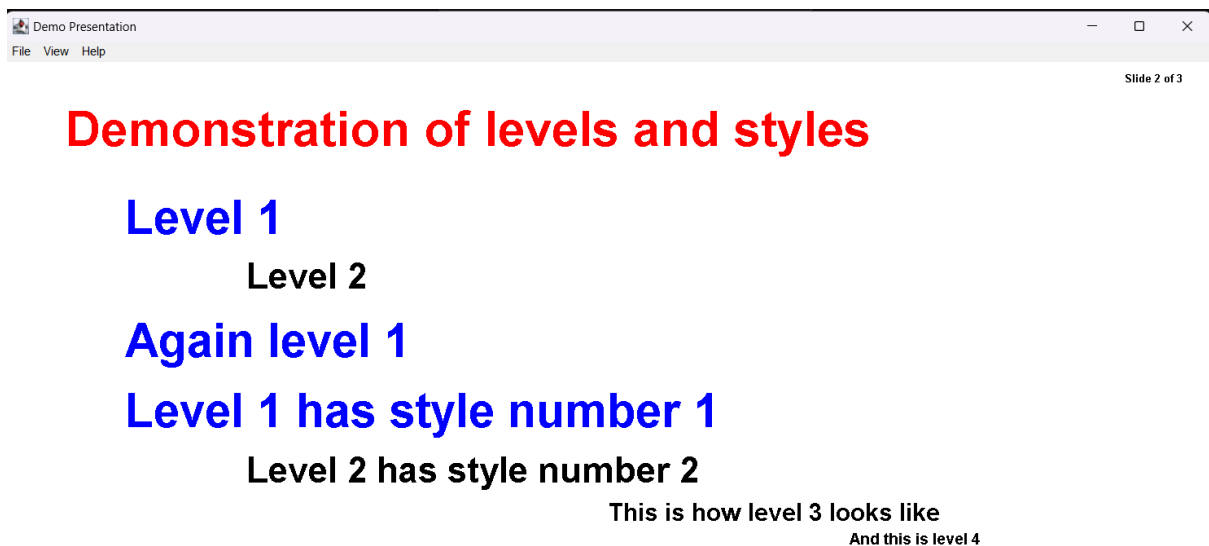
## 5. Navigation Bar

### 5.1. Go to

When you click go to, the app will navigate you to a slide you choose. For example, I want to go to slide 2.

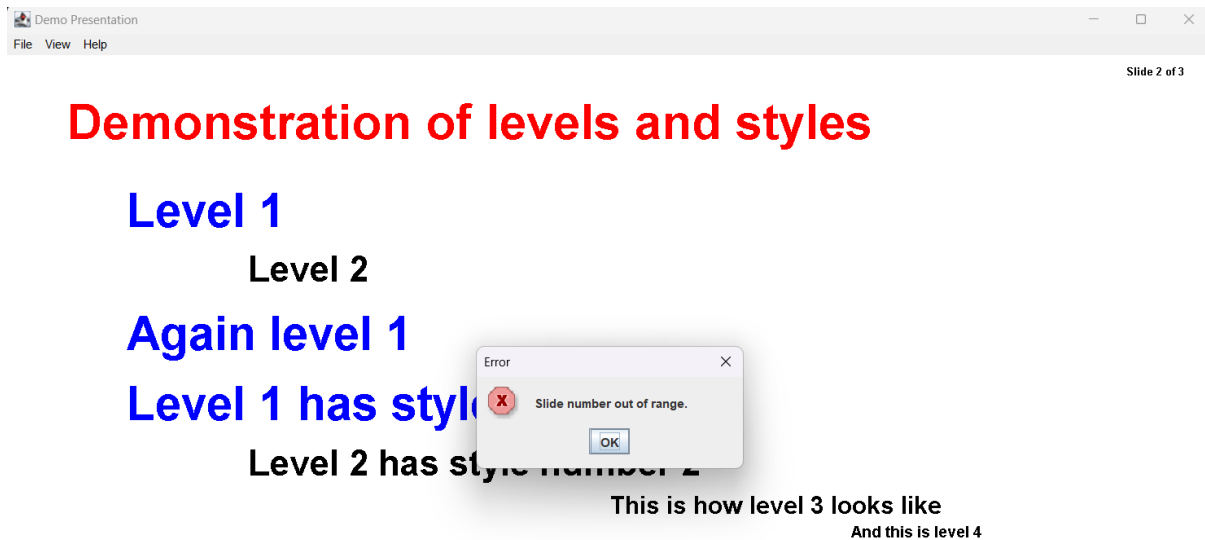


*Fig 5.1: Go to slide 2*



*Fig 5.2: Navigate to slide 2*

When you enter a slide number out of range it will pop up an error.



*Fig 5.3: Slide number out-of-range error*

## 5.2. Prev

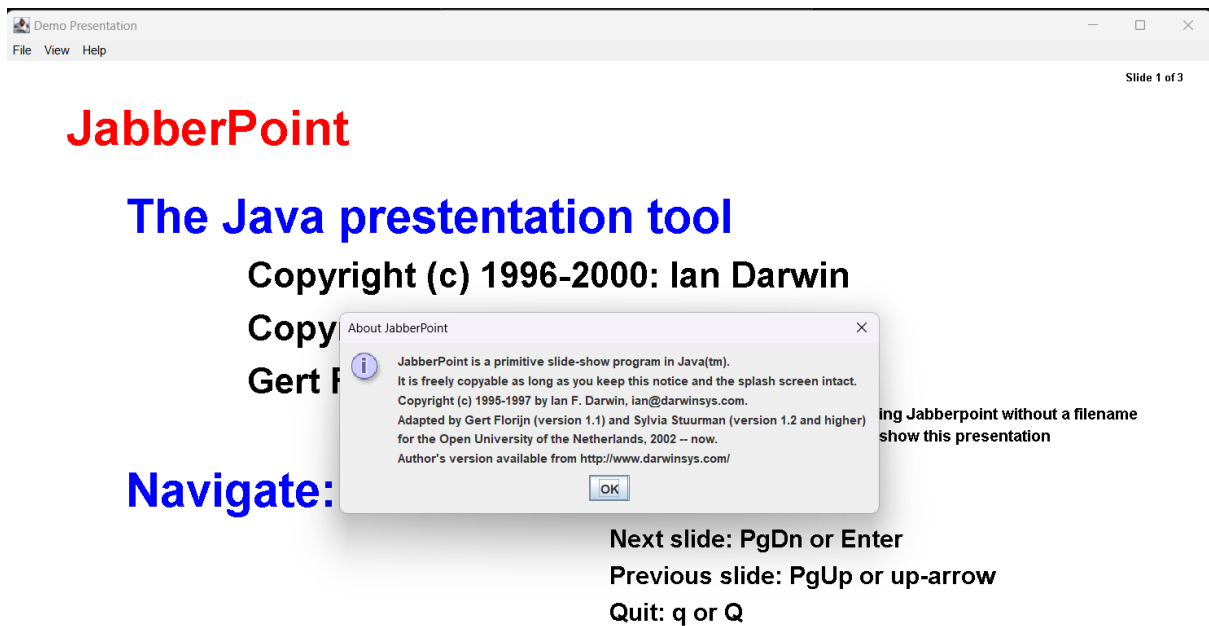
You can go to the previous slide by clicking “Prev” button or PgUp or up-arrow on your keyboard.

## 5.3. Next

You can go to the next slide by clicking “Next” button or PgDn or Enter on your keyboard.

## 6. About Application

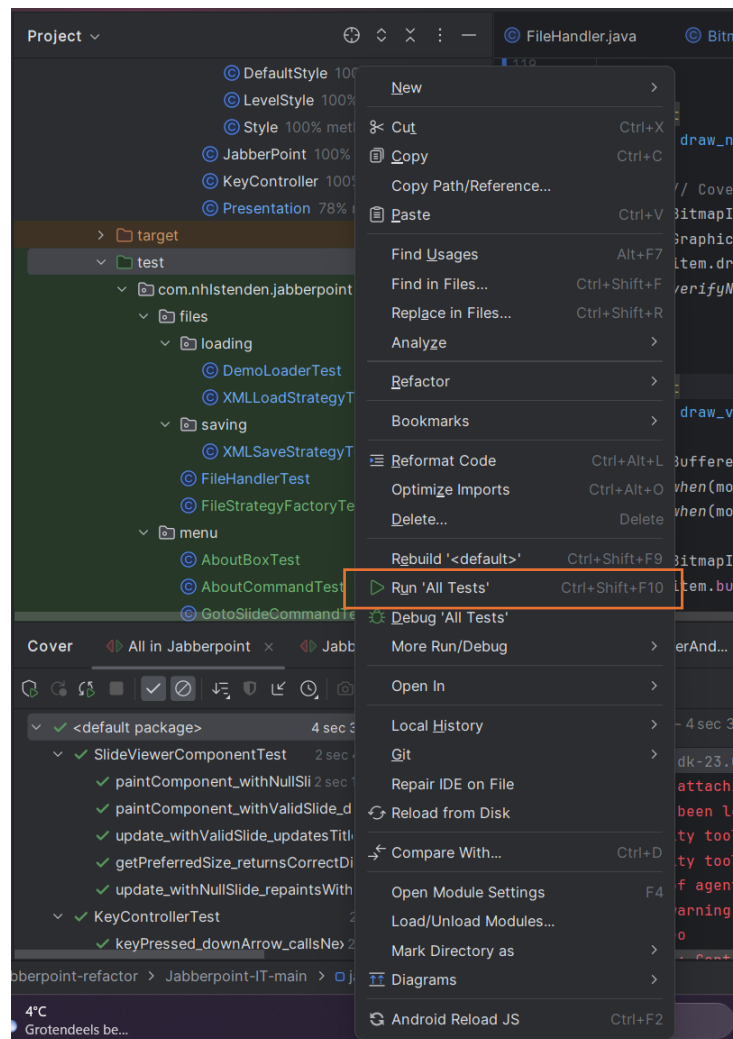
You can click to button “About” on the “Help” menu bar to open the about box.



*Fig 6.1: About box*

## 7. Testing

You can run all the tests by right-clicking on the test folder and choosing to run all the tests.



*Fig 7.1: How to run all the tests*

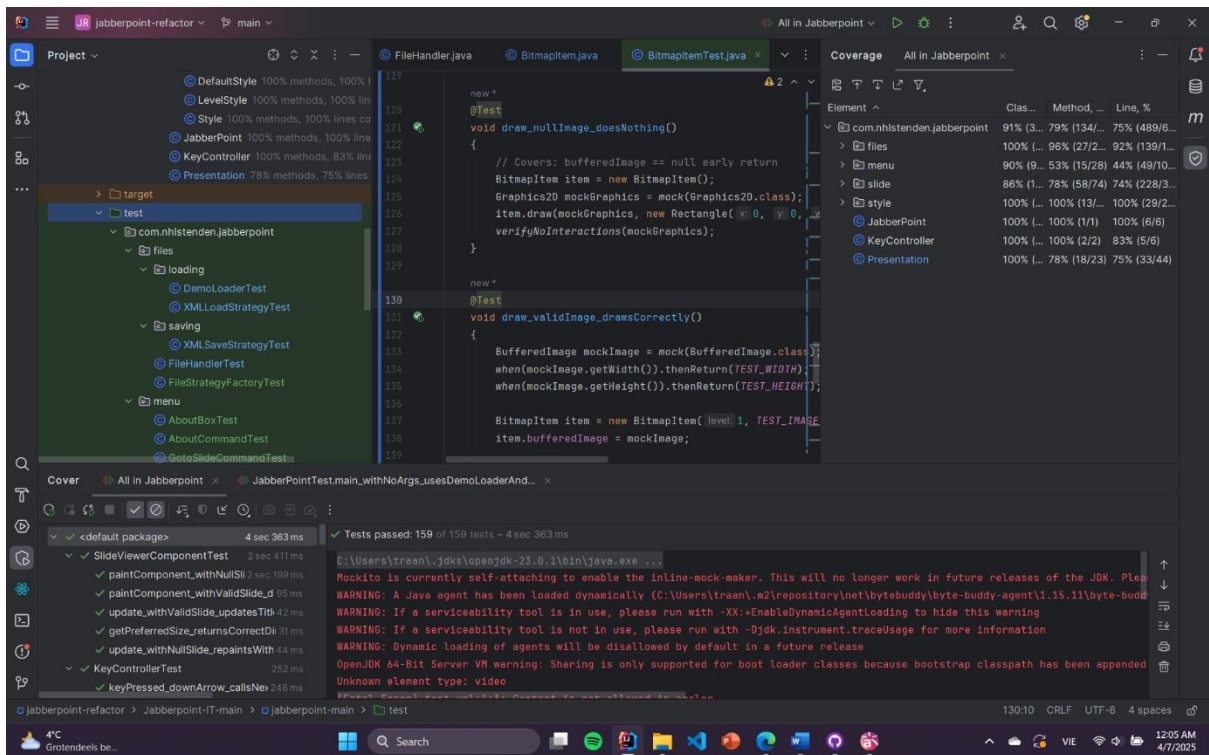


Fig 7.2: Run tests with coverage

To ensure that the application can run smoothly, please ensure that you do not run the application on IntelliJ version 2025 because this version contains several bugs that may cause complications errors or related errors while running. It's recommended to use the 2024 version.