Information

Lab Goals

In this lab, we will cover how to make a simple graphical application using the Swing toolkit in Java. We will be installing a visual drag and drop GUI creation tool called WindowBuilder into Eclipse, and use it to make an app.

The goals for this lab are:

- To understand what graphical applications are, and how they are driven
- To get WindowBuilder installed
- How to use WindowBuilder
- Discuss how Swing works, and things like Widgets and Layouts
- How to use events to make things happen
- Make a simple graphical application

Information

Graphical Applications

Up until now, all you have done is make simple command line applications that read in data from the console, perform a calculation, and then print the result to the console. These applications can be pretty powerful, but the world now basically requires any serious application to have some sort of graphical front end. The world may be transitioning to web apps and applications with HTML / JS front ends, but rest assured, desktop applications are not going anywhere anytime soon.

Building graphical applications requires you to think completely differently than how you normally think when you approach application programming. This is the reason why students think graphical applications are hard to build, and then complain two days before the assignment is due that they don't know how to start or things like that. The key to graphical applications is to start early, and have a clear vision of what you want to implement.

A clear vision is very important. Normally, when you are given a set of requirements for a command line application, you can happily write Java code from the start to finish, by breaking the problem down into stages, and then starting at the beginning, writing a stage, testing it, and then moving onto the next stage. Command line applications are also mostly linear which reduces complexity of what you have to write. In graphical applications this isn't always the case. Graphical applications might need to immediately display information and offer the user different actions that they can perform. They normally aren't linear at all, and have to maintain "state" that is always consistent and can be shown to the user.

It is far easier to plan out what applications will look like, how they will feel to use, and their underlying design before you start to code. So it becomes important to have a vision of what you want to implement before you start.

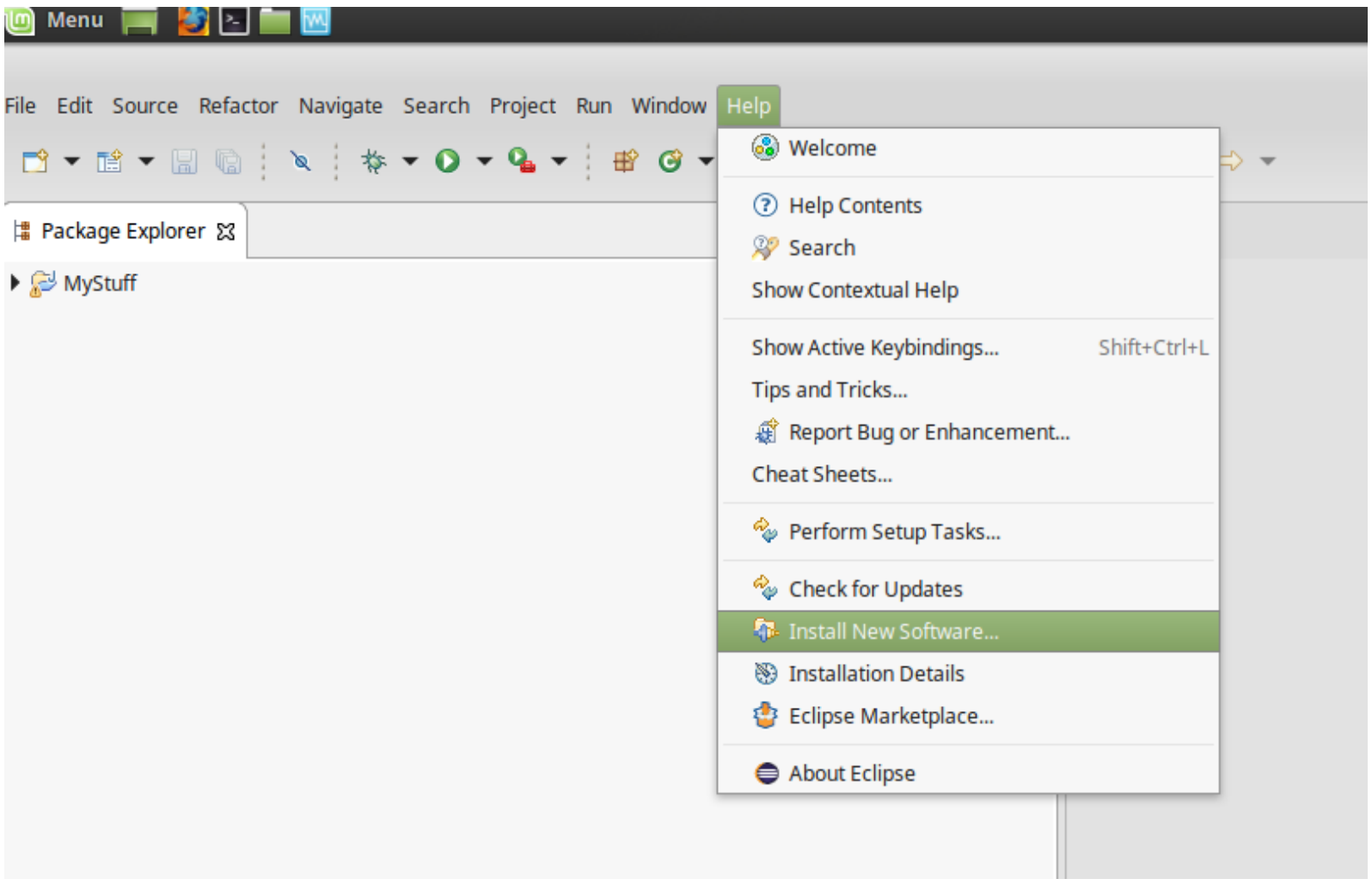
Information

Getting Started

Lets get WindowBuilder installed. WindowBuilder is a plug-in for Eclipse, and it enables you to drag and drop graphical widgets onto a window, and WindowBuilder generates valid Swing code that represents what you see on your screen. Its a fantastic tool, and it can really help speed up development. Saying that, you don't have to use WindowBuilder. You can write Swing code by hand, but it will take a very long time.

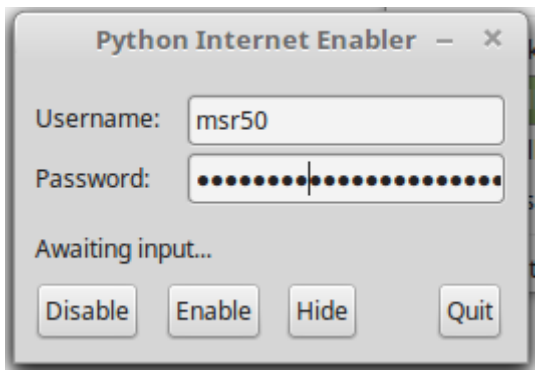
Step One:

Launch Eclipse. Click Help > Install New Software



Step Two:

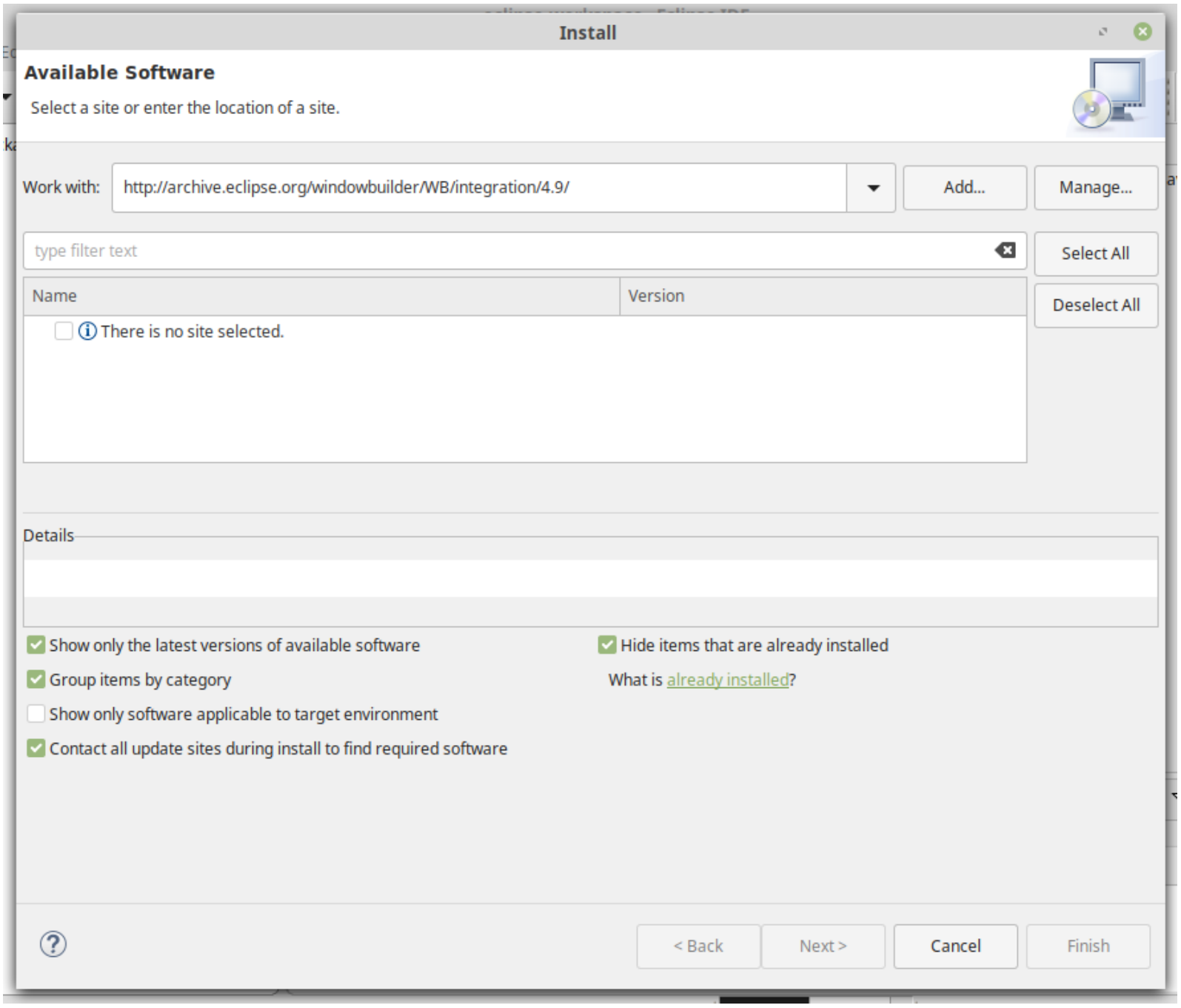
You need to make sure that your computer has been "internet enabled". This is to make sure your computer can reach the outside internet through the firewall. Open the Linux Mint menu, and search for "Python Internet Enabler". Enter your details and press enable.



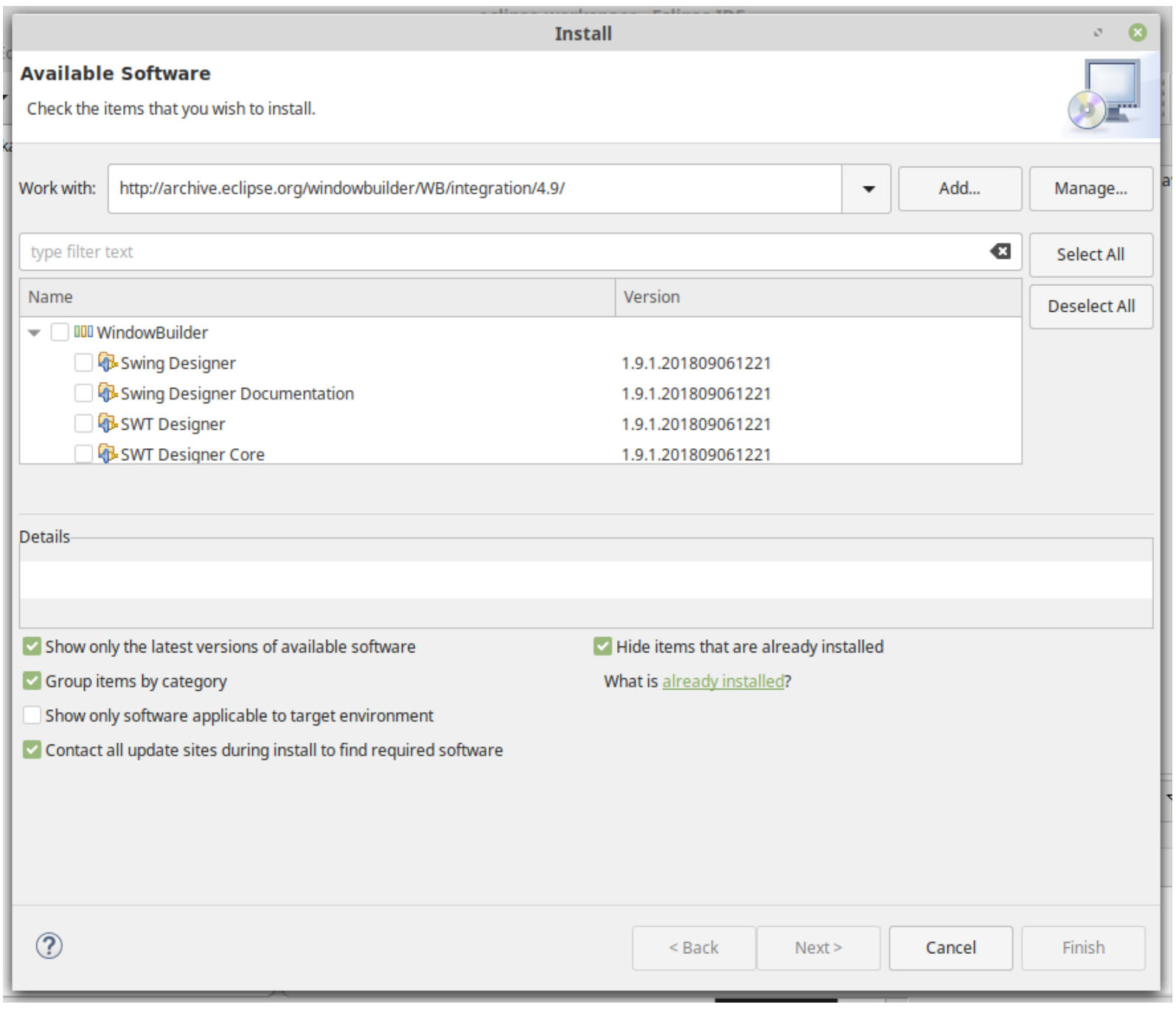
Now, in the Eclipse Window to install software, enter the following URL in the software sources box:

http://archive.eclipse.org/windowbuilder/WB/integration/4.9/

Note, the 4.9 is the version of Eclipse you have. Look at Help > About Eclipse to check your version.



Press the Add... button after you have pasted the Link in, and press Add to fetch the software list from that URL (you can leave the Name field blank). Your window should now look like:



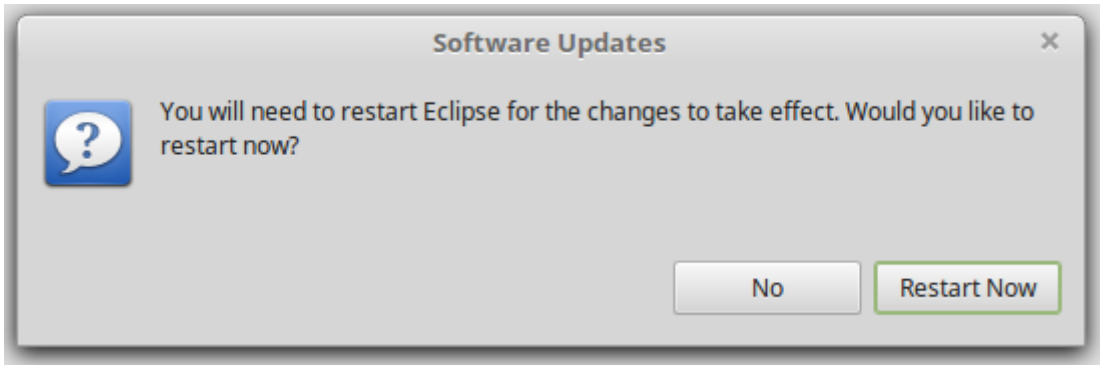
Tick the top WindowBuilder box to install all components. Press Next.

Press Next again.

Accept the license conditions and press Finish.

Eclipse will then go and install WindowBuilder in the background, with no real progress bar. How frustrating right? You can use the lower right corner to get an idea on progress.

When you get a prompt to restart Eclipse, the install is done. Restart Eclipse.



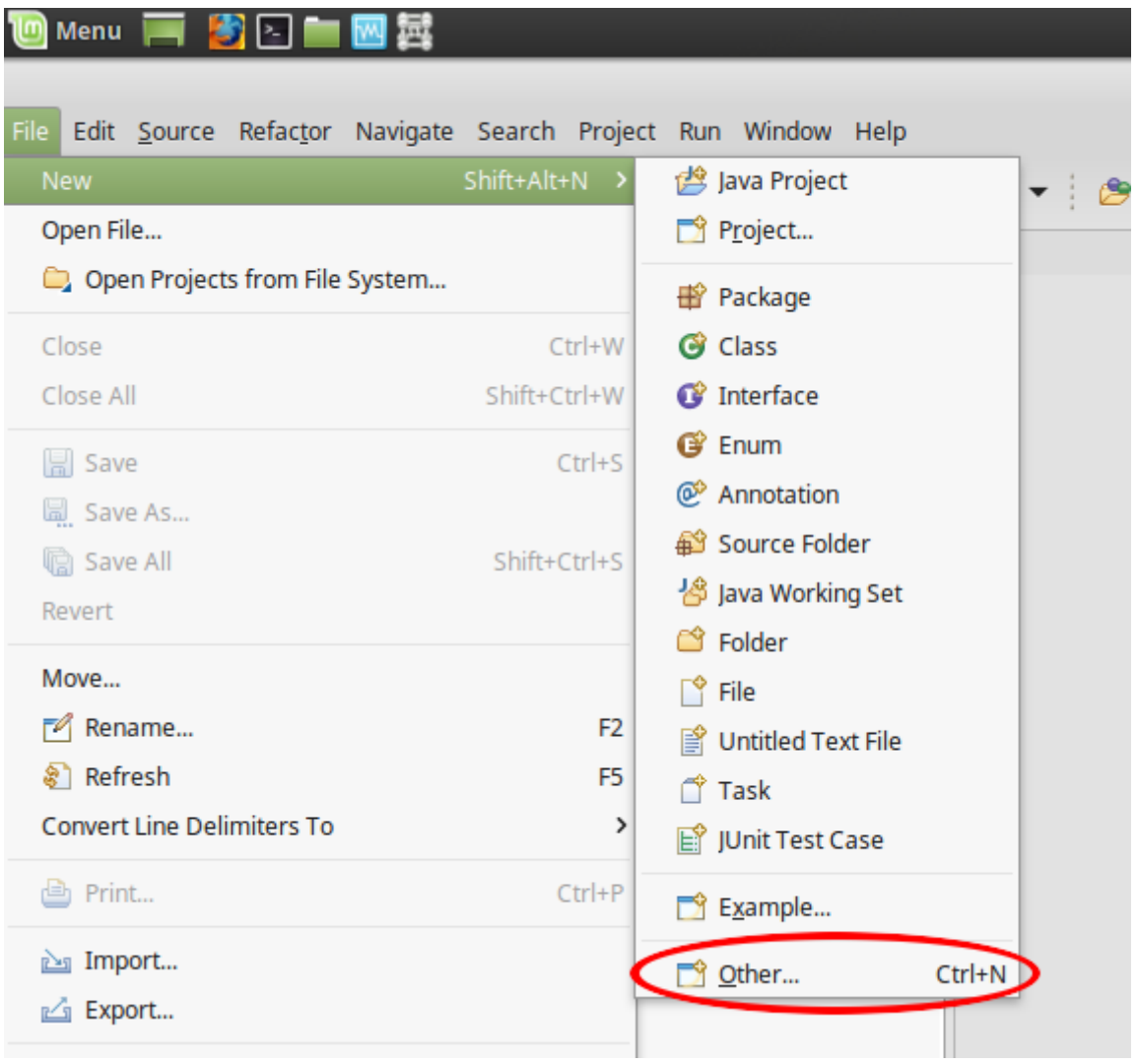
We are done.

Information

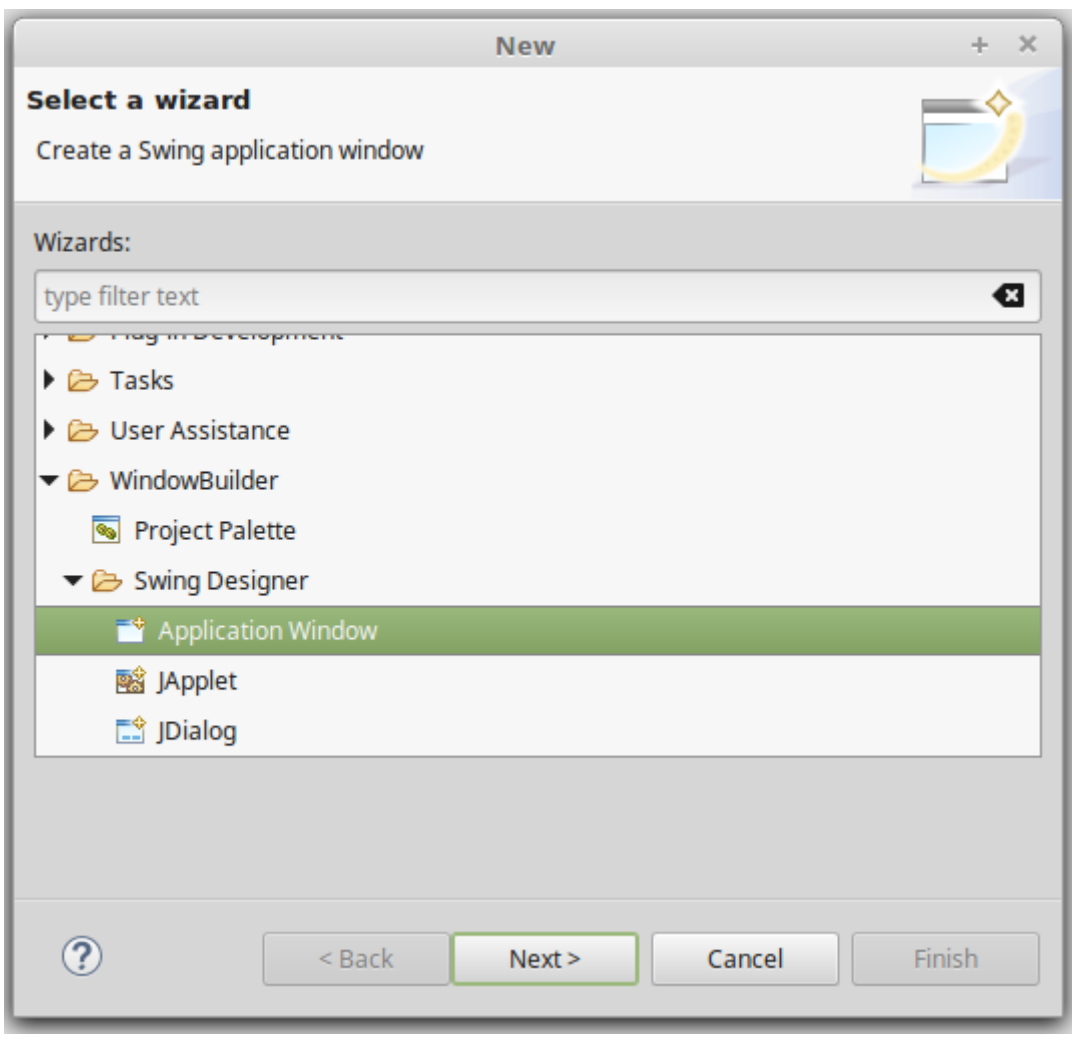
Lets create a new WindowBuilder Project:

Make a new project like you normally do, with File > New > Java Project

Once that has been done, press File > New > Other



Then Search for WindowBuilder in the window that appears, open it up and select "Swing Designer" and "Application Window"



Press Next. Name your App.

I'm going to name mine "StarmansCoolApp".

Once you press Finish, you will see something like this:

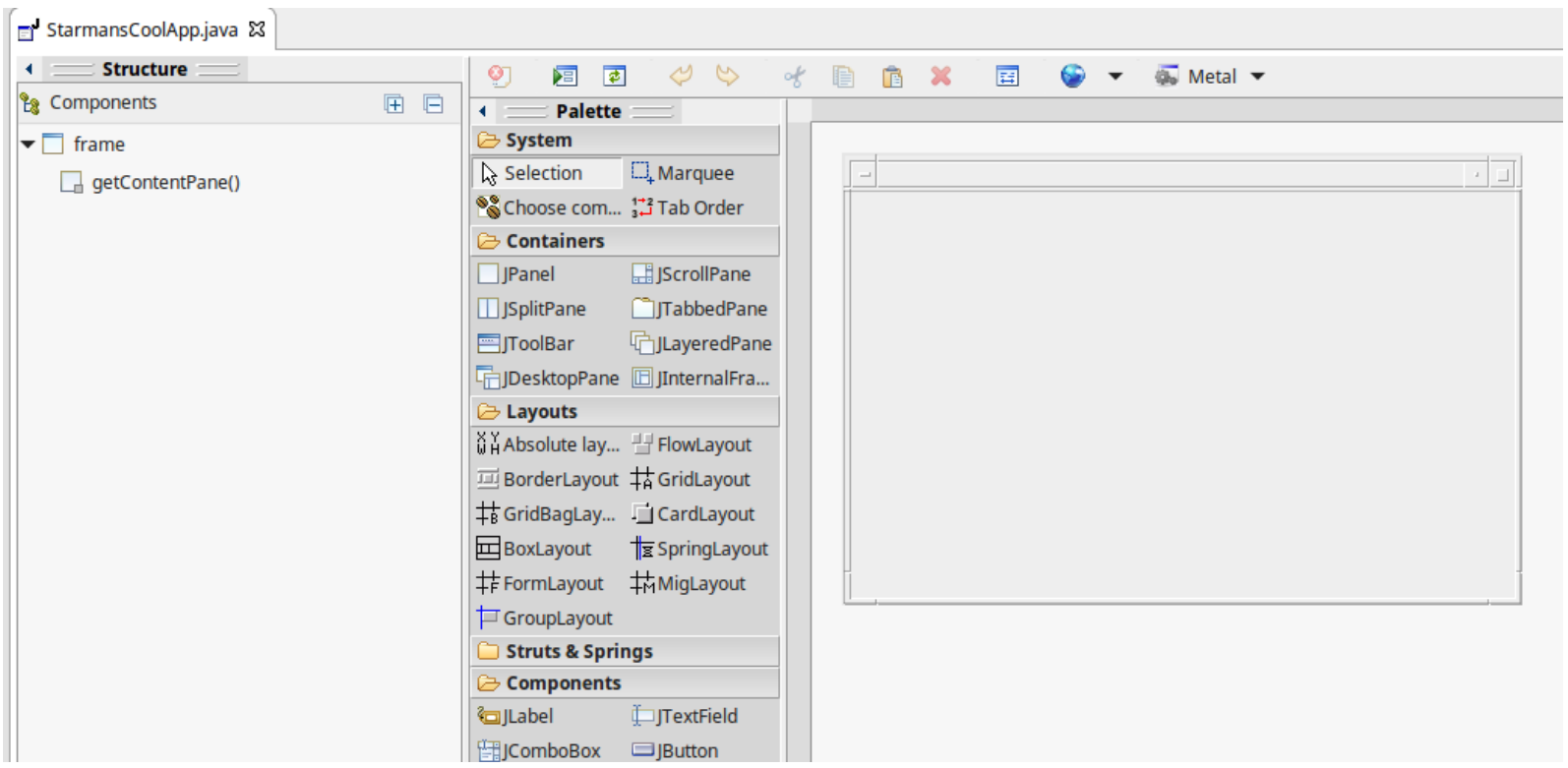
StarmansCoolApp.java

```
1  import java.awt.EventQueue;
4
5  public class StarmansCoolApp {
6
7      private JFrame frame;
8
9      /**
10      * Launch the application.
11      */
12     public static void main(String[] args) {
13         EventQueue.invokeLater(new Runnable() {
14             public void run() {
15                 try {
16                     StarmansCoolApp window = new StarmansCoolApp();
17                     window.frame.setVisible(true);
18                 } catch (Exception e) {
19                     e.printStackTrace();
20                 }
21             }
22         });
23     }
24
25     /**
26      * Create the application.
27      */
28     public StarmansCoolApp() {
29         initialize();
30     }
31
32     /**
33      * Initialize the contents of the frame.
34      */
35     private void initialize() {
36         frame = new JFrame();
37         frame.setBounds(100, 100, 450, 300);
38         frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
39     }
40
41 }
42
```

Note: If you have red error lines throughout your code, make sure the module-info file does NOT exist. If it does just delete it. This file adds needless configuration that we don't need.

And you will see a tab at the bottom of the window. It will say "Source" and "Design". Source shows you your source code. "Design" opens up the graphical drag and drop plugin. Click Design.

You will see something like this:



Thats it. Lets start talking about how we make a graphical application.

◄ Quiz 5 - JUnit Testing and Exceptions Jump to... Quiz 6 - GUI Development (OLD) ►