

Programmeerimise põhikursus Javas

Loeng 9

<http://courses.cs.ttu.ee/pages/ITI0011>

Outline

- Homework stuff
- codingbat
- **GUI, JavaFX continue**
- III HW
- I HW example

Homework submission

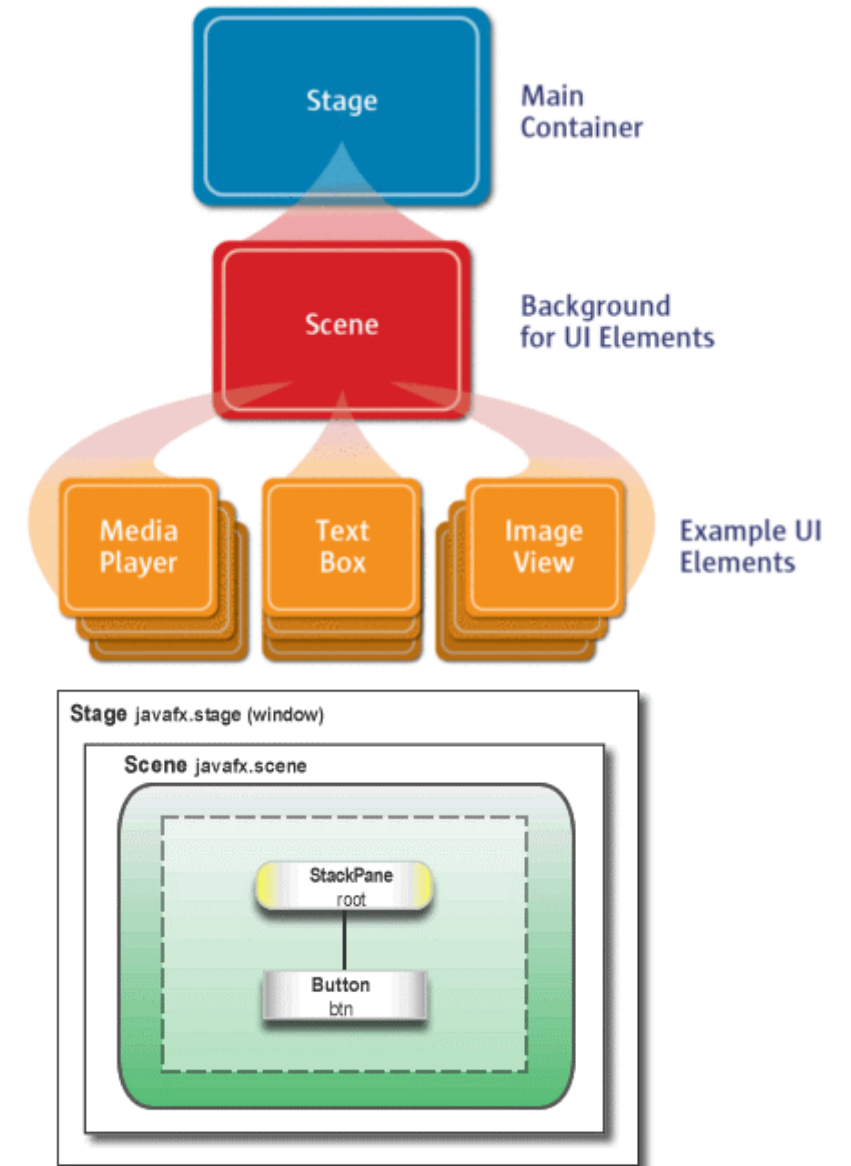
- <https://courses.cs.ttu.ee/pages/ITI0011:git>
- Homeworks into HW1, HW2, HW3 and HW4 folders
- **Check your score table to see git status**
- Homework 3 to be pushed into git latest **November 16th 23:59**
 - into folder "HW3"
 - **Android version**
- **Course code example in git:**
<http://firstname.lastname@git.ttu.ee/kursused/iti0011/materjalid.git>
- Use UNI-ID to access materials (not visible in browser)

JavaFX in Java 1.7

- <https://courses.cs.ttu.ee/pages/ITI0011:JavaFX>
- Add jfxrt.jar manually into build path
 - project properties
 - Java Build Path
 - Libraries
 - Add External JARs..
 - Locate jfxrt.jar (usually in lib folder in Java install folder: e.g. C:\Program Files\Java\jre1.8.0_20\lib\ext)
- Check "Hello world" application
- In case of problems, there are some links

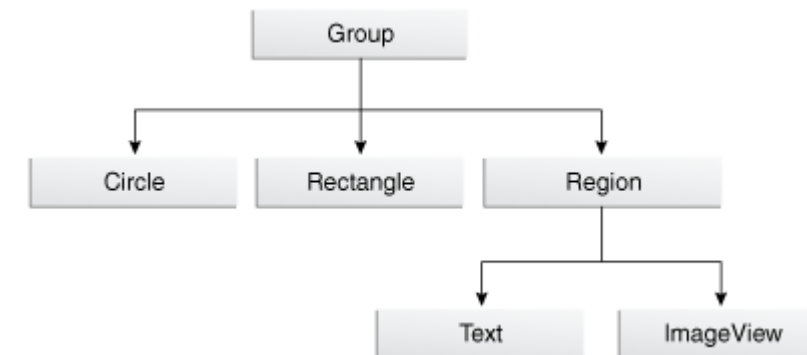
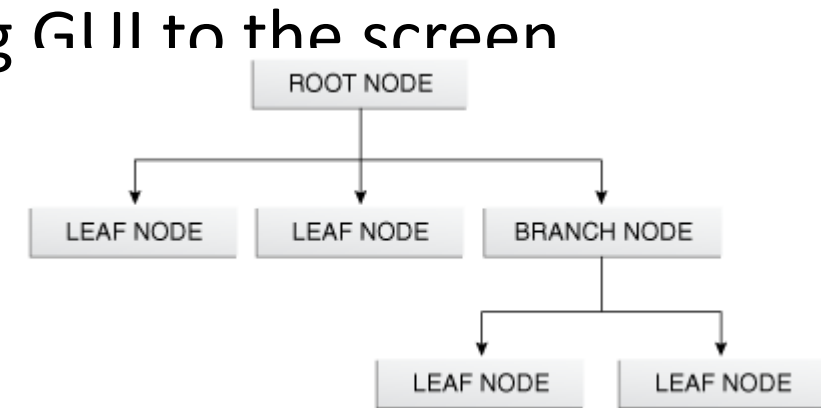
Structure of JavaFX application

- It's like a theatre play:
 - The Stage is the main container which is usually a Window with a border and the typical minimize, maximize and close buttons.
 - Inside the Stage you add a Scene which can, of course, be switched out by another Scene.
 - Inside the Scene the actual JavaFX nodes like AnchorPane, TextBox, etc. are added.



Everything is a node

- JavaFX Scene Graph API is responsible of rendering **GUI to the screen**
- A scene graph is a tree data structure
- Retained mode API - maintains internal model of all graphical objects
 - what objects to display
 - what areas of screen need repainting
 - how to render it all in the most efficient manner
- Instead of invoking drawing methods directly, instead use scene graph API
- Individual items held within the application are known as *nodes*



Observable List

- List elements can be observed
- When the list is changed, some actions can be taken
- JavaFX uses observable list for nodes
- When a node is changed, JavaFX knows about it
 - When a shape is changed, it can be redrawn
 - When a layout is changed, it can be rerendered etc.
- Most of the container elements have method **getChildren ()** which returns observable list of nodes: **ObservableList<Node>**
- This list can be used to add, get, set and remove elements