Course Introduction & GUI Part I

ISTE-121

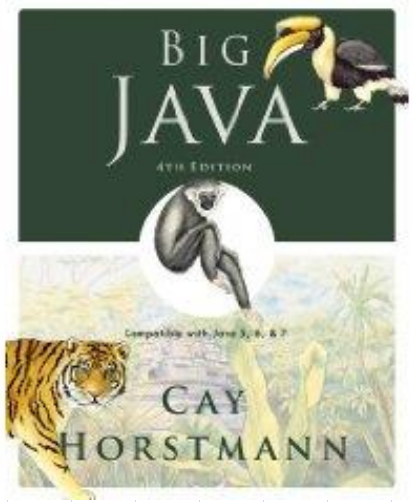
Introduction and Overview

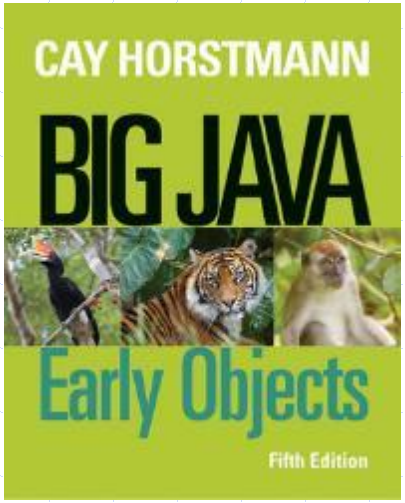
Instructor: Alan Mutka

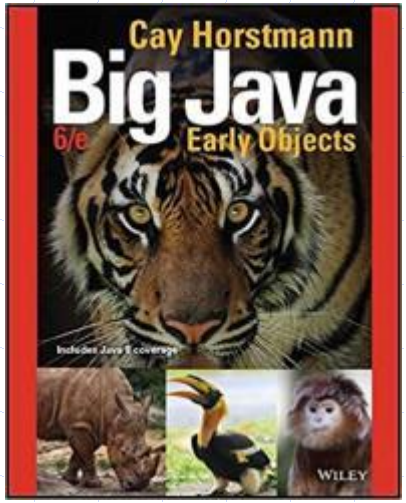
alan.mutka@croatia.rit.edu

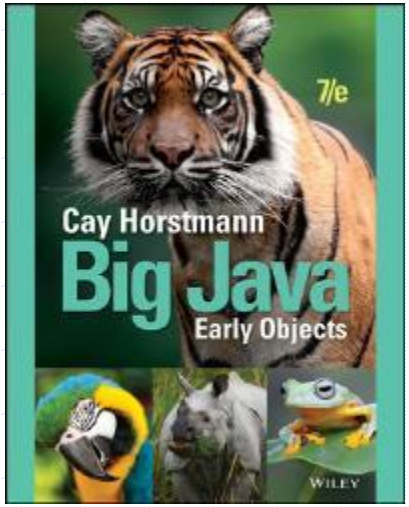
Text Reference

- The text for the course is "Big Java 6th Ed" by Cay Horstmann
- Both Text & Lectures are the primary source of information
- ISTE-120 & ISTE-121

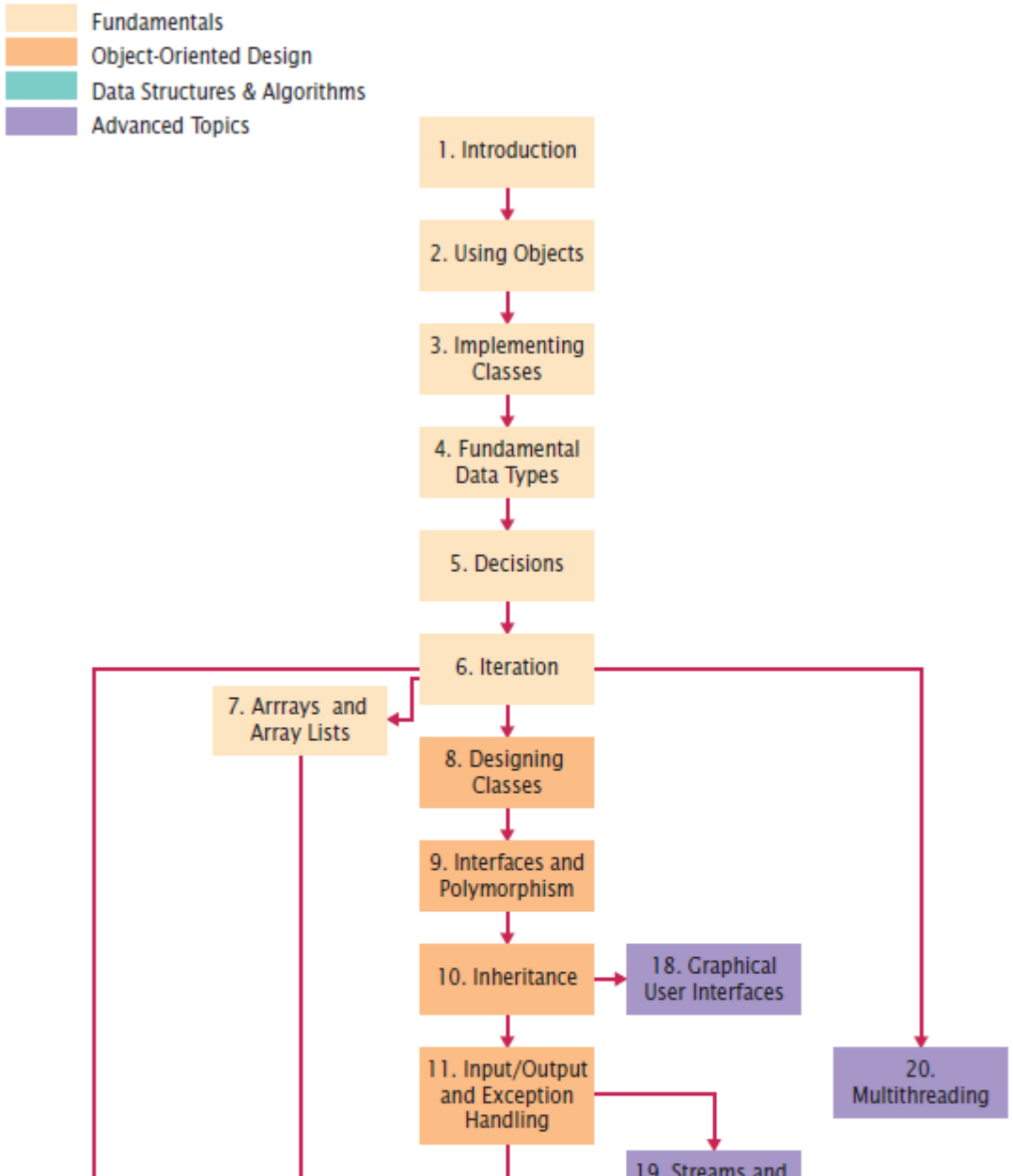






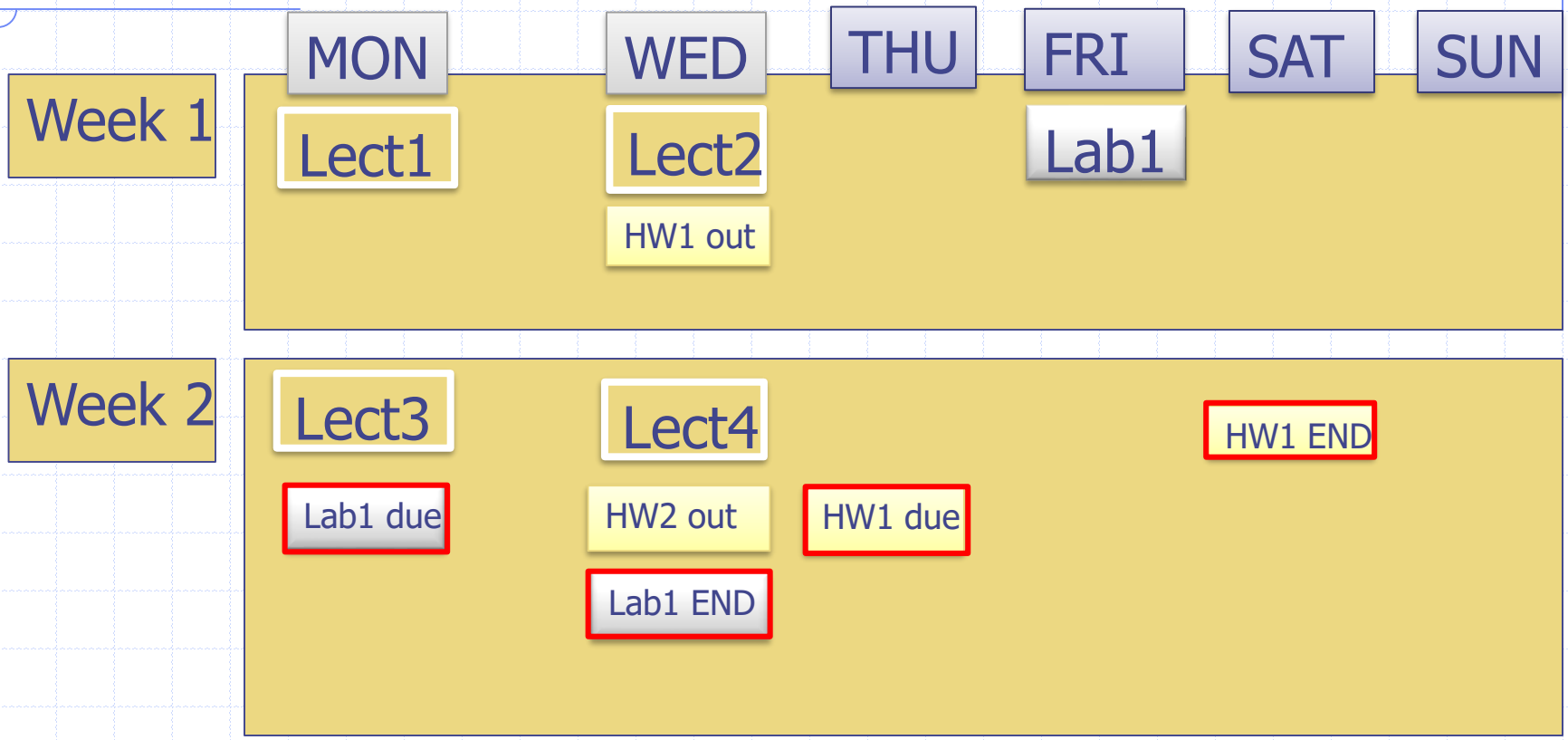


Big Java
Book chapters




R·I·T | myCourses

Syllabus 800&801



Grading

Grading

Grading scale (minimum percent): A = 94; A- = 90; B+ = 87; B = 83; B- = 80; C+ = 77; C = 73; C- = 70; D = 60

F = <60 or < 75% over-all exam average

Students are required to have at least a 75% average of the exams in order to pass the course.

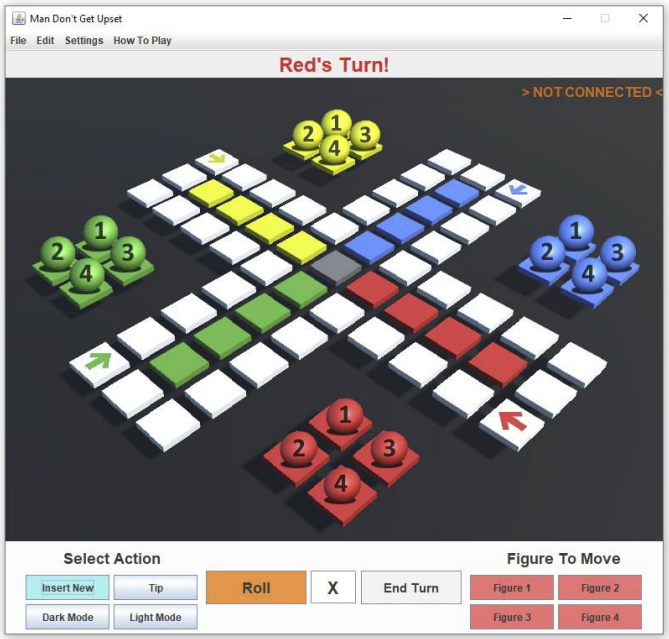
Components of evaluation:

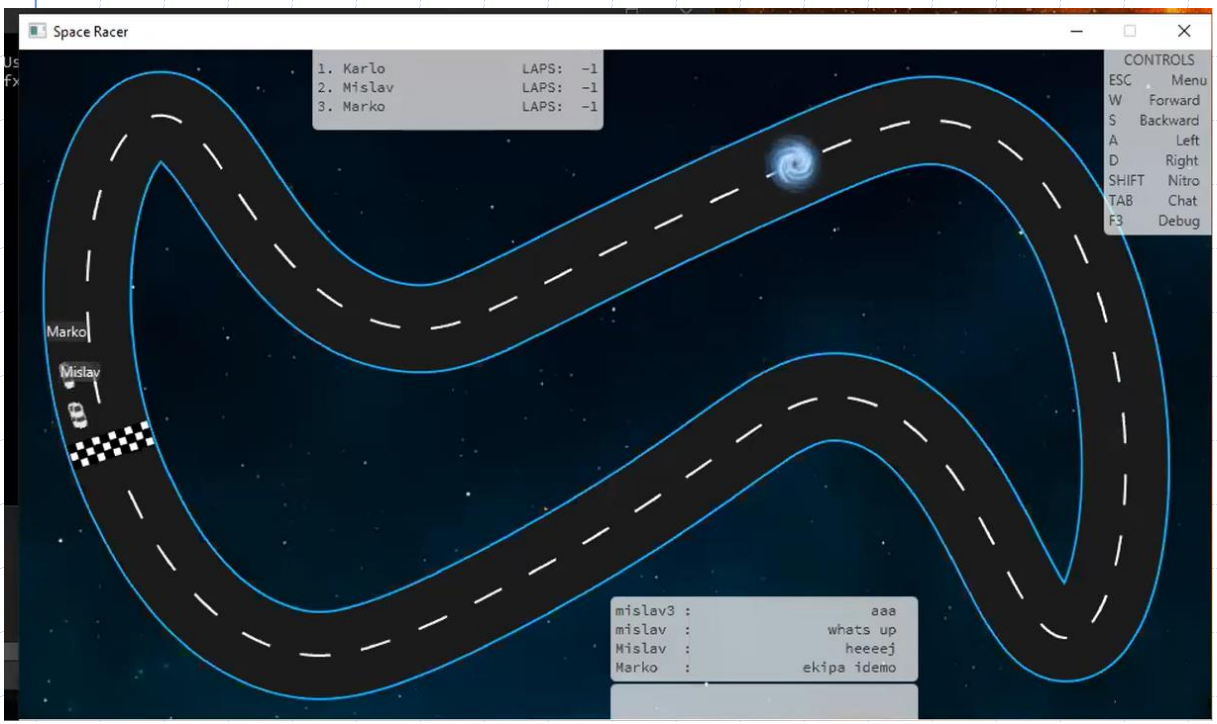
Component	Points/%
Practical 1	8
Practical 2	15
Practical 3	10
QUI	6
Laboratory	19
Homework	17
Project	25
Total:	**100**

Course learning outcomes based grading table:

	P1	P2	P3	QUI	LAB	HW	PR	ECTS	Points
ECTS	0.48	0.9	0.6	0.36	1.14	102	1.5		
Points	**8**	**15**	**10**	**6**	**19**	**17**	**25**		
CLO1				1	4	2	3	0.6	10
CLO2	5			1	3	2	1	0.72	12
CLO3	3			1	4	8	6	1.32	22
CLO4		15		1	2	5	9	1.92	32
CLO5			10	1	6		2	1.14	19
CLO6				1			4	0.1	5

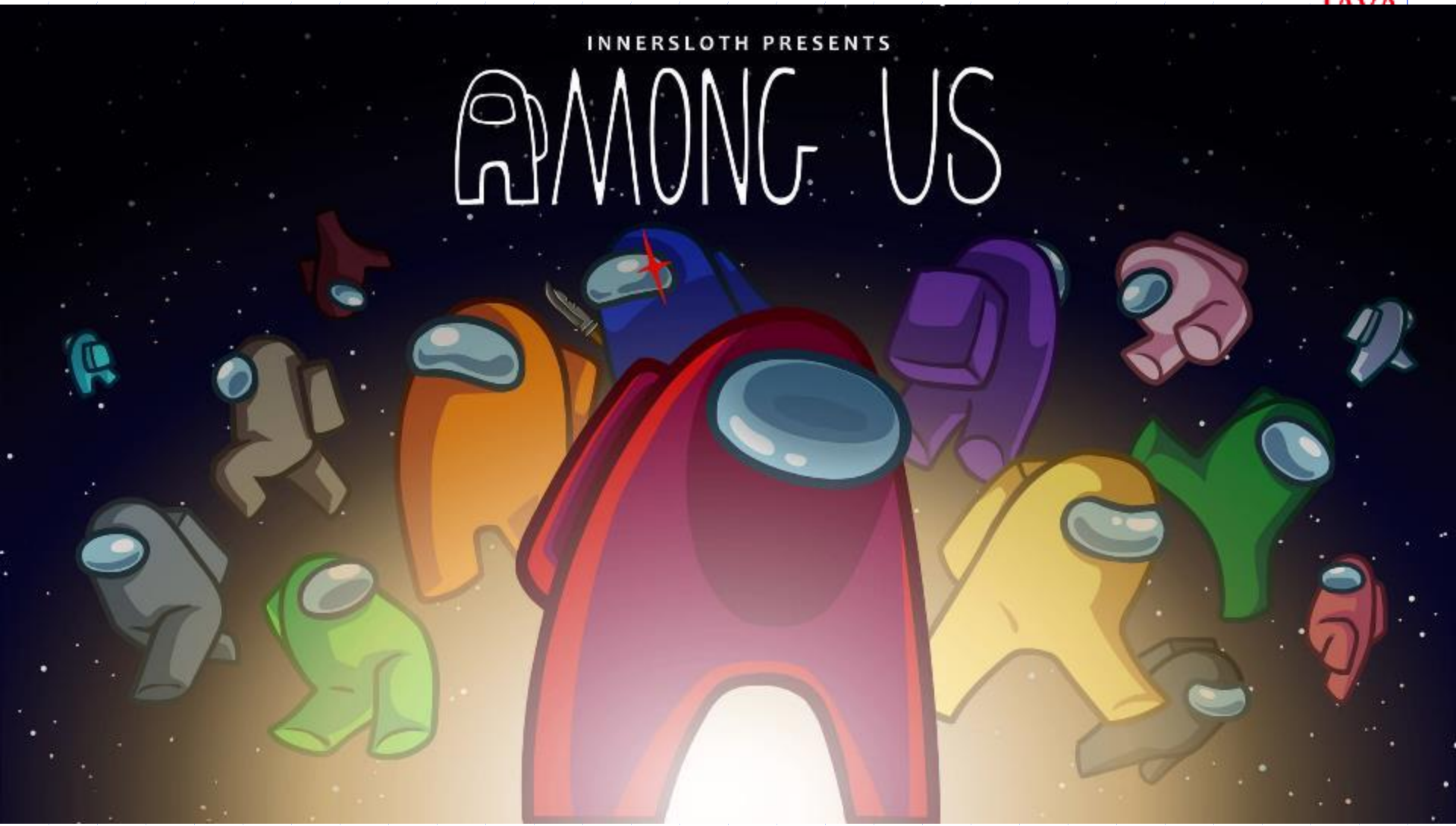
Project





- GUI Layout managers
- Events
- TextAreas, Menus
- Binary Stream IO
- Threads
- Networking
- Sorting & Searching
- Recursion
- Stacks, Queues, Lists
- Jars and Packages

Project



Java Applications

- So far, we have been writing text-based user interface Java programs for users to enter and display data
- The n… exact seque…
- Progr… xt instru… e to condi…
- *proce…*

```
 ----jGRASP exec: java TestJack
Building code: admin
The code must be 3 characters long.
Building code: ADM
Room number: 0
The room number must be in the range 1...99999.
Room number: 120989
The room number must be in the range 1...99999.
Room number: 234
Jack number: 0
The jack number must be in the range 1...24.
Jack number: 99
The jack number must be in the range 1...24.
Jack number: 5
Jack type: empty
The type must be "ENET", "PH", or "VID".
Jack type: P_H
The type must be "ENET", "PH", or "VID".
Jack type: PH
ADM:234:05 is of type PH
 ----jGRASP: operation complete.
```

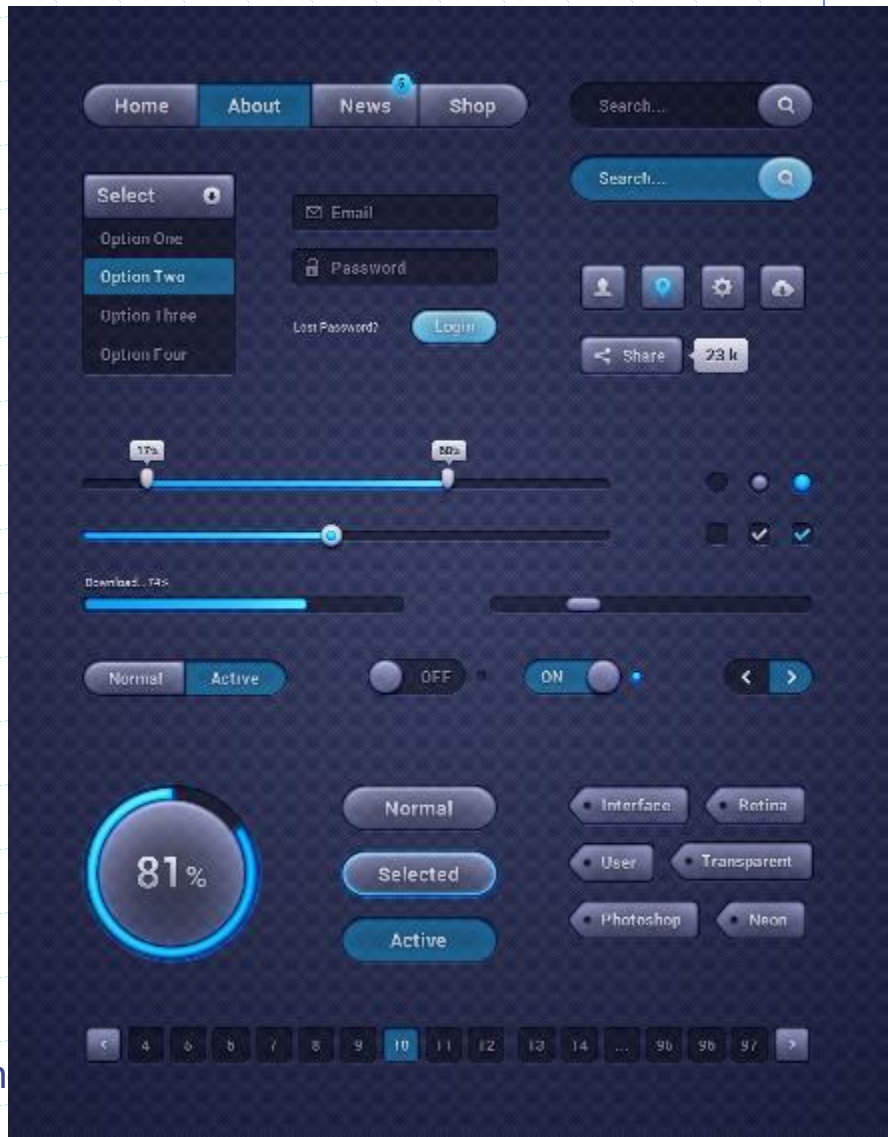
User Interface

```
----jGRASP exec: java TestJack
Building code: admin
The code must be 3 characters long.
Building code: ADM
Room number: 0
The room number must be in the range 1...99999.
Room number: 120989
The room number must be in the range 1...99999.
Room number: 234
Jack number: 0
The jack number must be in the range 1...24.
Jack number: 99
The jack number must be in the range 1...24.
Jack number: 5
Jack type: empty
The type must be "ENET", "PH", or "VID".
Jack type: P_H
The type must be "ENET", "PH", or "VID".
Jack type: PH
ADM:234:05 is of type PH
----jGRASP: operation complete.
```

- Text-based used interface
- Graphical User Interface (GUI)

Graphical User Interface (GUI)

- GUI (pronounced GOO-EE)
- Gives a program its look and feel
- Event-Driven Programming



Java GUI packages
JavaFX vs. Swing vs. AWT

- **AWT (Abstract Windows Toolkit) GUI** library – first java GUI support
- AWT was replaced by a more robust and flexible library **Swing** in 1998.
- Java 8 introduced in 2014 the **JavaFX GUI**
  - JavaFX is a software platform for creating and delivering desktop applications, as well as rich web applications that can run across a wide variety of devices. JavaFX has support for desktop computers and web browsers on Microsoft Windows, Linux, and macOS, as well as mobile devices running iOS and Android

JavaFX installation: Step #1

This is not required for java 8, but required for Java 11!

- Download JavaFX windows SDK and export it into an unchangeable directory path, for example: ../ISTE121/javafx-sdk-17.0.1
- JavaFX SDK site: https://gluonhq.com/products/javafx/

JavaFX installation: Step #1

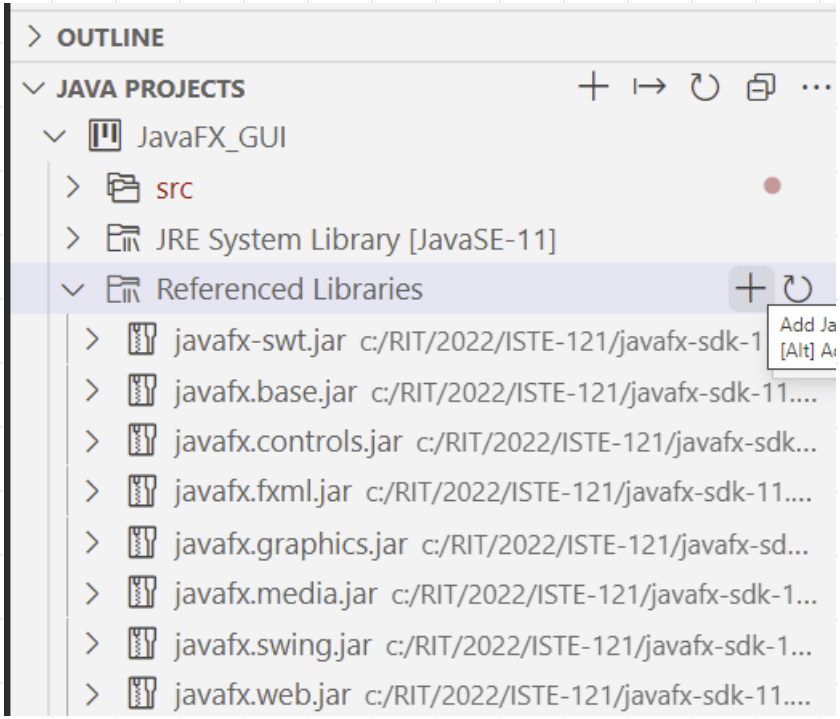
macOS	17.0.1	aarch64	SDK	Download [SHA256]
macOS	17.0.1	aarch64	jmods	Download [SHA256]
macOS	17.0.1	aarch64	Monocle SDK	Download [SHA256]
macOS	17.0.1	x64	SDK	Download [SHA256]
macOS	17.0.1	x64	jmods	Download [SHA256]
macOS	17.0.1	x64	Monocle SDK	Download [SHA256]
Windows	17.0.1	x64	SDK	Download [SHA256]
Windows	17.0.1	x64	jmods	Download [SHA256]
Windows	17.0.1	x64	Monocle SDK	Download [SHA256]
Windows	17.0.1	x86	SDK	Download [SHA256]
Windows	17.0.1	x86	jmods	Download [SHA256]
Windows	17.0.1	x86	Monocle SDK	Download [SHA256]
Javadoc	17.0.1		Javadoc	Download [SHA256]

JavaFX installation: Step #2

- VSCODE - Create a new Java project
- Configure project to use java version 11 (jdk11.0.12_7)
- Add all JAR files from the javafx-sdk-17.0.1\lib directory

JavaFX installation: Step #3

- Add all JAR files from the javafx-sdk-11.0.2\lib directory



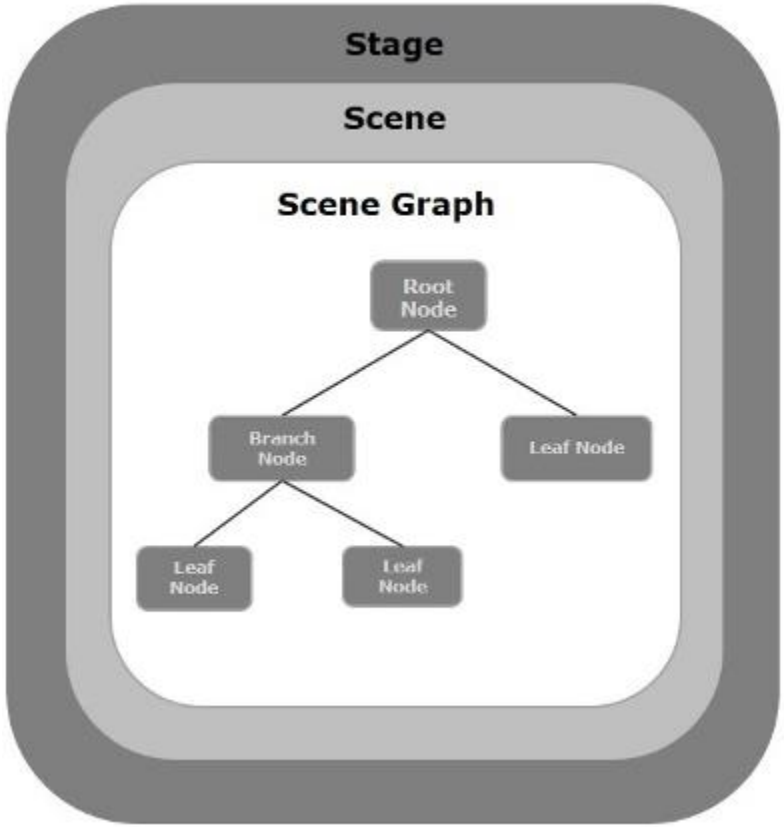
JavaFX installation: Step #4

- Add vmArgs to the „launch.json”. Please change the path to your JavaFX SDK path

"vmArgs": "--module-path C:\\RIT\\2022\\ISTE-121\\javafx-sdk-11.0.2\\lib --add-modules javafx.controls"

```
{
    // Use IntelliSense to learn about possible attributes.
    // Hover to view descriptions of existing attributes.
    // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387
    "version": "0.2.0",
    "configurations": [
        {
            "type": "java",
            "name": "Launch Current File",
            "request": "launch",
            "mainClass": "${file}",
            "vmArgs": "--module-path C:\\RIT\\2022\\ISTE-121\\javafx-sdk-11.0.2\\lib --add-modules javafx.controls"
        },
        {
            "type": "java",
            "name": "Launch JavaFXGUI1",
            "request": "launch",
            "mainClass": "JavaFXGUI1",
            "projectName": "JavaFX_GUI_447263fe",
```

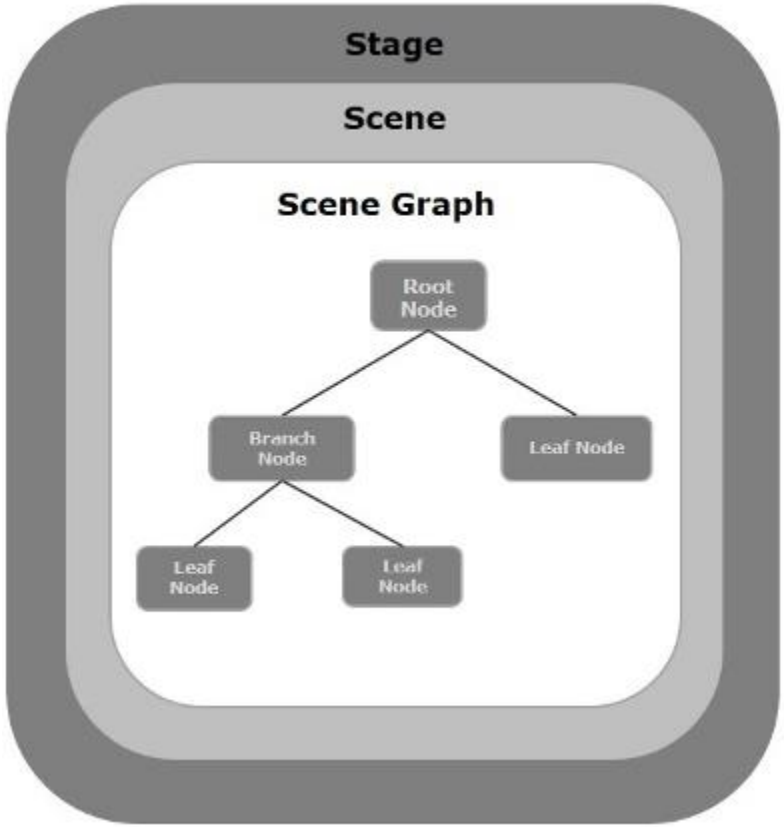
JavaFX application



- Stage – a window containing all objects of a JavaFX application
- **javafx.stage** package

https://www.tutorialspoint.com/javafx/javafx_application.htm

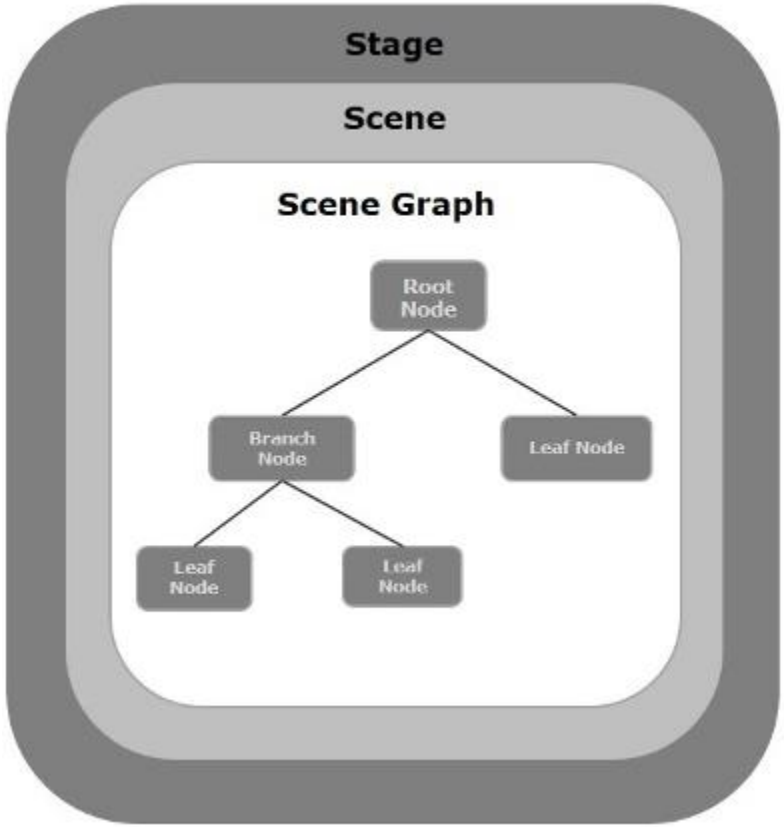
JavaFX application



- Scene – represents the physical contents of a JavaFX application
- **javafx.scene** package

https://www.tutorialspoint.com/javafx/javafx_application.htm

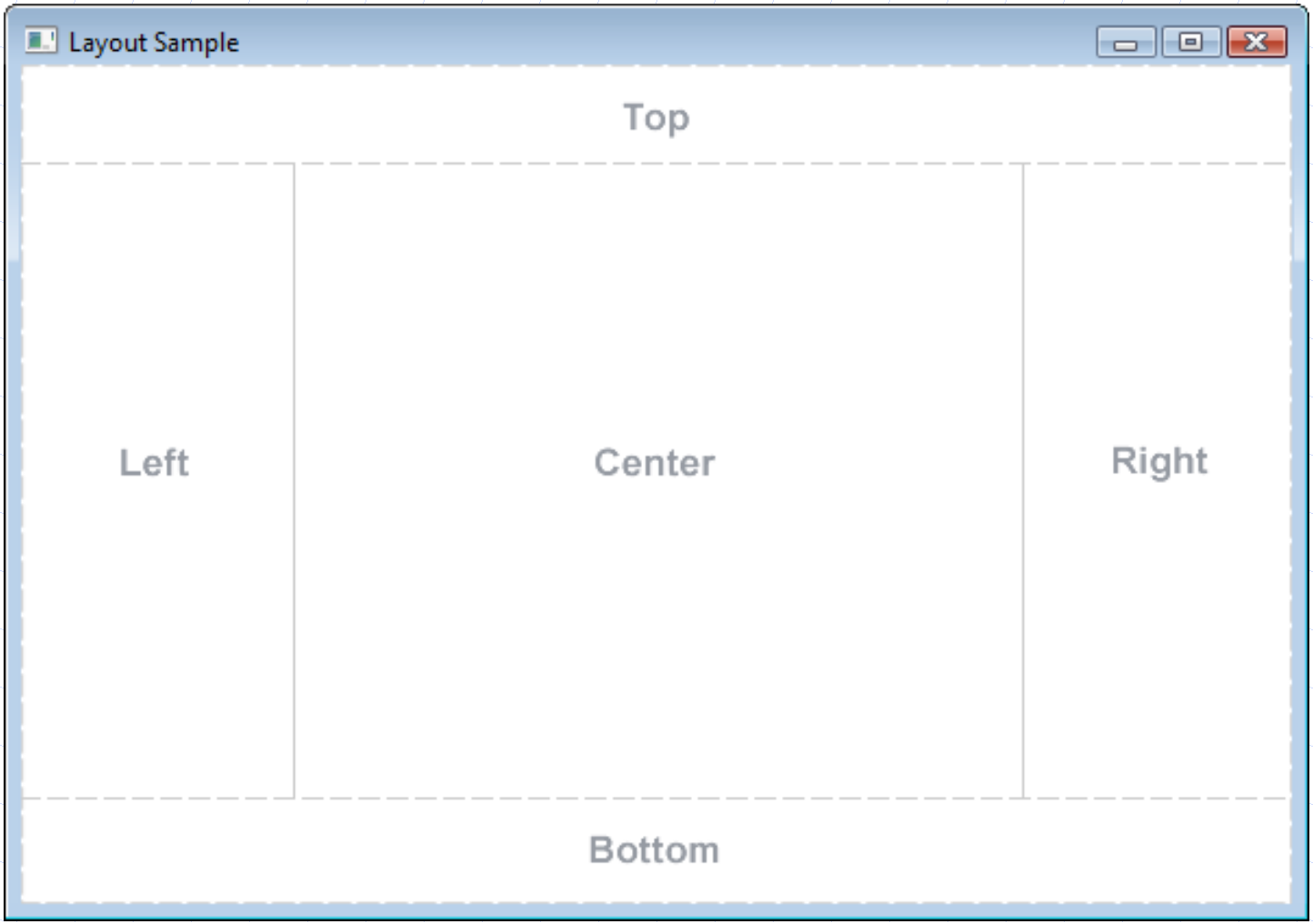
JavaFX application



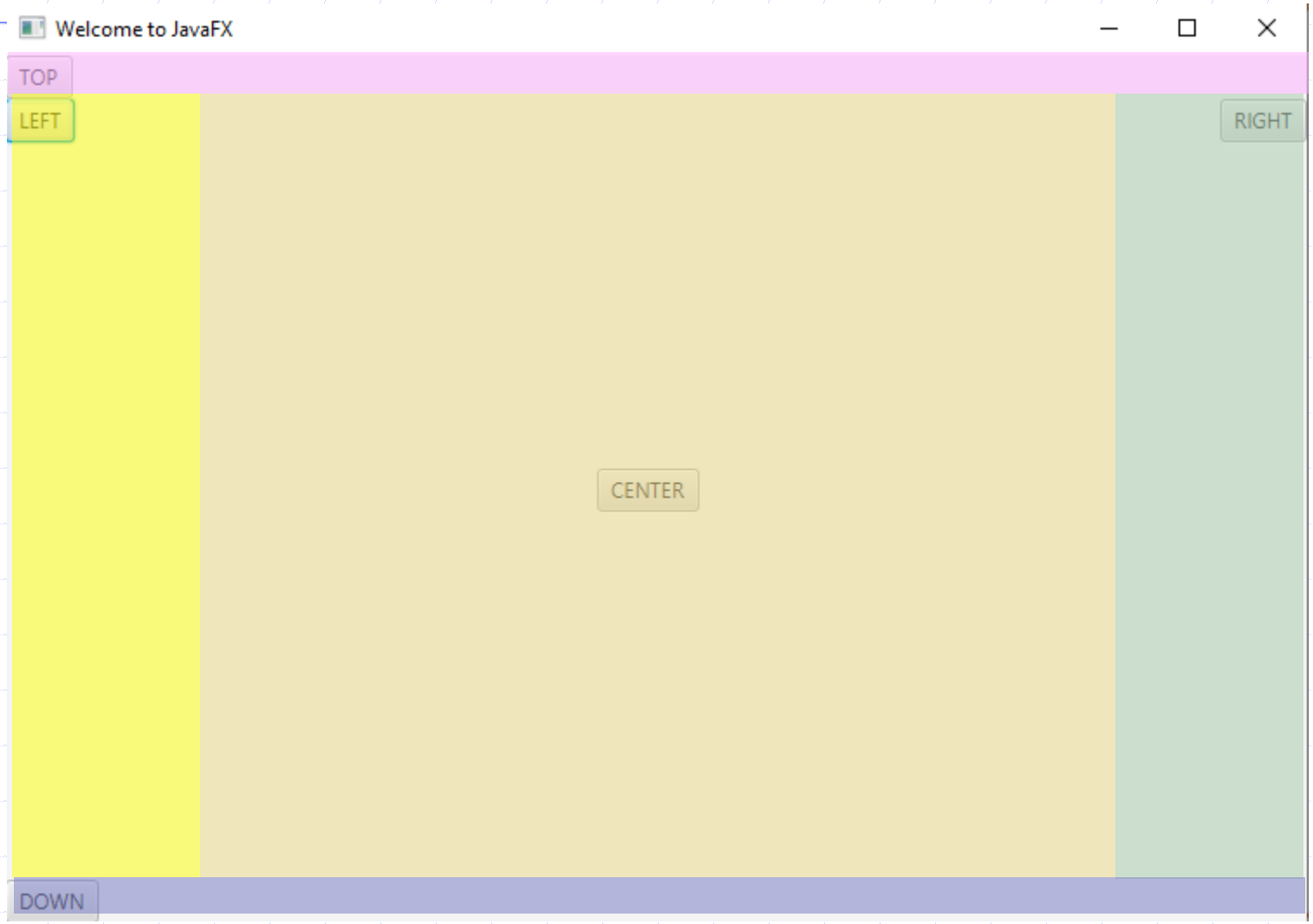
- **Scene Graph and Nodes** – a tree-like data structure (hierarchical) representing the contents of a scene
- **Node** – visual/graphical object of a scene graph
- A node may include:
  - Geometrical object - circle, rectangle, polygon
  - UI Controls – Button, Checkbox, Choice Box, Text Area

https://www.tutorialspoint.com/javafx/javafx_application.htm

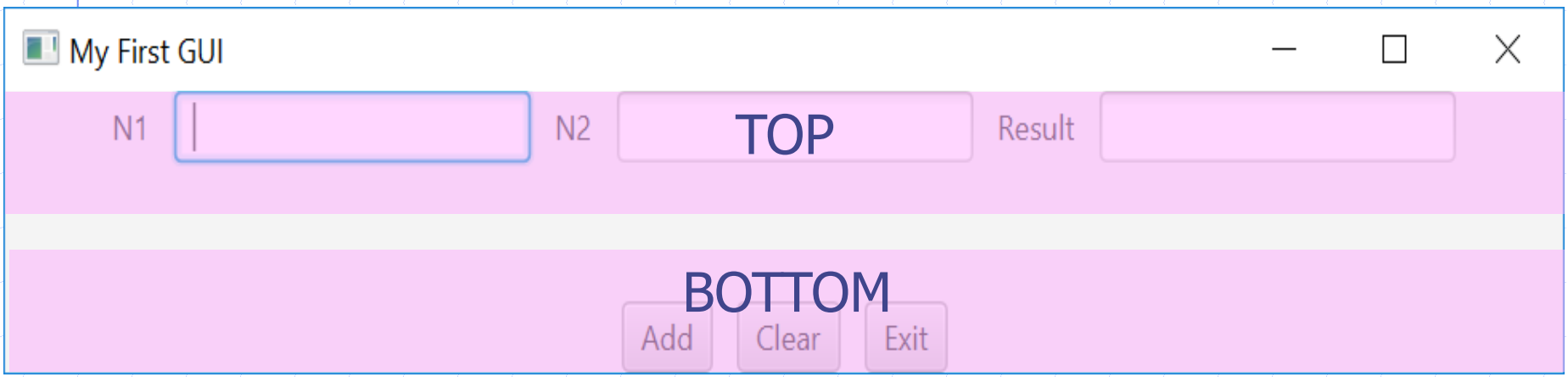
BorderPane Layout Manager



BorderPane with Buttons

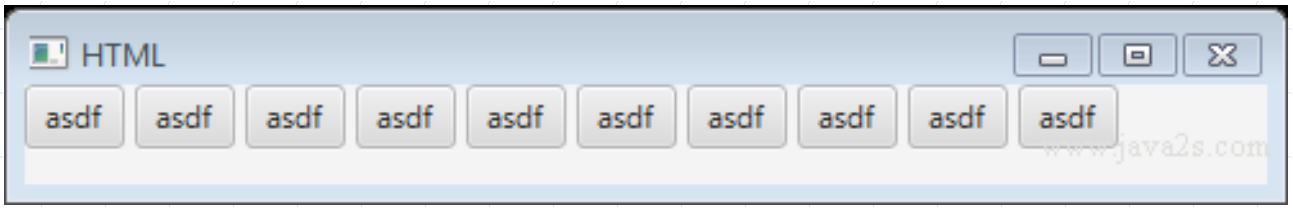


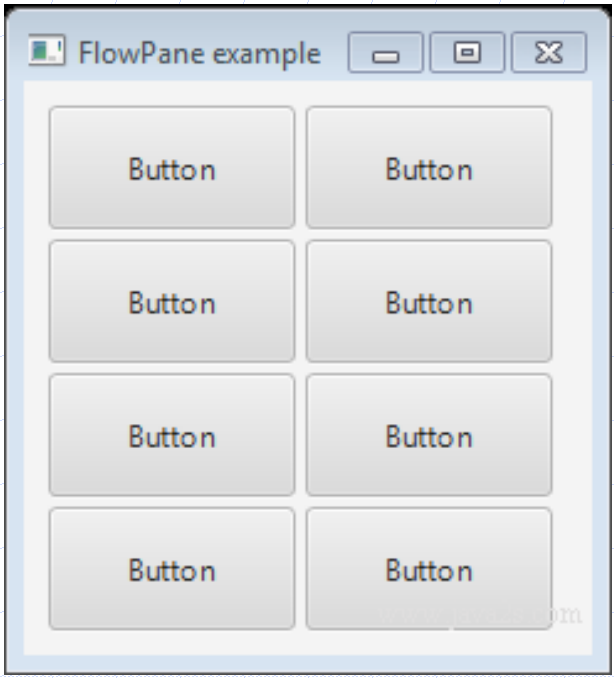
How to create something like this?



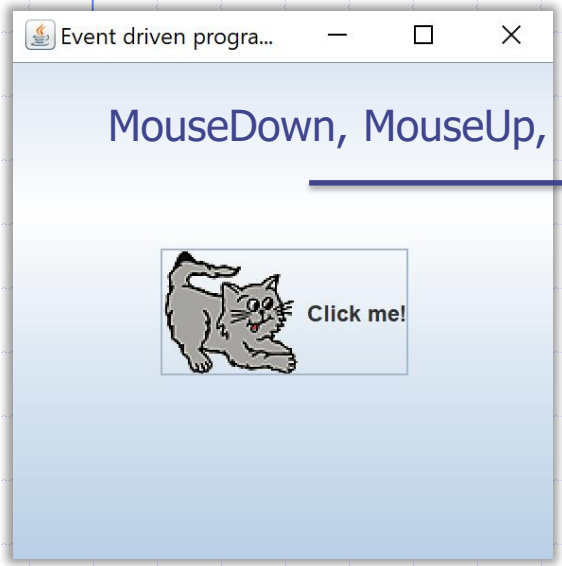
- Inside the TOP, add another layout, for example the **FlowLayout**

FlowPane Layout Manager





Event-Driven Programming



The event is triggered, we need to listen…

```
btnAdd.setOnAction(this);
```

```
public void handle(ActionEvent evt){
      System.out.println("YEAP!");
}
```

- The GUI program is driven by events, or actions initiated by the user, the operating system or the program itself
  - The user clicks on a button, enters text in a textbox, or chooses a menu option