



Dialog Window

- A Dialog window is an independent subwindow meant to carry temporary notice
- Most Dialogs present an error message or warning to a user, but Dialogs can present images, directory trees, or just about anything



Packages

- The Java Application Programming Interface (API) contains hundreds of predefined classes that you can use in your programs.
- These classes are organized into what we call packages
 - **Packages** contain classes that have related purpose



JOptionPane Class

- A way to get input from the user is by using the JOptionPane class which is found in the javax.swing package
- JOptionPane makes it easy to pop up a standard dialog box that prompts users for a value or informs them of something



Importing the class and or entire package

- The statement

- `import javax.swing.JOptionPane;`

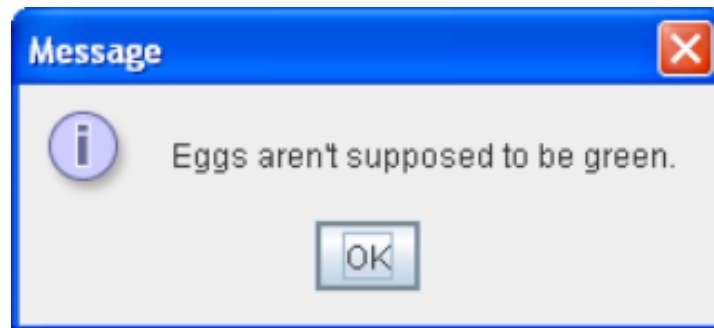
- indicates that we want to import the class `JOptionPane` from the `javax.swing` package.

- Or the entire swing package written as

- `import javax.swing.*;`

For example:an informational dialog:

- `JOptionPane.showMessageDialog(null, "Eggs are not supposed to be green.");`





JOptionPane Dialogs

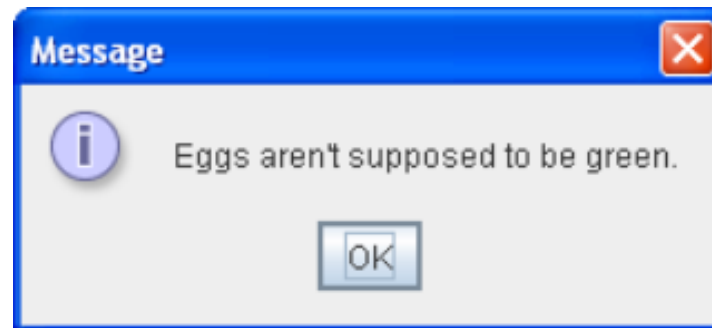
- **Icons used by JOptionPane (Java look and feel)**

- Question ?
- Information i
- Warning !
- Error x

Default Dialog

- //default title and icon

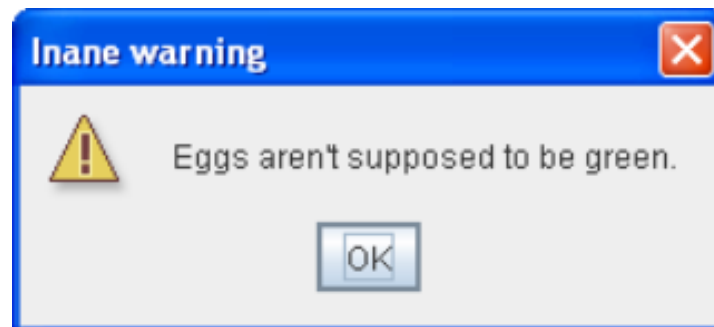
```
JOptionPane.showMessageDialog(null, "Eggs are  
not supposed to be green.");
```



Warning Dialog

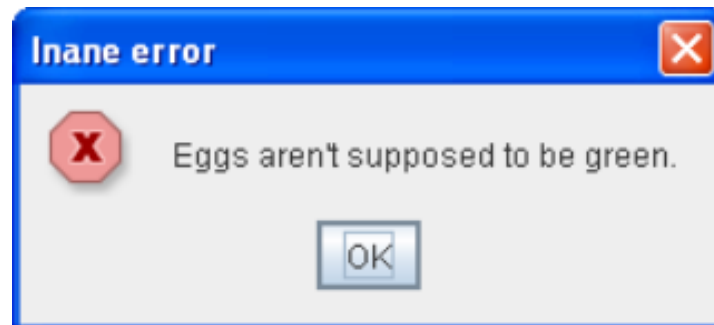
- //custom title, warning icon

```
JOptionPane.showMessageDialog(null , "Eggs are not  
supposed to be green " , "Inane Warning",  
JOptionPane.WARNING_MESSAGE);
```



Error Icon

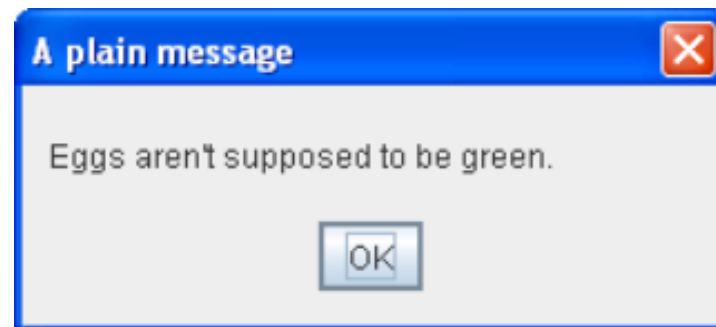
- //custom title, error icon JOptionPane.showMessageDialog(null, "Eggs are not supposed to be green.", " Warning ", JOptionPane.ERROR_MESSAGE);



No Icon

- //custom title, no icon

```
JOptionPane.showMessageDialog(null, "Eggs are not supposed to be green.", "A plain message", JOptionPane.PLAIN_MESSAGE);
```





Getting input from the Keyboard

- `String name= JOptionPane.showInputDialog(null, "Please Enter Your Name");`
- `JOptionPane.showMessageDialog(null, name);`
- `System.exit(0);`

- The statement,

`String name = (String)JOptionPane.showInputDialog(null, "name");`
creates a `JOptionPane` input dialog, which will display a dialog with a message, a textfield and an OK button

- This returns a `String` which we will save in the `name` variable.



- The statement,

- String name;
- String msg;
- name = JOptionPane.showInputDialog(null, "name");
- msg = "hello" + name + "!";
- JOptionPane.showMessageDialog(null, msg);
- System.exit(0);

- creates the welcome message, which we will store in the msg variable.