

DEPENDENCY INJECTION – AN EXAMPLE

THE CONTROLLER CLASS CAN TREAT UI ELEMENTS IN THE FXML AS MEMBER VARIABLES

CONTROLLER JAVA CODE

```
public class Controller {  
  
    @FXML  
    private Button goButton;  
  
    @FXML  
    private WebView browser;  
  
    @FXML  
    private TextField urlTextField;  
  
    @FXML  
    public void goHandler(ActionEvent e) {  
        browser.getEngine().load(urlTextField.getText());  
    }  
}
```

FXML FILE

```
<TextField fx:id="urlTextField" text="http://doxydonkey.blogspot.in">  
    <opaqueInsets>  
        <Insets bottom="4.0" left="4.0" right="4.0" top="4.0" />  
    </opaqueInsets>  
</TextField>  
<Button id="goButton" fx:id="goButton" mnemonicParsing="false" onAction="#goHandler">  
    <opaqueInsets>  
        <Insets bottom="4.0" left="4.0" right="4.0" top="4.0" />  
    </opaqueInsets>  
</Button>  
<TextField fx:id="currentDisplayedUrlText" text="{urlTextField.text}">  
    <opaqueInsets>  
        <Insets bottom="8.0" left="4.0" right="4.0" top="10.0" />  
    </opaqueInsets>  
</TextField>  
</children>  
</fx:include>  
<WebView fx:id="browser" minHeight="200.0" minWidth="200.0" prefHeight="504.0" />
```

THE CONTROLLER CLASS CAN TREAT
UI ELEMENTS IN THE FXML AS MEMBER
VARIABLES

(BTW REMEMBER THAT THE FXML
ALSO HAS A LINK TO THE NAME
OF THE CONTROLLER CLASS, THAT'S
NOT IN THE SNIPPING HERE)

CONTROLLER JAVA CODE

```
public class Controller {  
    @FXML  
    private Button goButton;  
  
    @FXML  
    private WebView browser;  
  
    @FXML  
    private TextField urlTextField;  
  
    @FXML  
    public void goHandler(ActionEvent event) {  
        browser.getEngine().load(urlTextField.getText());  
    }  
}
```

FXML FILE

```
<TextField fx:id="urlTextField" text="http://doxydonkey.blogspot.in">  
    <opaqueInsets>  
        <Insets bottom="4.0" left="4.0" right="4.0" top="4.0" />  
    </opaqueInsets>  
</TextField>  
<Button id="goButton" fx:id="goButton" mnemonicParsing="false" onAction="#goHandler">  
    <opaqueInsets>  
        <Insets bottom="4.0" left="4.0" right="4.0" top="4.0" />  
    </opaqueInsets>  
</Button>  
<TextField fx:id="displayedUrlText" text="{urlTextField.text}">  
    <opaqueInsets>  
        <Insets bottom="8.0" left="4.0" right="4.0" top="10.0" />  
    </opaqueInsets>  
</TextField>  
</children>  
</fx:include fx:id="browser" minHeight="200.0" minWidth="200.0" prefHeight="504.0" />
```

THE LINK RELIES ON TWO BITS OF SYNTAX

THE @FXML ANNOTATION
THE FX:ID TAG

**AS YOU CAN SEE - AT HEART, FXML
IS A WAY TO SET UP MEMBER
VARIABLES OF COMPLICATED CLASSES
AT RUNTIME**

THIS IS EXACTLY WHAT THE

STRATEGY PATTERN

IS ALL ABOUT

AND IT IS ALSO A PRETTY CLASSIC EXAMPLE OF

DEPENDENCY INJECTION