CISC 3120

C15: JavaFX: Styling, FXML, and MVC

Hui Chen

Department of Computer & Information Science

CUNY Brooklyn College

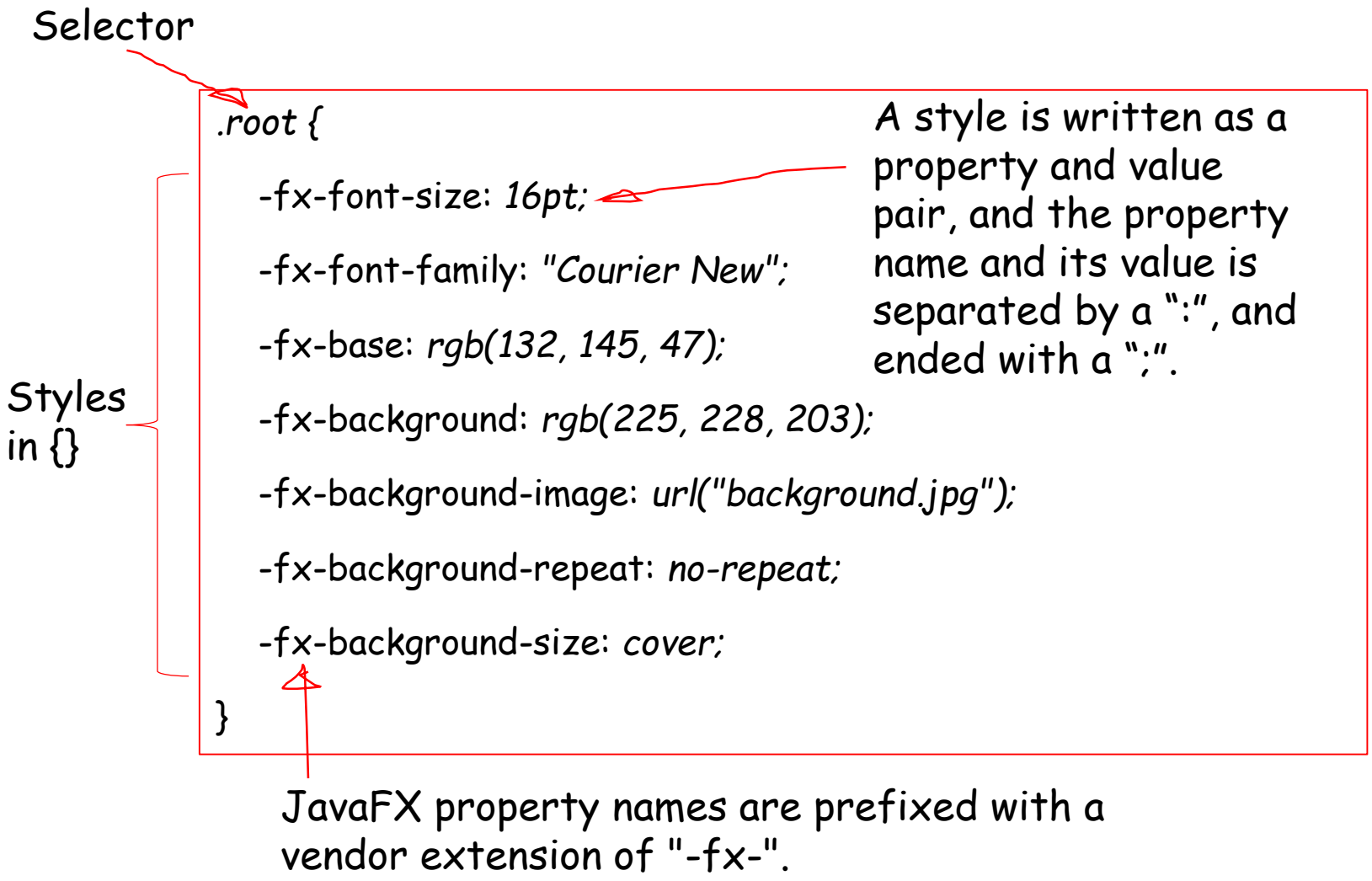
Outline

- Recap and issues
- Styling user interface with CSS
- FXML and Model-View-Controller pattern
- If time permits
  - Canvas, charts, and others
  - Interoperate with Swing
- Assignments
  - Project 3 (group activity)

JavaFX Cascading Style Sheets

- Control appearance of JavaFX interface using Cascading Style Sheets
- Cascading Style Sheets (CSS)
  - A World-Wide-Web Consortium (W3C) standard
  - Originally designed as a simple mechanism for adding style (e.g., fonts, colors, spacing) to Web documents
  - See https://www.w3.org/Style/CSS/
  - CSS level 1, 2, and 3 (some still under development)
- JavaFX CSS (JavaFX 8)
  - Based on W3C CSS level 2.1 with some addition on current work on CSS level 3
  - Aimed at providing a uniform method to style both desktop and web applications

An Example of JavaFX CSS



Apply Styles

- Styles are applied (but not necessarily selected for) to Nodes in the Scene-graph
  - First applied to the parent, then to its children
- A node is styled after it is added to the scene graph.
- A node is re-styled
  - when the following changes made to the node's pseudo-class state, style-class, id, inline style, or parent
    - Pseudo-class state: e.g., MouseEvent.MOUSE_ENTERED
  - When stylesheets are added to or removed from the scene.

CSS Selectors

- CSS selectors are used to match styles to scene-graph nodes
  - Type selector
  - Class selector
  - ID selector

Type Selector

- Select based on type name returned by Node's getTypeSelector method
- Analogous to a CSS type selector
- See style and code example in
  - JavaFXCssStyledCsQuoteApp

Class Selector

- Select based on the value of the styleClass property of the Node
  - A Node can have multiple style classes
- Analogous to a CSS class selector
- See style and code example in
  - JavaFXCssStyledCsQuoteApp

ID Selector

- Select based on the ID of the Node
  - The ID of a Node can be set using Node's setId method
  - ID is should be unique
- Analogous to a CSS ID selector
- See style and code example in
  - JavaFXCssStyledCsQuoteApp

Context Selector

- Selection based on contextual information
- Example:
  - #brooklyn-orange-next-quote Text { … }
  - matches a Node whose type name is "Text" and the Node is a descendent of the Node whose ID is #brooklyn-orange-next-quote
  - See CSS 3 Selectors for more
    - https://www.w3.org/TR/css3-selectors/

Swing & JavaFX

- Swing is a successful toolkit for more than a decade
- Why JavaFX
  - To provide applications with such sophisticated GUI features
    - Smooth animation, web views, audio and video playback
  - Styles based on Cascading Style Sheets (CSS)

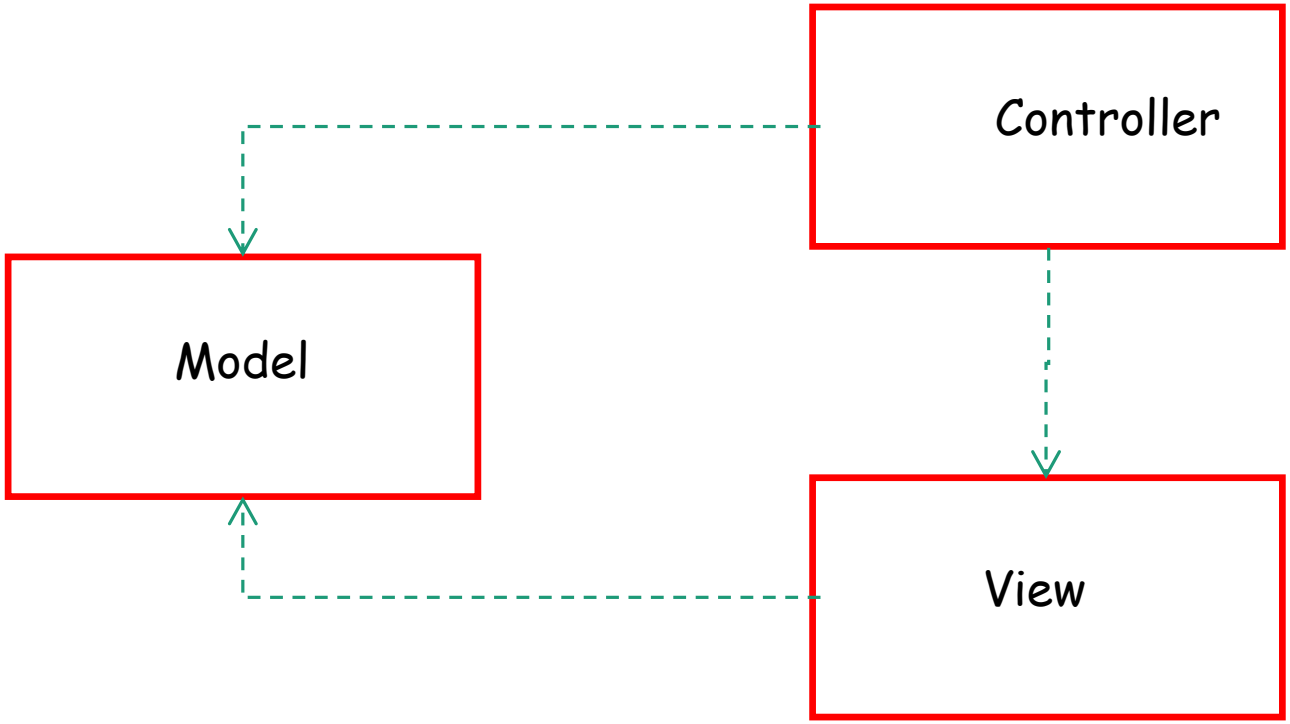
User Interface Design

- Thus far, the interface design is tied to the code.
- Is there any other way to do it?

User Interface Design with FXML

- FXML
  - XML-based language
  - XML = Extensible Markup Language
- Help build a user interface separated from the application logic

Model-View-Controller



- Key separation of concerns: view and controller depend on model, but model depends on neither.

Model-View-Controller

- It separates the three,
  - (Model) the modeling of the domain
  - (View) the presentation,
  - (Controller) and the actions based on user input into three separate classes
- A fundamental design pattern for the separation of user interface logic from business logic.

Model

- Manages the behavior and data of the application domain
- Responds to requests for information about its state (usually from the view)
- Responds to instructions to change state (usually from the controller).

View

- Manages the display of information.

Controller

- Interprets the mouse and keyboard inputs from the user
- Inform the model and/or the view to change as appropriate.

Computer Quote App

- Model (or Domain)
  - Computer science authors and what they said
- View
  - The interface shows the quotes
- Controller
  - Intercept users' mouse clicks
  - Inform model (or domain) about quote to display
  - Inform view to update the quote to be displayed

Use Eclipse for JavaFX FXML Project

- If from scratch
- Download and install JavaFX Scene Builder 2.0
  - Oracle does not offer the binary any more
  - Source code is distributed with the OpenJFX project
  - Download & install from a 3rd party provider
- Create a Maven project
- Create Controller class (always name it as a Controller)
  - Use @FXML to annotate fields and methods
- Create FXML file
  - You can open & edit it using the Scene Builder 2.0
  - Set handler

Example: JavaFXFXMLCsQuoteApp

- Define the Model
  - ComputerScienceQuoteDataModel.java
- Define the View
  - fxml_mainscene.fxml
- Define the Controller
  - FXMLGridViewController.java

Example: JavaFXFXMLCsQuoteApp

- Entry Point of the Application

```
private final static String APP_TITLE = "Quotations in Computer Science";
  private final static String MAIN_SCENE_FXML = "fxml_mainscene.fxml";
  @Override
  public void start(Stage primaryStage) throws IOException {
     Pane mainPane =
          (Pane)FXMLLoader.load(getClass().getResource(MAIN_SCENE_FXML));
     Scene mainScene = new Scene(mainPane);
      primaryStage.setTitle(APP_TITLE);
     primaryStage.setScene(mainScene);
     primaryStage.show();
  }
```

JavaFX & Swing Interoperability

- Swing applications can use JavaFX
- JavaFX applications can use Swing

Grow your skills & knowledge

- CISC 3620 Computer Graphics
  - 2D and 3D graphics
- CISC 3320 Operating Systems
  - Concurrency, processes, and threads

Assignments

- Practice
- Project 3