



# Java Swing

Sadekujjaman Saju

SUST CSE – 16

Software Engineer – Re:cruit (Present)

Machine Learning Engineer – Inverse.AI (2018-2021)

# Objectives

- Java basics
  - Objects & Classes
  - Inheritance & Polymorphism
  - Abstraction & Encapsulation
- Intro of Swing
  - What is Swing?
  - Swing components

# Java Basics

- What is JAVA ???



# What is Java?

- A programming language.

&

- A platform(JDK & JRE).



# Why Java??

- Why should learn it???
- Simple !!!!!!!
- Platform Independent.
- Object Oriented
- Multi-thread
- Secured
- Robust
- More...



# Where Java??

- Mobile application (Android)
- Web application
- Desktop (Media player)
- Embedded systems
- Enterprise applications (Banking)
- Games
- More & more...



# Fundamentals



- Hello World!!!

# Fundamentals

- Hello World!!!

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello World!!!");  
    }  
}
```



# More fundamentals

- Hello World!
- Variables
- Condition (If – Else)
- Flow Control & Loop (For, While)
- Array & List
- Functions



# Object Oriented Programming



- What??
  - Programming paradigm.
  - Technique for System Modeling.
  - Concept of classes and objects.



# Object Oriented Programming

- What??
  - Programming paradigm.
  - Technique for System Modeling.
  - Concept of classes and objects.
- Why??
  - Ease in Software design.
  - Ease in Software maintenance.
  - Reusable Software.



# Object Oriented Programming



A simple programming paradigm.

- Ideas:
  - Objects & Classes
  - Inheritance & Polymorphism
  - Abstraction & Encapsulation



# Objects & Classes

- What is an Object?



# Objects & Classes

## What is an Object?

- Real world entity
- Contains / Represent data
- Has two part. ( state and behavior).
- States/Attributes: Age, Height, Weight.
- Behaviors: Barking, Running, Jumping.



# Objects & Classes

- What is Classes?



# Objects & Classes

## What is Classes?

- Class reflects concept (templates).
- Class is a blue print of an object.
- Class defines an object.
- Objects are instantiated from the class.



# Objects & Classes



Class: blue print of a car, design/template.

- four wheels, 5 doors, front seats and
- back seats and an engine.



Object:

- Can be any brand and made of the manufacturer
- Mercedes, BMW, Ford.



# Objects & Classes



## Syntax

```
public class class_Name{  
    Block of the code  
}
```



## Example

```
public class ClassExample {  
    // Body of the code here  
}
```

# Inheritance & Polymorphism



- Inheritance?
  - Child-parent relationship between classes.
  - Code re-usability
  - Code readability

# Inheritance & Polymorphism



- Polymorphism?
  - One task is perform by different ways!
  - modifying the parent's methods and creating different versions of them
  - Method overriding & overloading.

# Abstraction & Encapsulation



- Abstraction?
  - Hiding details.
  - Expose essential information only.
  - Achieve via abstract classes and interfaces



# Abstraction & Encapsulation



- Encapsulation?
  - Data hiding from external classes.
  - Binding data into a single unit.
  - A simple java class ( Java Bean)

!!! SWING !!!

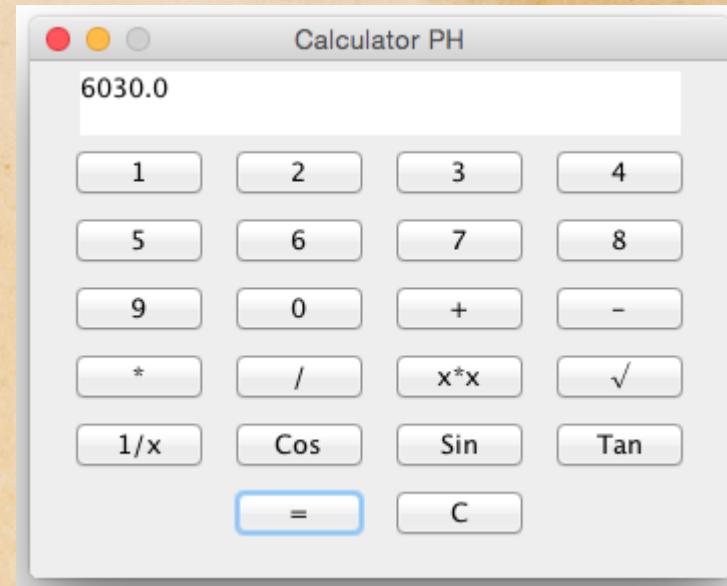
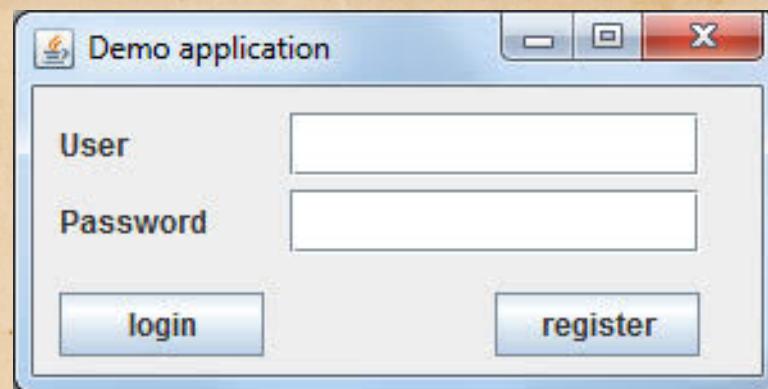


# Swing

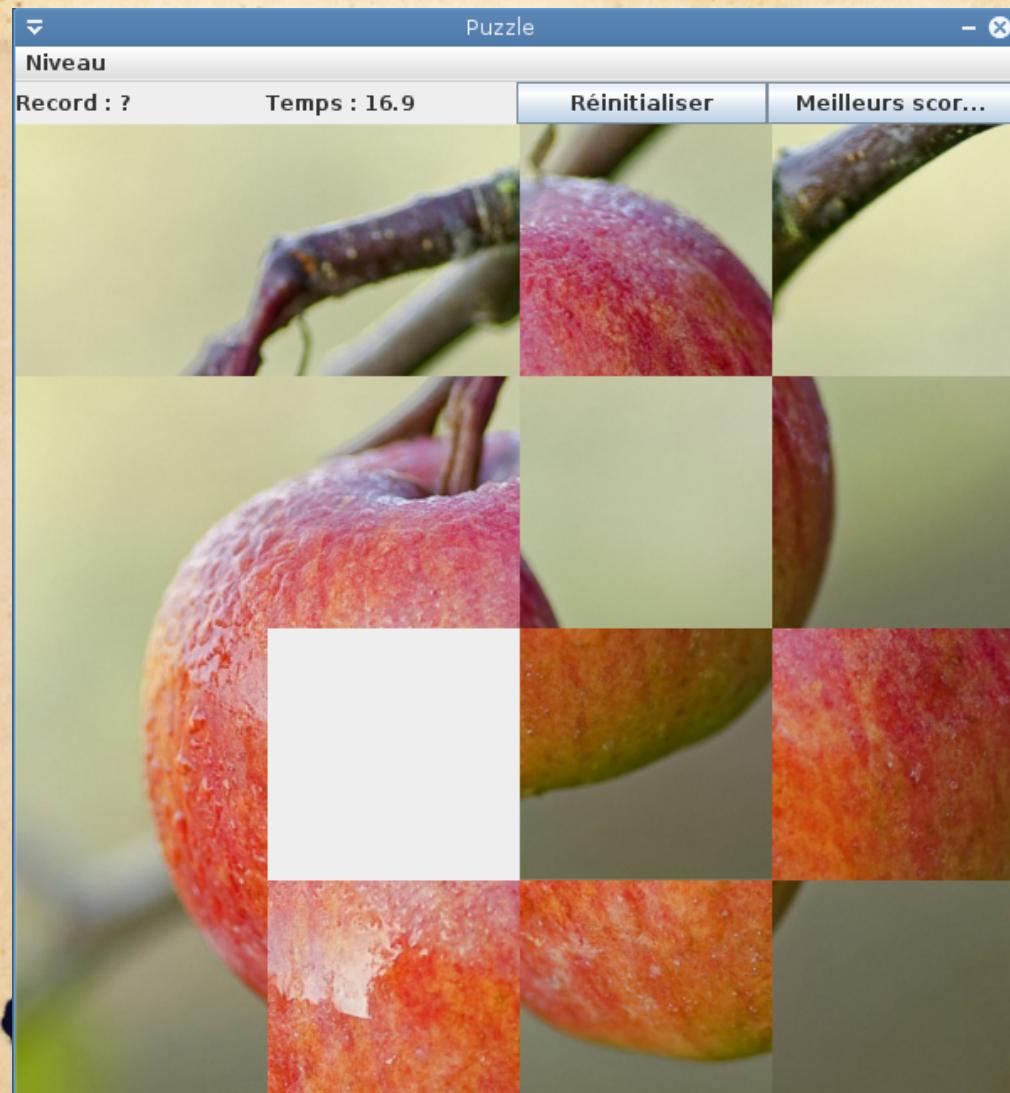
- What is Swing?



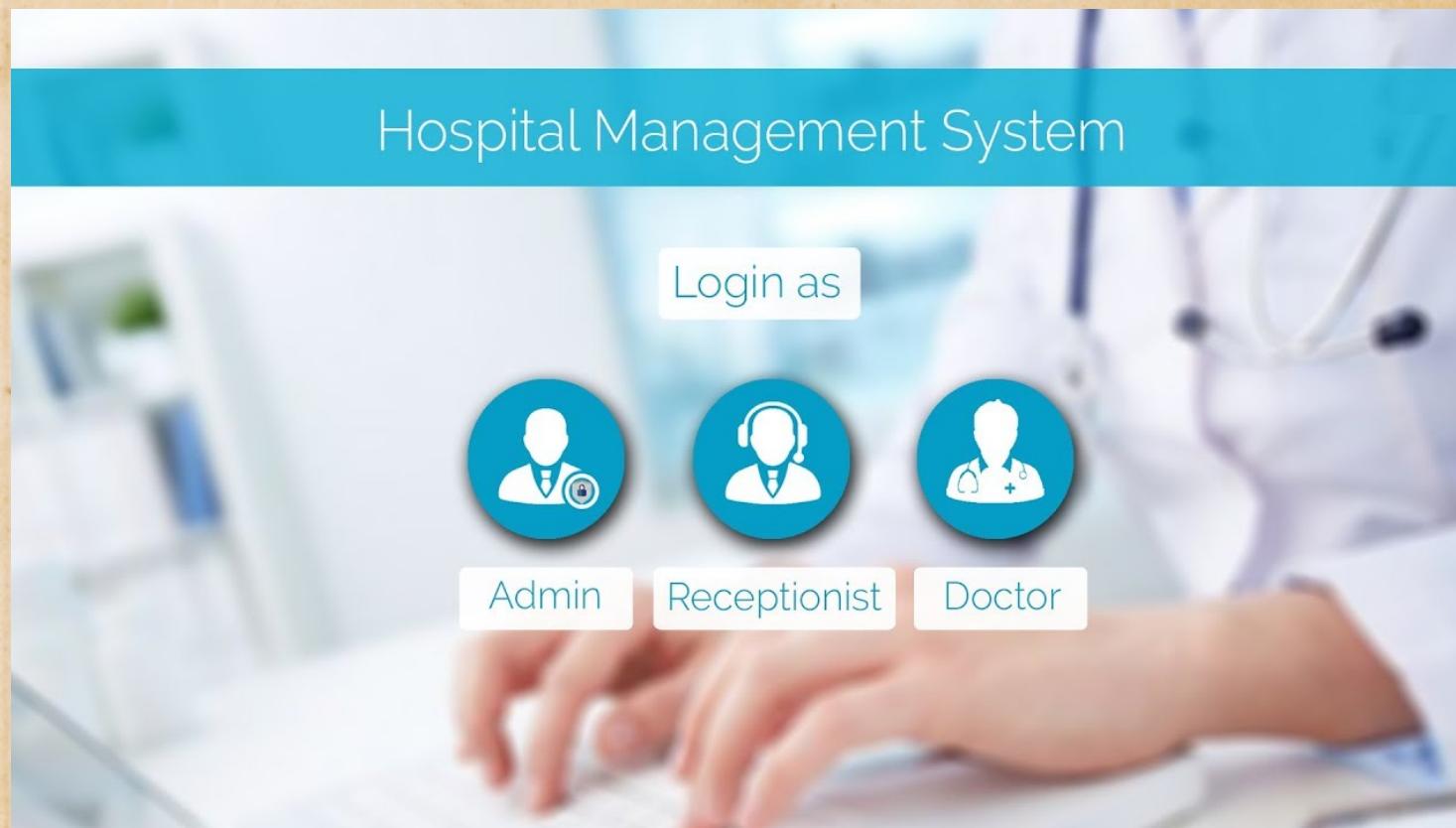
# Swing



# Swing



# Swing



# What is Swing?

- A Java module.
- Part of Java Foundation Classes(JFC)
- JFC – Provides GUI.
- Used to create window-based applications.
- Entirely written in java
- Platform independent



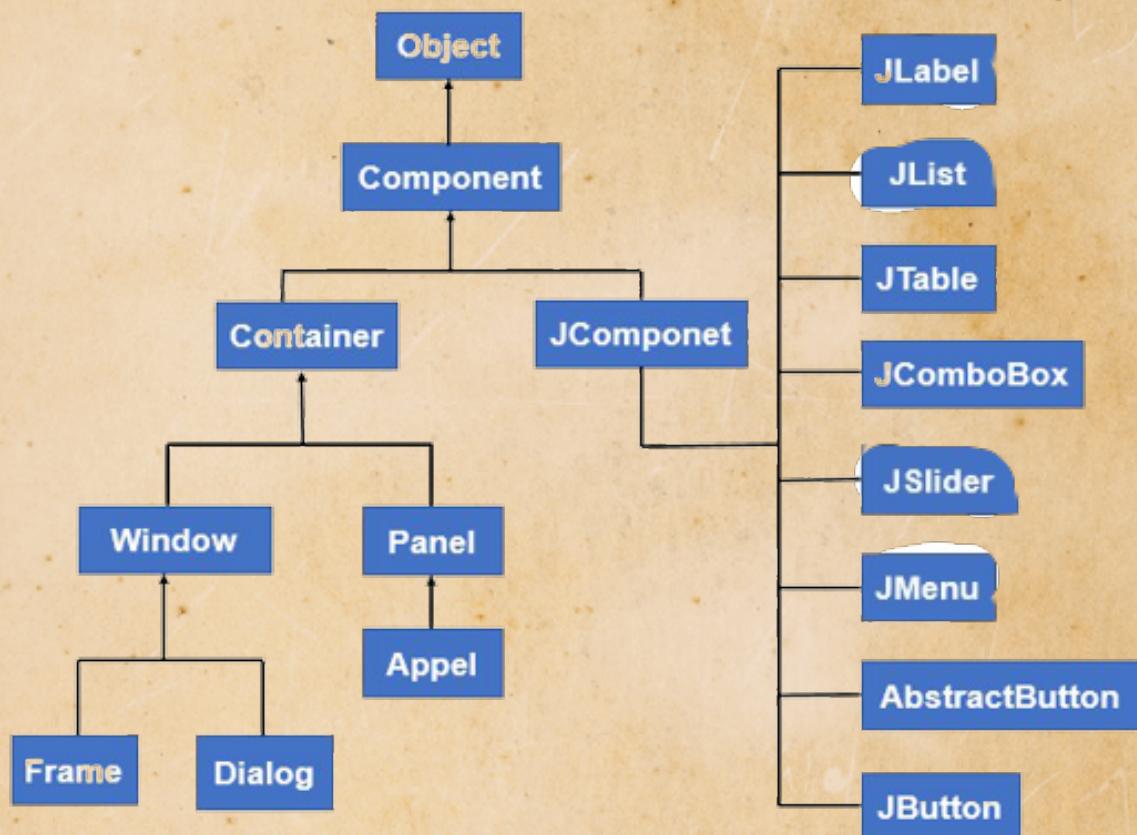
# What is Swing?

- Provides lightweight components.
- JButton, JTextField, JRadioButton etc.
- Follows MVC pattern.



# Components Tree

Hierarchy of Java Swing classes



# Swing components

- Object: All classes derive from Object.
- Component: Shown on the screen,
  - Can interact with users.



# Swing components



- Container: Hold components.
- Window: Specialized container
  - Has a border and title bar.
  - Buttons that minimize, maximize and close.

# Swing components

- JFrame: Serves as the basis for GUI.
  - Swing app includes at least one Jframe.
- JComponent: Basis for all other swing components.

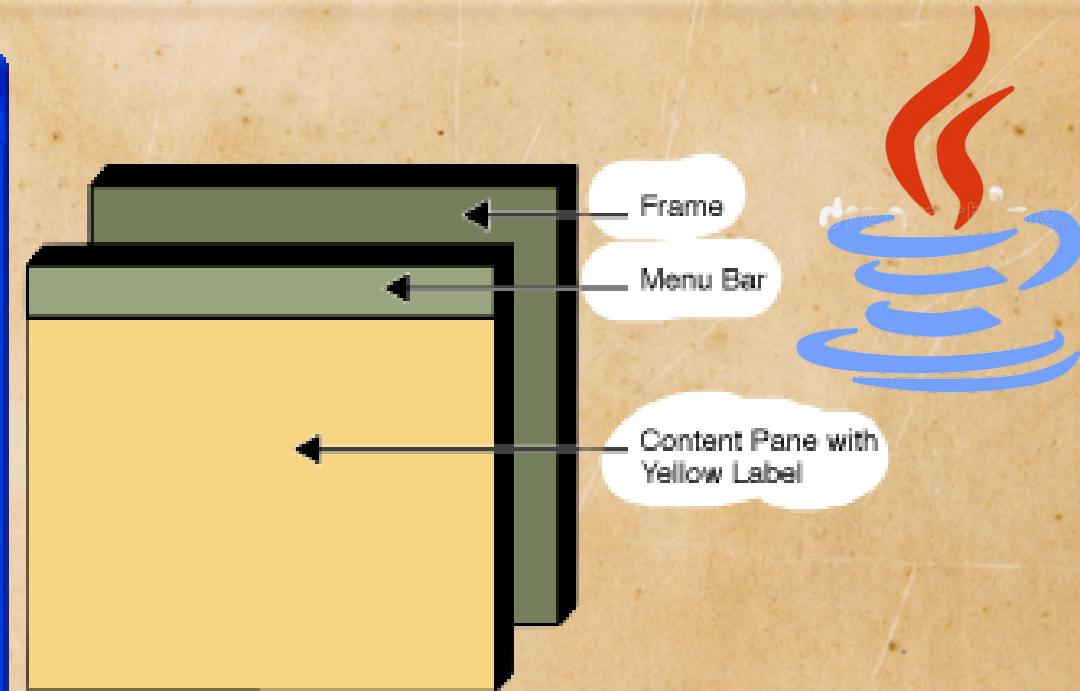


# Swing components

- JPanel: Used to organize and control the layout of other components.
- JLabel: Displays simple text value.



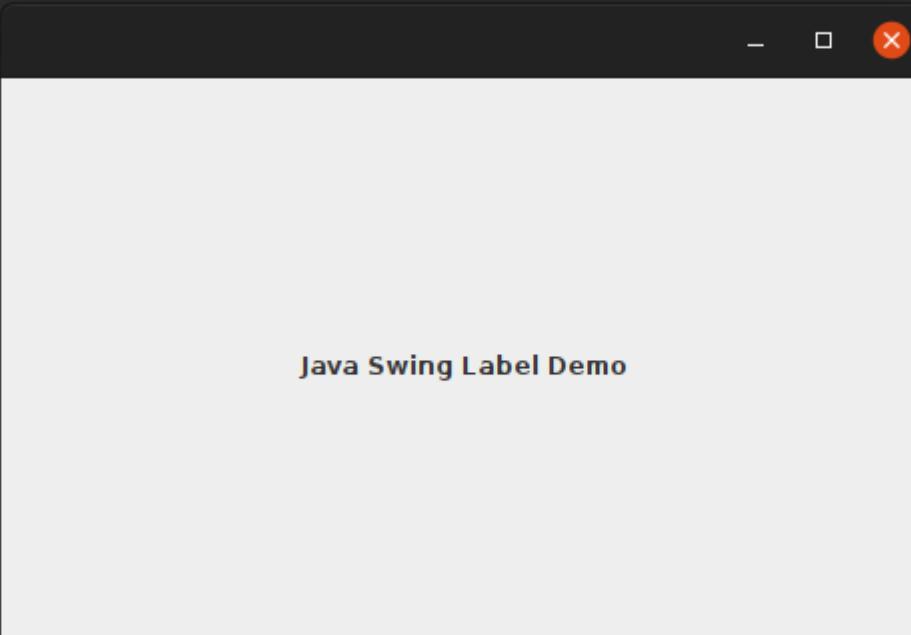
# Top level containers



# JLabel

- Used to displays text on the screen.

```
public static void main(String[] args) {  
    JLabel label = new JLabel("Java Swing Label Demo",  
    SwingConstants.CENTER);  
    JFrame frame = new JFrame("Java Swing Label Demo");  
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    frame.add(label);  
    frame.pack();  
    frame.setVisible(true);  
}
```



# JTextField

- Allows to edit a single line text.

```
public static void main(String[] args) {
    JFrame f = new JFrame("TextField Example");
    JTextField t1 = new JTextField("Hello World! I'm
    t1.setBounds
    JTextField t
    t2.setBounds
    f.add(t1);
    f.add(t2);
    f.setSize(40
    f.setLayout(
    f.setVisible
}
```



# JButton

- Allows to click a button.

```
public static void main(String[] args) {  
    JFrame f = new JFrame("TextField Example");  
    JButton login = new JButton("Login");  
    login.setBo  
    JTextField  
    t1.setBounds(100, 100, 200, 30);  
    JTextField  
    t2.setBounds(100, 150, 200, 30);  
    f.add(t1);  
    f.add(t2);  
    f.add(login);  
    f.setSize(400, 300);  
    f.setLayout(new FlowLayout());  
    f.setVisible(true);  
}
```



# Swing applications

- Notepad.
- Canteen management.
- Patient management.
- Calculator
- Online Game
- More & more...





Thank You !!!!