Fpt University

FPT POLYTECHNIC

LẬP TRÌNH JAVA 3

BÀI 3: LAYOUT, TIMER, JSLIDER

PHẦN 1

www.poly.edu.vn

Mục tiêu

- Kết thúc bài học này bạn có khả năng
  - Layout Manager
    - FlowLayout
    - BorderLayout
    - CardLayout
    - TabbedPaneLayout
    - GridLayout
    - GridBagLayout
    - BoxLayout
    - GroupLayout
  - Timer - ProgressBar
  - JSlider

LAYOUT MANAGER

- Các component trong Java Swing được chia làm hai loại là container và children, trong đó container là loại component dùng để chứa các component khác, ví dụ như JFrame, JPanel… còn children là các component như JLabel, Jbutton, JTextField… khi chúng ta đặt các component con lên các container thì việc đặt và sắp xếp theo vị trí nào đó và nó được gọi là Layout.
- Mỗi Container có một đối tượng Layout Manager
- Layout Manager là một đối tượng quyết định cách sắp xếp vị trí của các Component bên trong một Container.
- Các Layout Manager "implements" từ interface LayoutManager hoặc LayoutManger2.

FLOWLAYOUT

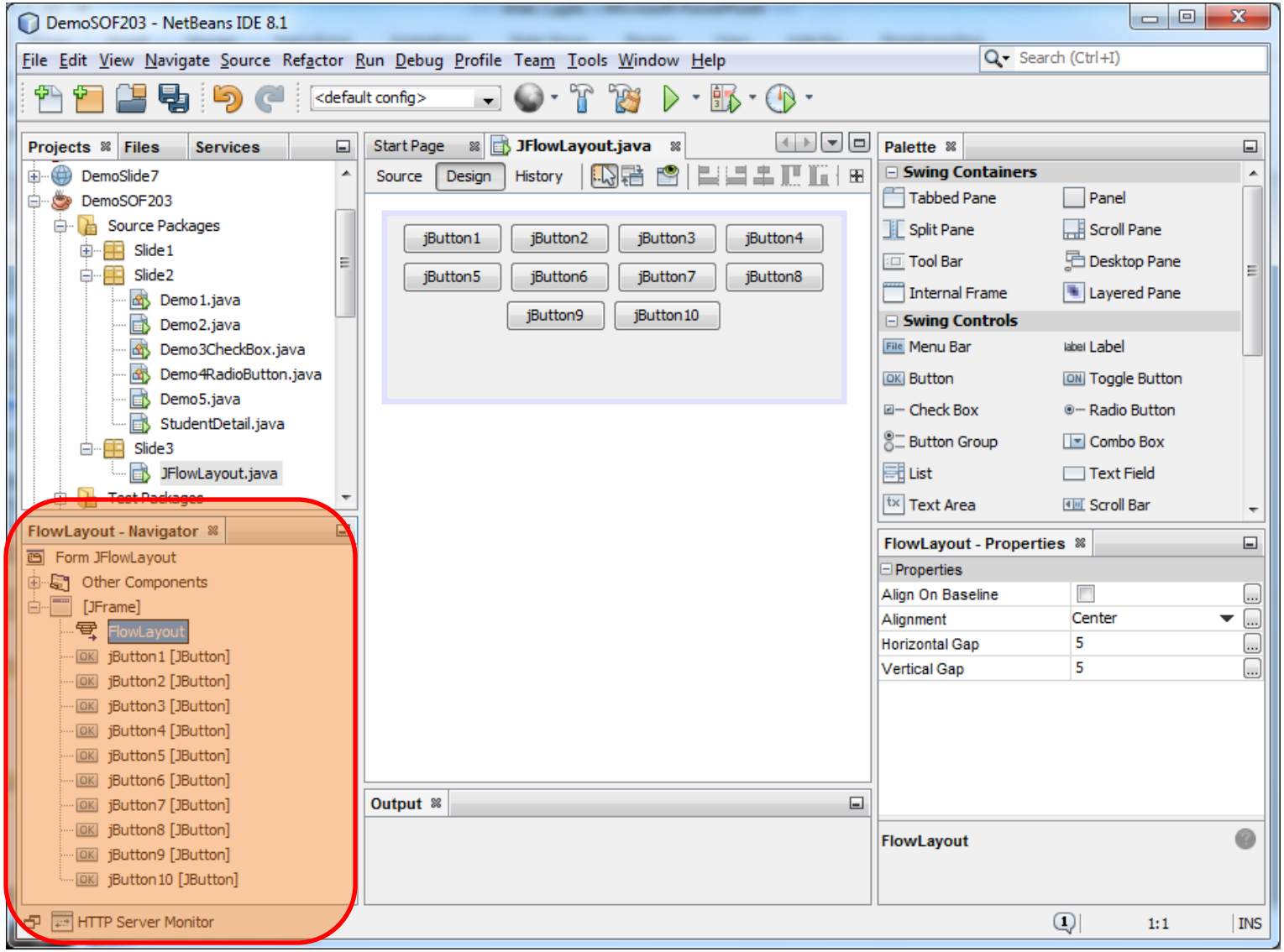
- Flow Layout bố trí các Component trong Container theo dòng, từ trái sang phải theo thứ tự thêm vào.
- Tạo dòng mới khi kích thước dòng còn lại không đủ chứa Component thêm vào.
- Flow Layout bố trí vị trí các Component phụ thuộc vào kích thước của Container.
- Mỗi dòng của các Component được window mặc định canh giữa theo chiều ngang . Có thể điều chỉnh canh trái hoặc phải

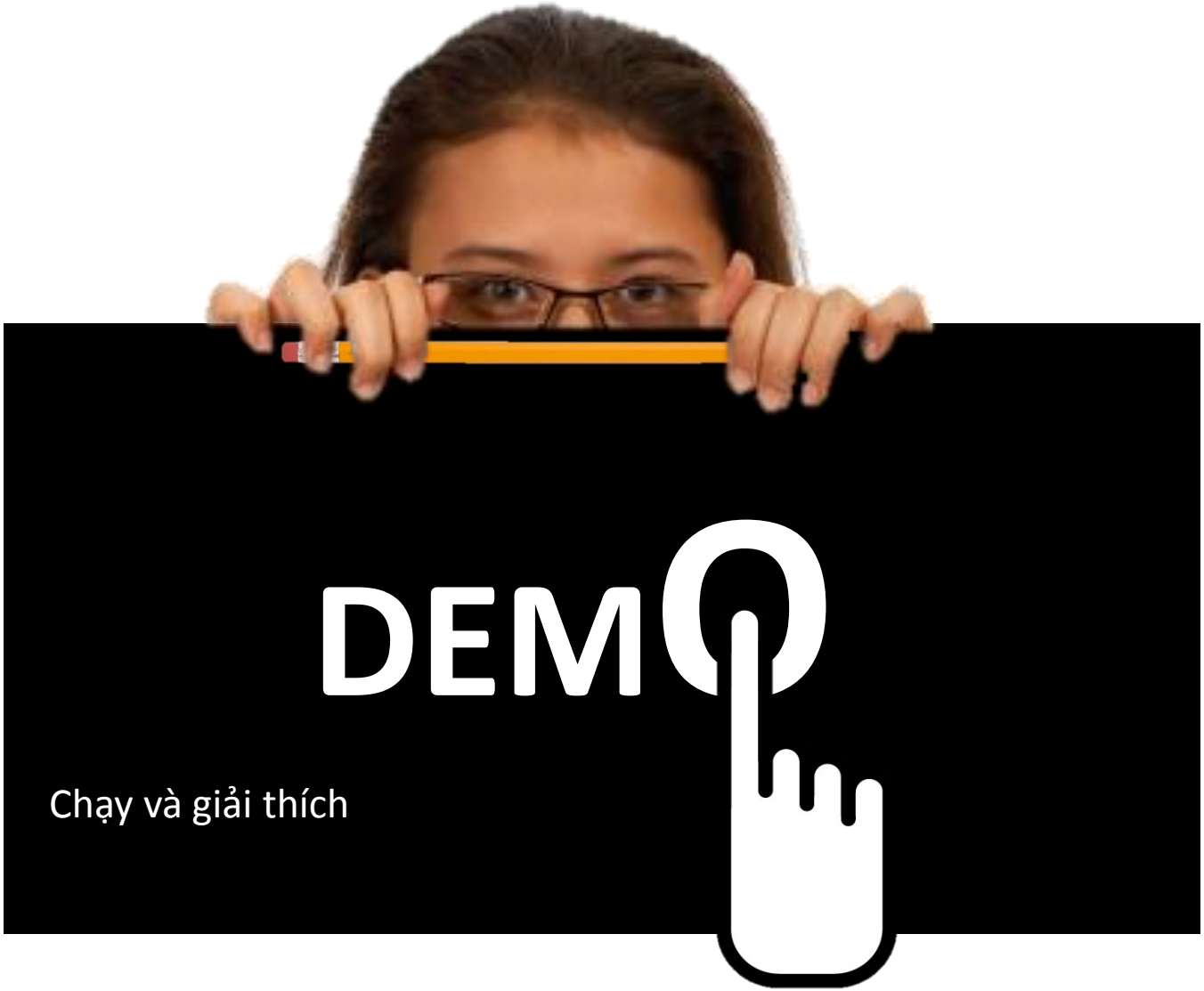
- Khởi tạo
  - public FlowLayout ()
    - align: FlowLayout.CENTER
    - vgap: 5px, hgap: 5px

  - FlowLayout (int align)
    - align: canh lề
    - FlowLayout.CENTER : Canh giữa
    - FlowLayout.LEFT; : Canh trái
    - FlowLayout.RIGHT; : Canh phải
  - FlowLayout(int align, int vgap, int hgap)
    - align : canh lề
    - vgap : kích thước chiều ngang
    - hgap: chiều dọc

- Phương thức
  - public void setAlignment(int align)
  - public void setHgap(int hgap)
  - public void setVgap (int vgap)
  - public int getAlignment()
  - public int getHgap ()
  - public int getVgap ()

Demo: FlowLayout

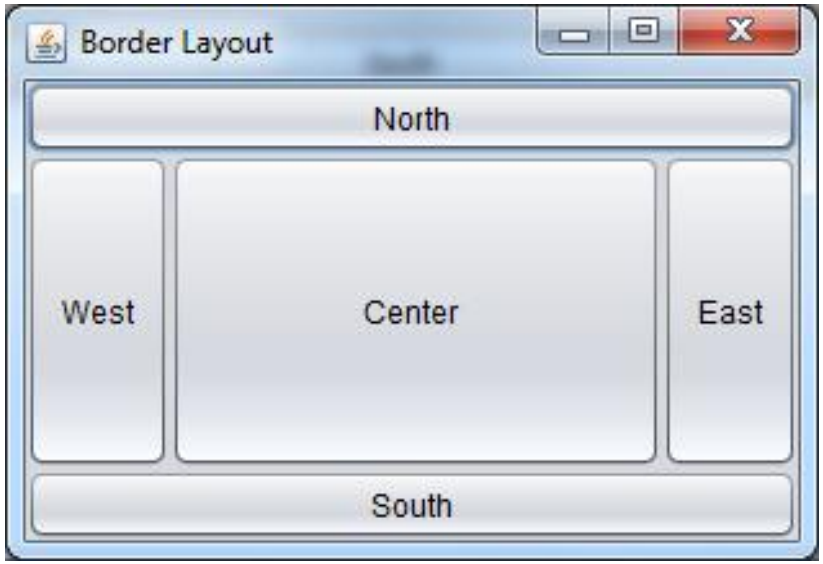




Chạy và giải thích

BORDER LAYOUT

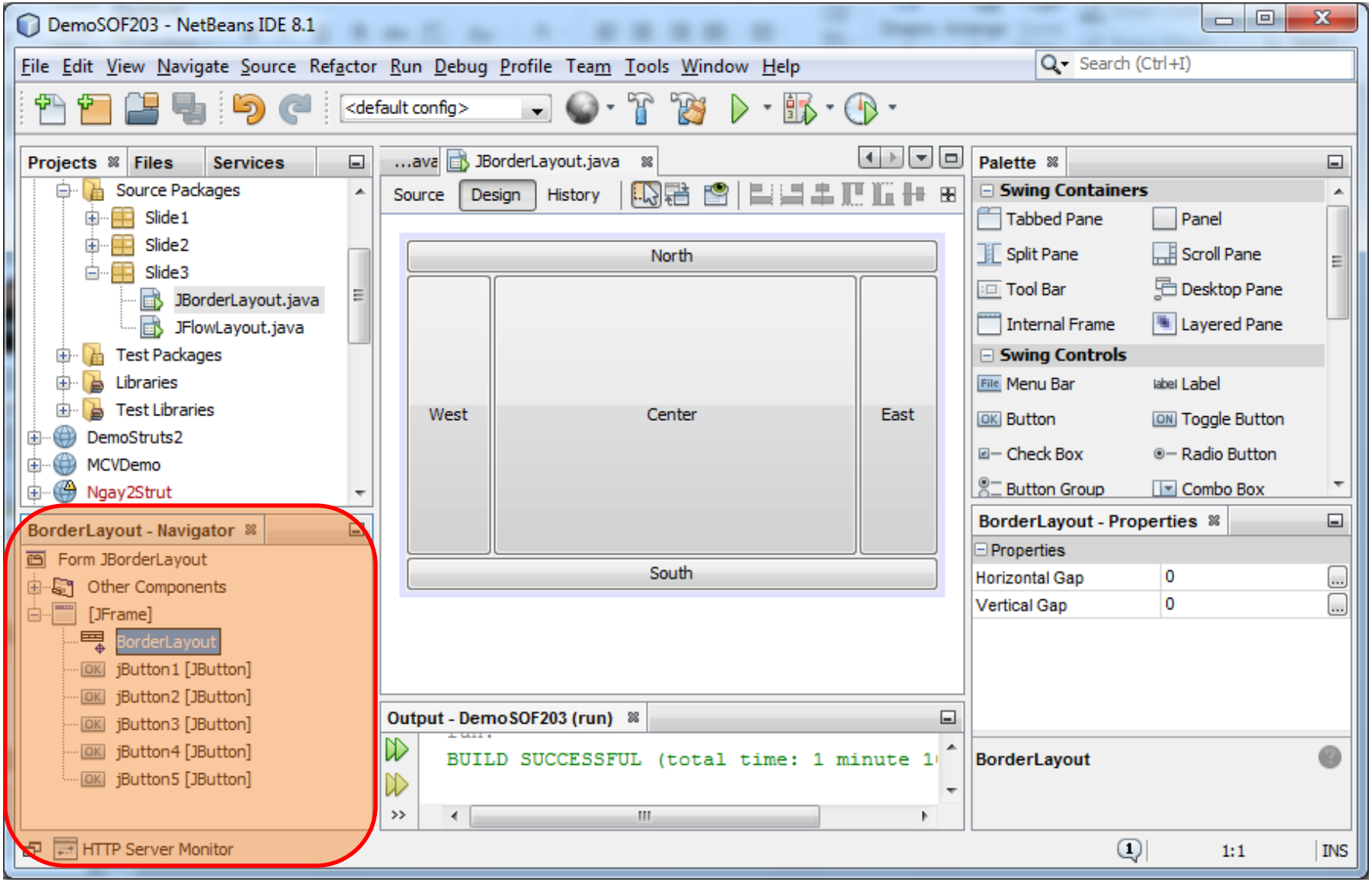
- Border Layout bố trí các Component bên trong Container theo 5 vùng: "North", "South", "East", "West" ,"Center".
- Nếu như không có 4 vùng : North, West, South, East. Thì vùng Center sẽ tràn đầy cửa sổ, thông thường khi đưa các control JTable, JTree, ListView, JScrollpane… ta thường đưa vào vùng Center để nó có thể tự co giãn theo kích thước cửa sổ giúp giao diện đẹp hơn.



BORDER LAYOUT

- khởi tạo:
  - public BorderLayout ()
    - hgap = 0
    - vgap = 0
  - public BorderLayout (int hgap, int vgap)
    - hgap: chiều ngang
    - vgap : chiều dọc

Demo: Border Layout

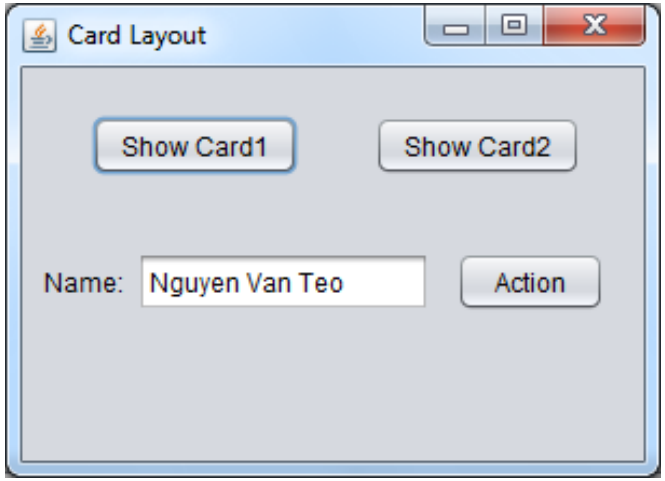


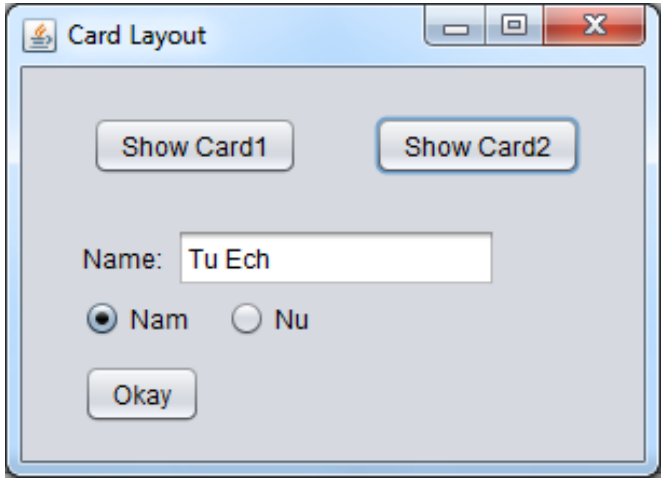
CARD LAYOUT

- Card Layout quản lý nhiều Card cùng một không gian hiển thị, tức là ứng với cùng 1 vị trí hiển thị đó thì ta có thể cho các control khác hiển thị tại những thời điểm khác nhau, mặc định control được add đầu tiên sẽ hiển thị.
- Card Layout giúp quản lý hai hay nhiều Component (thường là JPanel) để chia sẽ cùng một không gian hiển thị.
- Chỉ duy nhất Top Card được hiển thị.
- Mỗi "Card" có thể sử dụng Layout Manager riêng.
- Card nào cũng có thể là Top Card
- Có thể sử dụng **JTabbedPane** để thay cho **Card Layout**

CARD LAYOUT

- khởi tạo :
  - public CardLayout ()
    - hgap = 0
    - vgap = 0
  - public CardLayout (int hgap, int vgap)
    - hgap: chiều ngang
    - vgap : chiều dọc

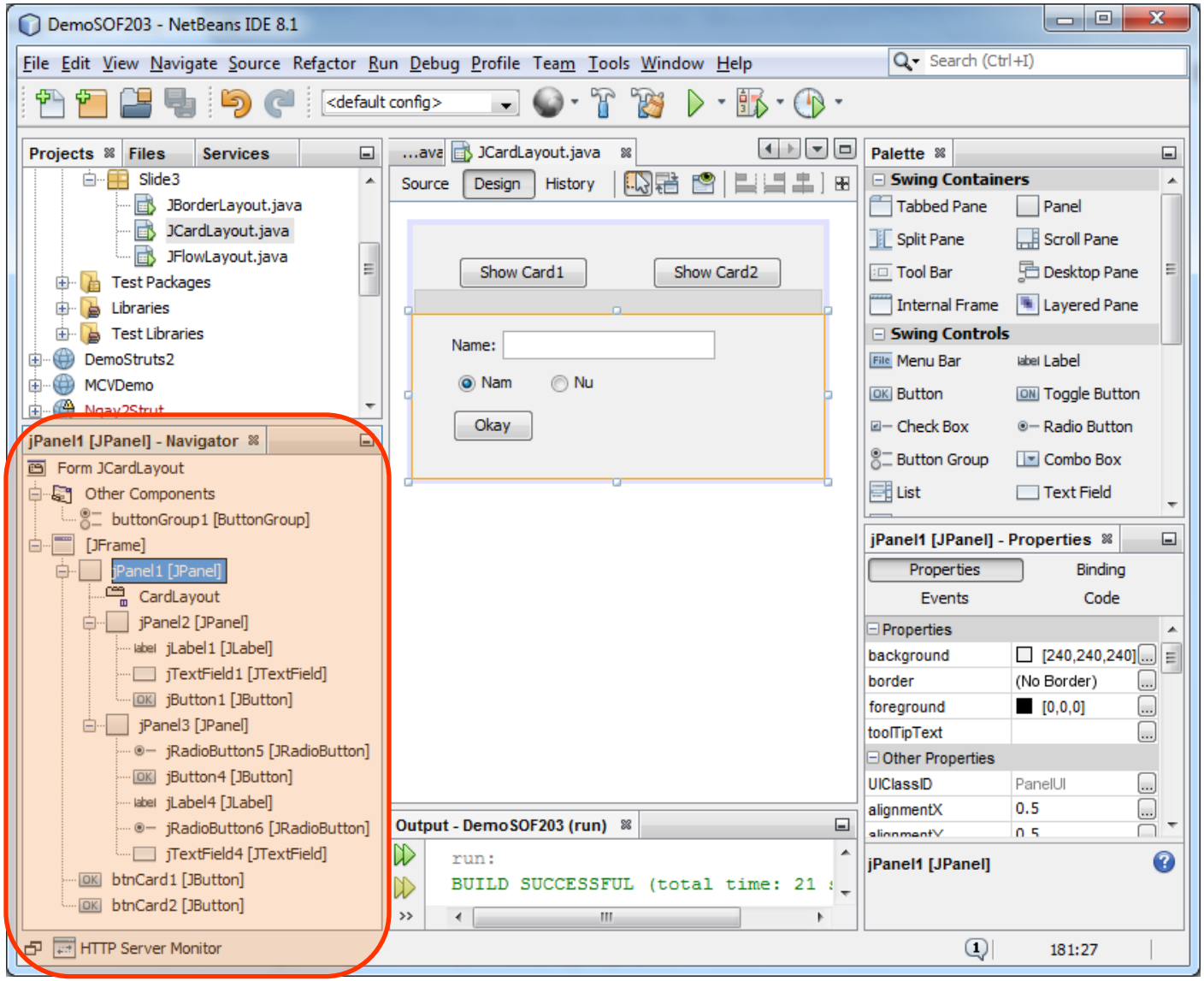


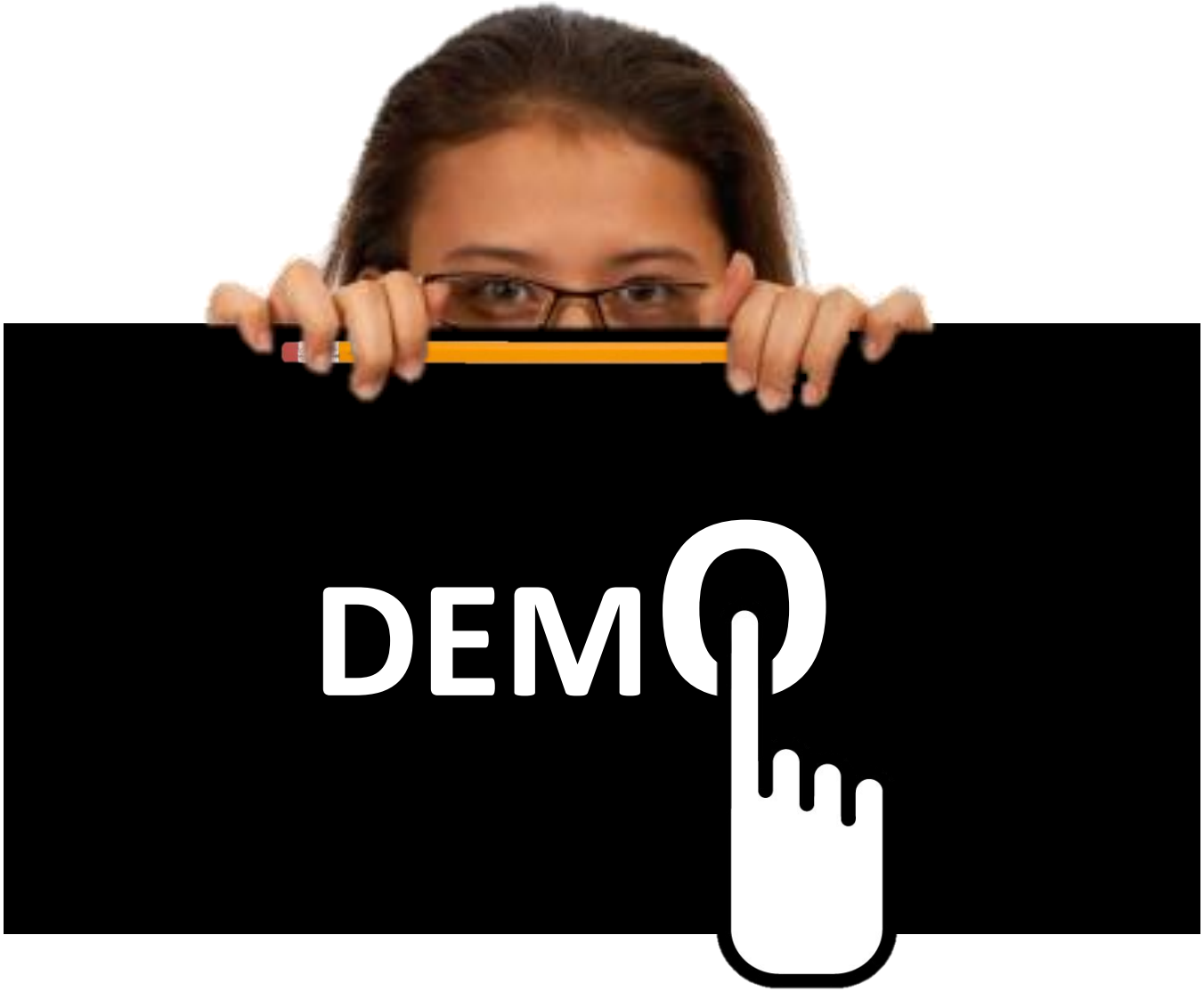


CARD LAYOUT

- Phương thức
  - public void setHgap(int hgap)
  - public void setVgap(int vgap)
  - public int getHgap()
  - public int getVgap()
  - public void next (Container parent)
  - public void previous(Container parent)
  - public void first(Container parent)
  - public void last(Container parent)
  - public void show(Container parent, String name)

Demo: Card Layout



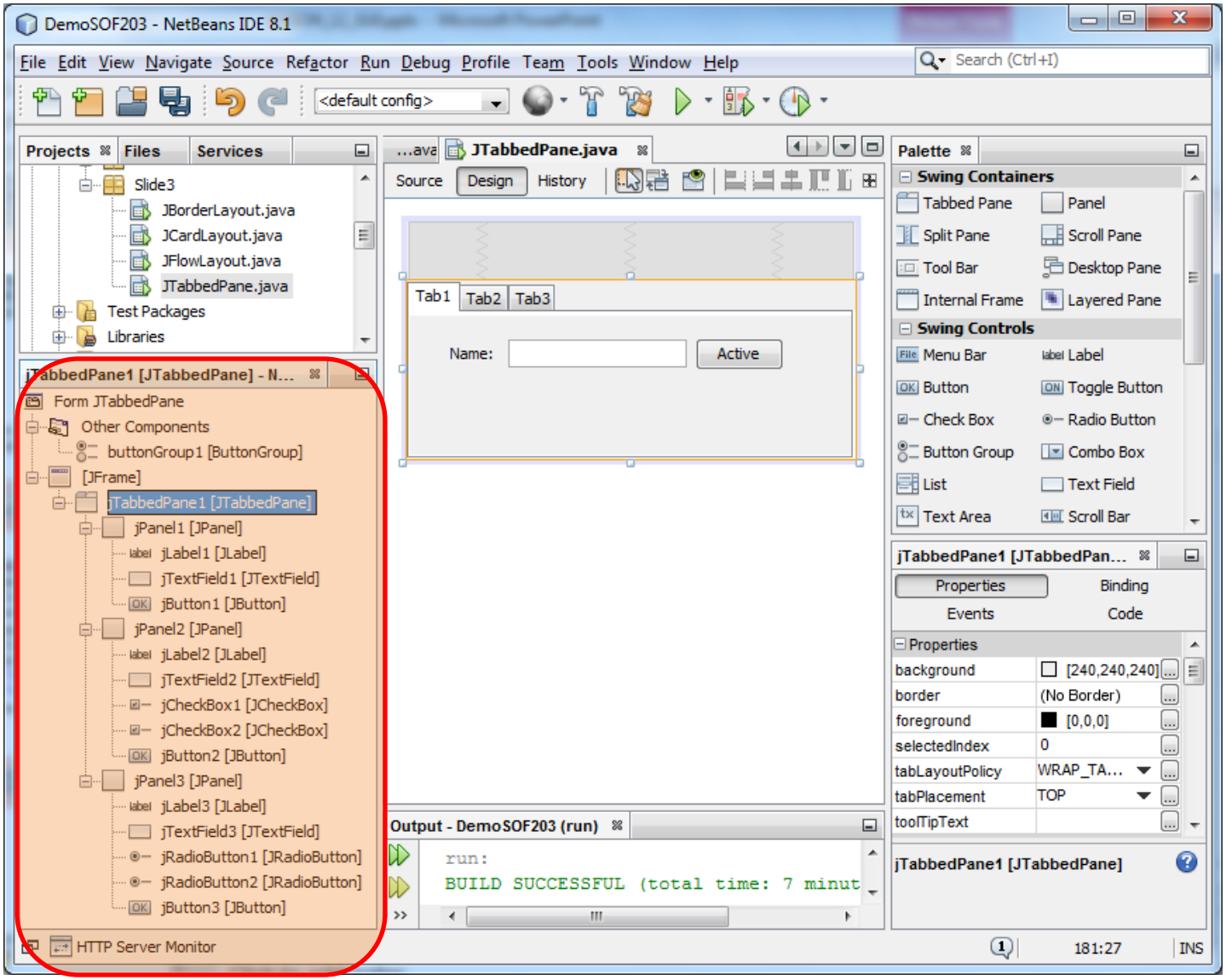

DEMO

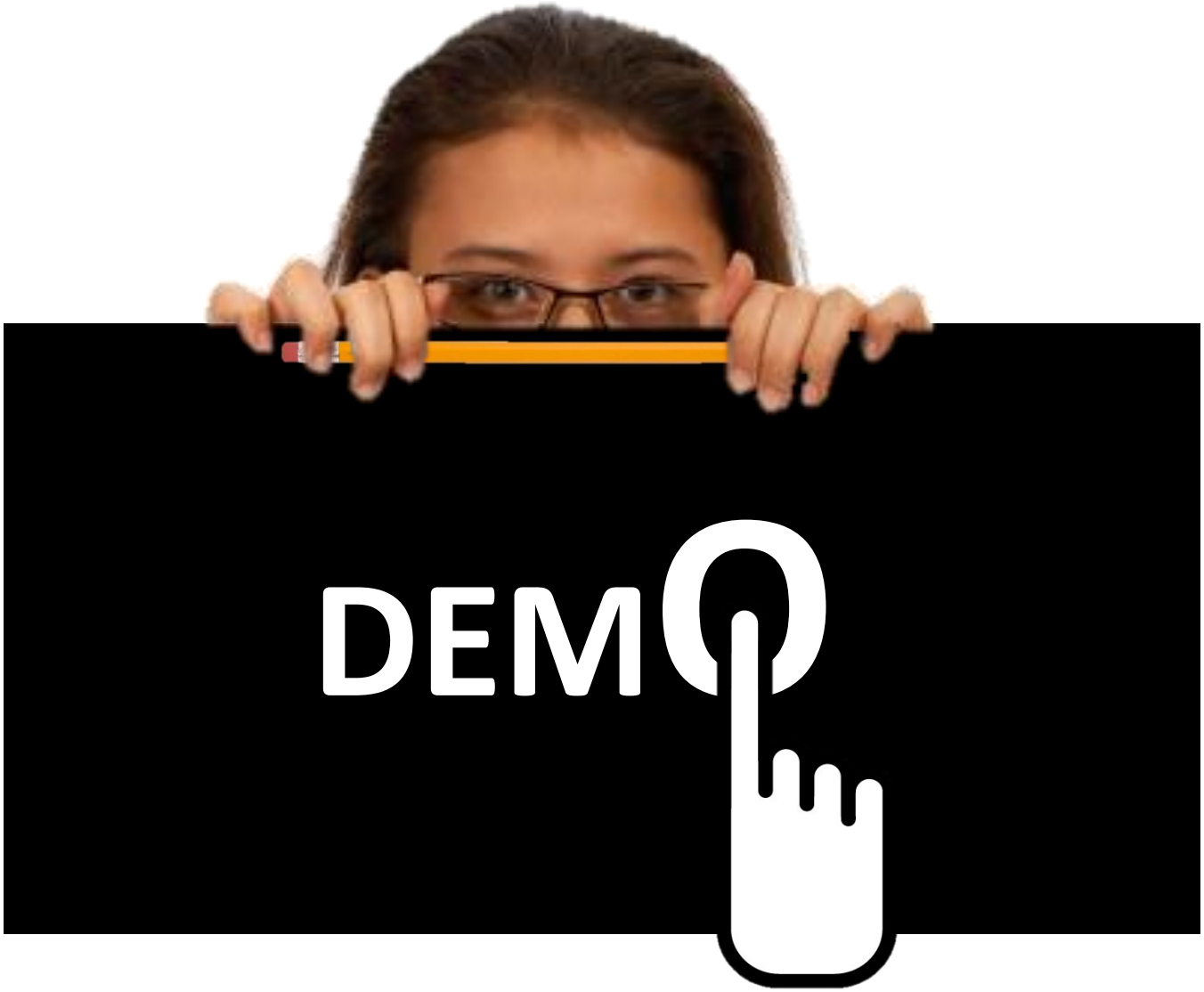
CARD LAYOUT

- Ngoài ra ta có thể dùng JTabbedPane để thay thế cho CardLayout, JTabbedPane có giao diện đẹp mắt và thân thiện với người sử dụng

CARD LAYOUT

Demo: JTabbedPane




DEMO

GRID LAYOUT

- Grid Layout bố trí các Component của Container vào bên trong một Grid với các Row và Column.
- Mỗi Component được bố trí trong 1 Cell của lưới.
- Tất cả các Cell có cùng kích thước bên trong Grid.
- Các Component được thêm vào Container theo thứ tự từ trái sang phải, từ trên xuống dưới (mặc định).

GRID LAYOUT

- Khởi tạo
  - GridLayout()
  - GridLayout(int rows, int cols)
  - GridLayout(int rows, int cols, int hgap, int vgap)

- Demo: GridLayout

GridBag Layout

- GridBag Layout bố trí các Component trong một Grid với các Row và Column.
- Mỗi Component bên trong Grid được RowSpan và ColumnSpan (giống table HTML)
- Width và Height của các Row/Column có thể khác nhau.
- GridBag Layout là một Layout Manager rất linh động cho việc bố trí các Component bên trong Container theo dạng Grid.
- GridBag Layout là một trong các Layout Manager **thường sử dụng nhất** mà Java Platform cung cấp.

GridBag Layout

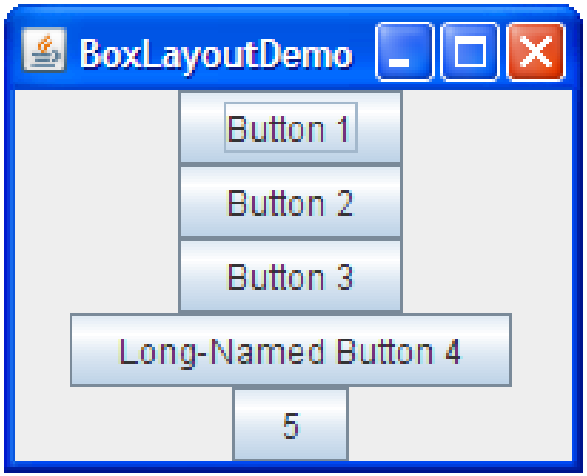
- Các thuộc tính của GridBagContraints
  - gridx ,gridy : vị trí dòng, vị trí cột
  - gridheight , gridwidth : số lượng dòng, số lượng cột
  - ipadx, ipady :
  - Insets
  - weightx, weighty
  - fill :
    - NONE, HORIZONTAL, VERTICAL, BOTH

- Các thuộc tính của GridBagContraints
  - anchor:

```
-----------------------------------------------
|FIRST_LINE_START    PAGE_START     FIRST_LINE_END|
|                                                 |
|                                                 |
|LINE_START            CENTER           LINE_END|
|                                                 |
|                                                 |
|LAST_LINE_START      PAGE_END       LAST_LINE_END|
-----------------------------------------------
```

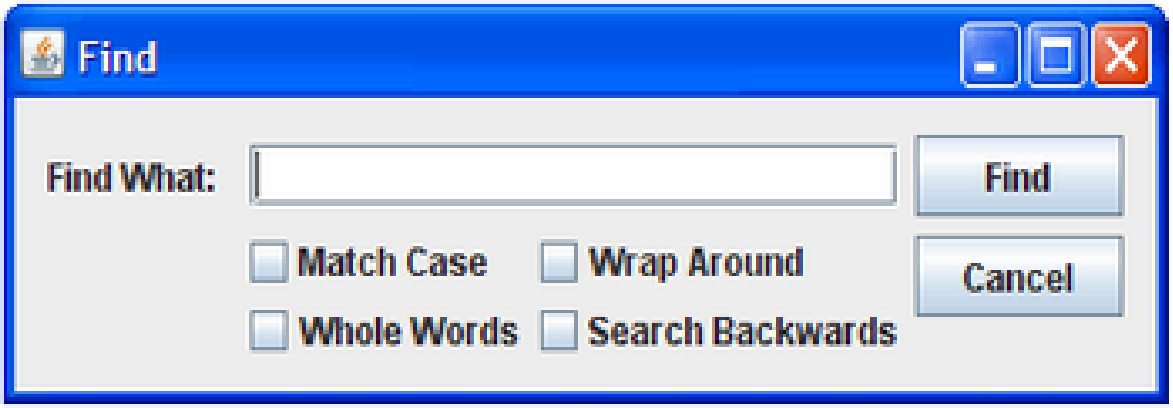
Box Layout

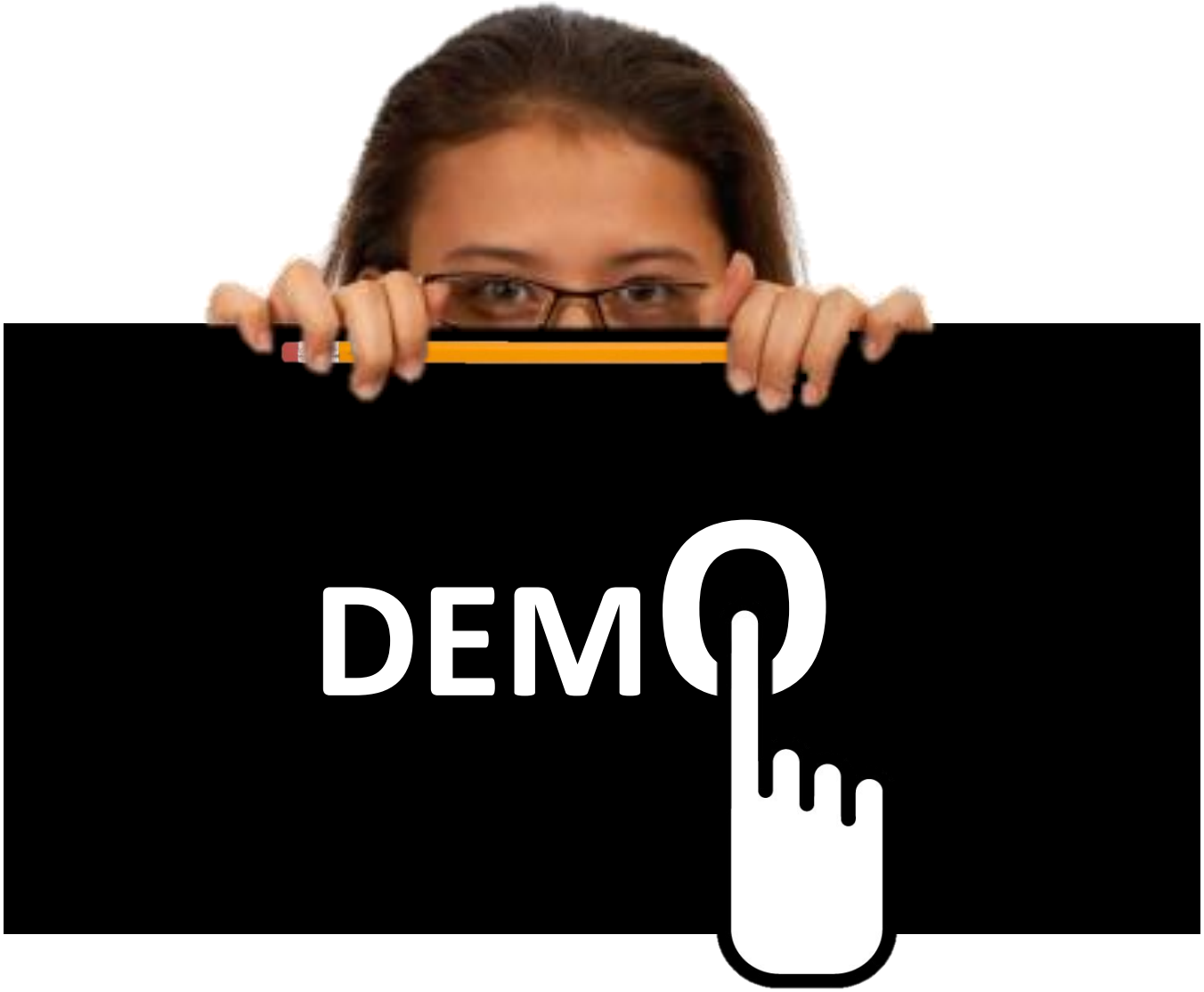
- Box Layout bố trí các Component bên trong Container theo 1 dòng theo trục X, hoặc là trục Y.
- BoxLayout(Container container, int align)
- container: chứa các Component
- axis:
- BoxLayout.X_AXIS : Trục X
- BoxLayout.Y_AXIS : Trục Y



GROUP LAYOUT

- Group Layout bố trí các Component bên trong Container theo chiều ngang và chiều dọc.
- Sự bố trí được thực hiện theo mỗi chiều riêng lẽ




DEMO

Fpt University

FPT POLYTECHNIC

LẬP TRÌNH JAVA 3

BÀI 3: LAYOUT, TIMER, JSLIDER

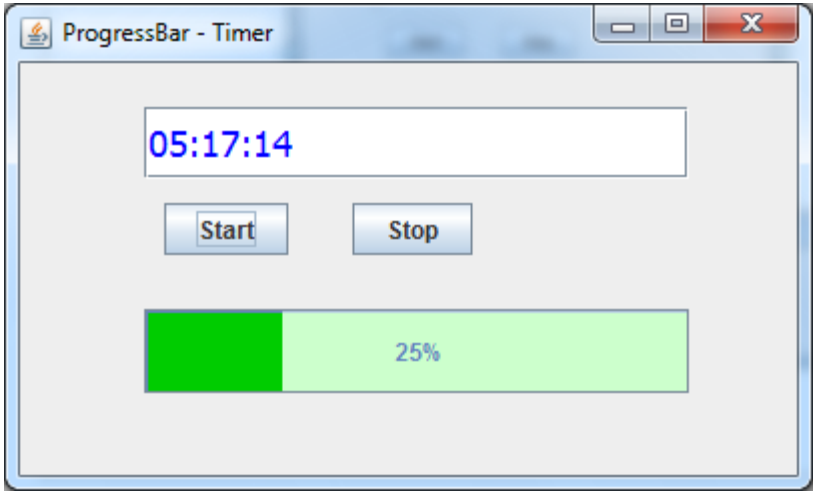
PHẦN 2

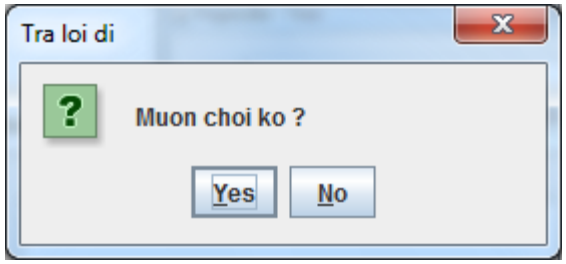
www.poly.edu.vn

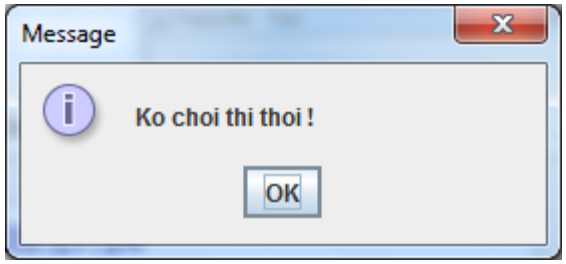
TIMER-PROGRESSBAR

- Timer dùng để kích hoạt một hoặc nhiều sự kiện ActionEvents sau một khoảng thời gian nhất định.
- Hàm tạo(constructor):
  Timer(int delay, ActionListener listener)
  Hàm tạo này sẽ tạo một sự kiện sau mỗi khoảng thời gian tính theo đơn vị mili giây

Demo


ProgressBar - Timer
05:17:14
Start
Stop
25%


Tra loi di
Muon choi ko ?
Yes
No

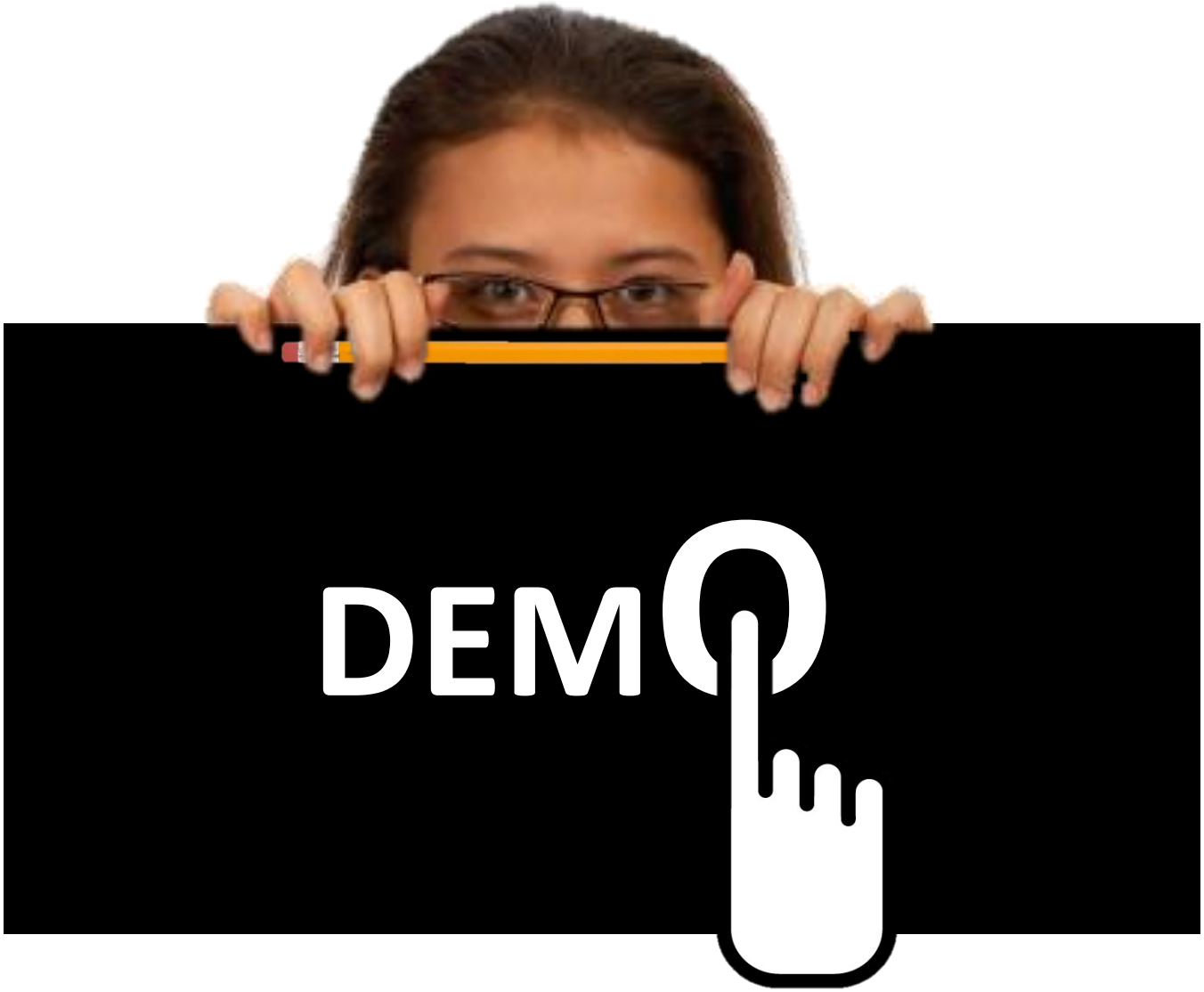

Message
Ko choi thi thoi !
OK

TIMER-PROGRESSBAR

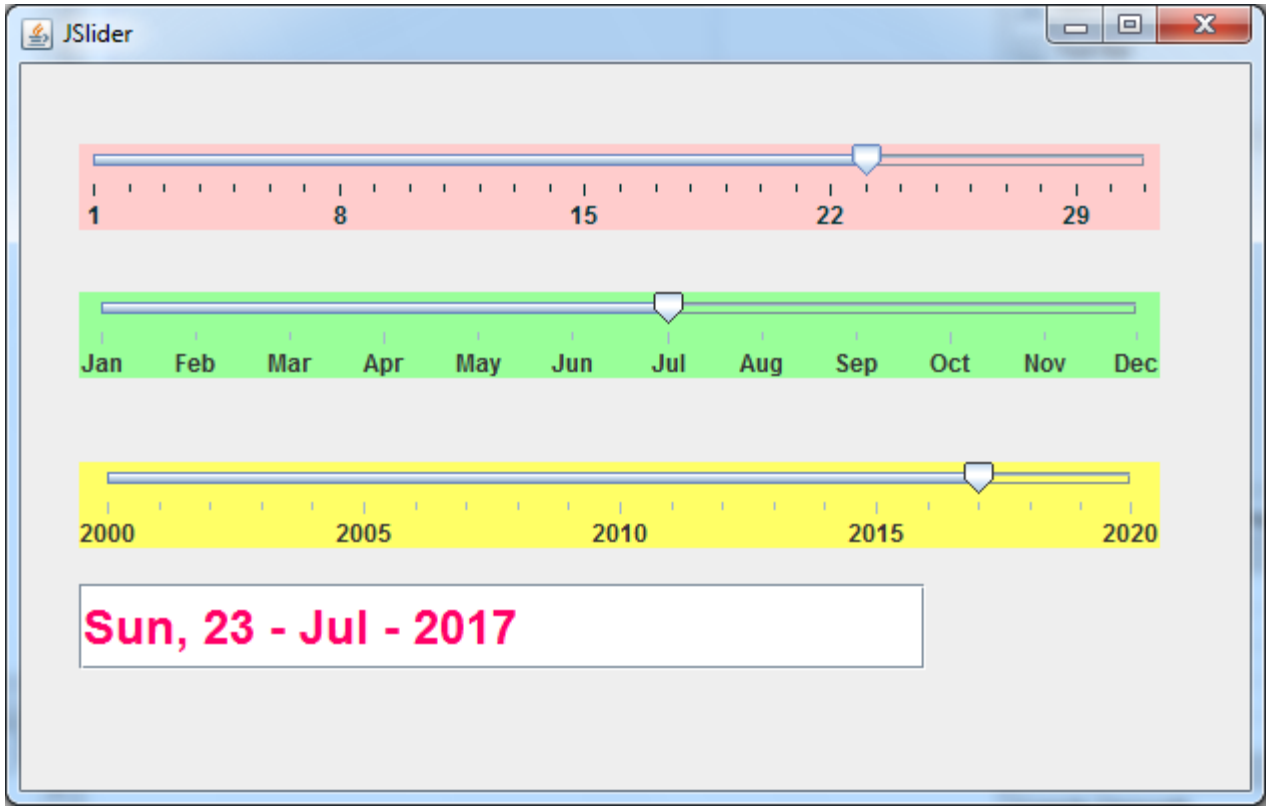
```
public class JProgressBarTimer extends javax.swing.JFrame {
    javax.swing.Timer t;
    public JProgressBarTimer() {
        initComponents();
        ActionListener a = new ActionListener() {
          public void actionPerformed(ActionEvent e) {
            SimpleDateFormat sdf = new SimpleDateFormat("hh:mm:ss");
            txtClock.setText(sdf.format(new Date()));
            int value = prgTienDo.getValue()+1;
            prgTienDo.setValue(value<=100?value:0);
          }
        };
        t = new javax.swing.Timer(1000, a);
    }
    //..........
```

```
  private void btnStartActionPerformed(java.awt.event.ActionEvent evt) {
    int i = JOptionPane.showConfirmDialog(null,"Muon choi ko ?","Tra loi di",
            JOptionPane.YES_NO_OPTION);
    if(i==0)
      t.start();
    else
      JOptionPane.showMessageDialog(null, "Ko choi thi thoi !");
  }
```

```
  private void btnStopActionPerformed(java.awt.event.ActionEvent evt) {
    t.stop();
  }
```


DEMO

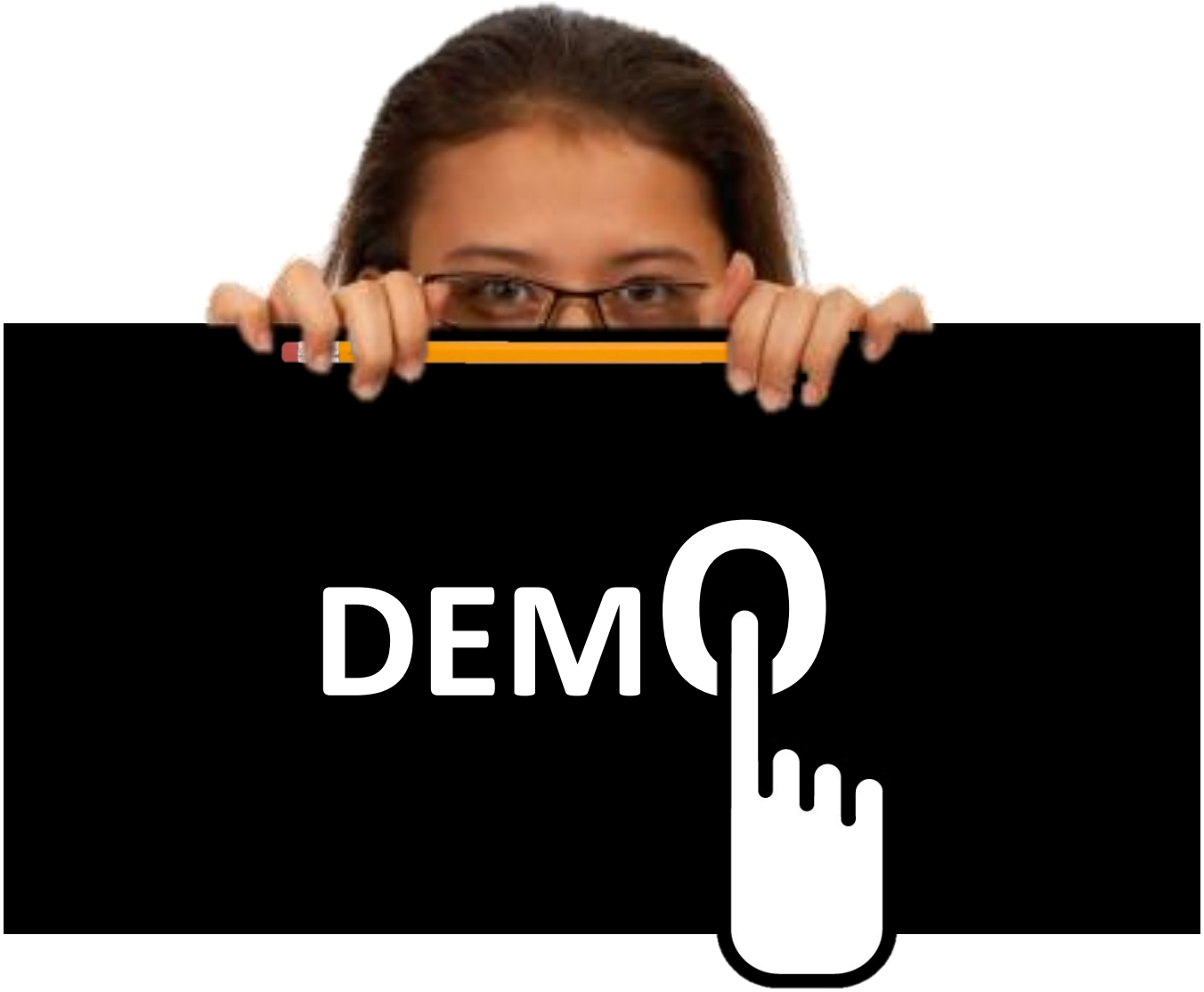
Demo JSlider



JSLIDER

```
public class JSliderDemo extends javax.swing.JFrame {
    public JSliderDemo() {
        initComponents();
        Hashtable data = new Hashtable();
        String [] sMonths= {"Jan","Feb","Mar","Apr","May","Jun","Jul","Aug",
            "Sep","Oct","Nov","Dec"};
        for(int i =0; i<12; i++){
          data.put(i, new JLabel(sMonths[i]));
        }
        sldThang.setLabelTable(data);
        Calendar can = Calendar.getInstance();
        sldNgay.setValue(can.get(Calendar.DAY_OF_MONTH));
        sldThang.setValue(can.get(Calendar.MONTH));
        sldNam.setValue(can.get(Calendar.YEAR));
        displayDate();
    }
    void displayDate(){
      SimpleDateFormat sdf = new SimpleDateFormat("EEE, dd - MMM - yyyy");
      int dd = sldNgay.getValue();
      int mm = sldThang.getValue();
      int yy = sldNam.getValue()-1900;
      Date d = new Date(yy, mm, dd);
      txtNgay.setText(sdf.format(d));
    }
    //.........

    private void sldNgayStateChanged(javax.swing.event.ChangeEvent evt) {
      displayDate();
    }
    private void sldThangStateChanged(javax.swing.event.ChangeEvent evt) {
      displayDate();
    }
    private void sldNamStateChanged(javax.swing.event.ChangeEvent evt) {
      displayDate();
    }
```


DEMO

TỔNG KẾT

- Layout Manager
  - FlowLayout
  - BorderLayout
  - CardLayout
  - TabbedPaneLayout
  - GridLayout
  - GridBagLayout
  - BoxLayout
  - GroupLayout
- Timer – ProgressBar
- JSlider


TIN HỌC VĂN PHÒNG QUỐC TẾ THEO CHUẨN ICDL
Syllabus 5 - Office 2007
LUẬT KINH TẾ
LUẬT KINH TẾ
KỸ THUẬT SOẠN THẢO VÀ TRÌNH BÀY VĂN BẢN
TIN HỌC VĂN PHÒNG QUỐC TẾ THEO CHUẨN ICDL
Syllabus 5 - Office 2007
Cảm ơn