

## **BÀI GIẢNG**

## NGÔN NGỮ LẬP TRÌNH JAVA

CHƯƠNG 10:

LẬP TRÌNH GIAO DIỆN

TS. Nguyễn Sĩ Thìn (nsthin@vku.udn.vn)

## Chương 10. Lập trình giao diện



#### Mục tiêu

Hiểu và vận dụng được các thành phần của AWT, SWING trong lập trình giao diện

Hiểu và vận dụng được phương pháp nắm bắt sự kiện

Tây dựng được một ứng dụng đồ họa đơn giản





## Nội dung



Chương 10: LẬP TRÌNH GIAO DIỆN

- 1. Giới thiệu
- 2. Các thành phần cơ bản của AWT
- 3. Quản lý giao diện
- 4. Nắm bắt sự kiện
- 5. Giới thiệu SWING



## 1. Giới thiệu lập trình giao diện



 AWT (Abstract Window Toolkit): Một tập các class (API) cho phép người dùng có thể lập trình giao diện người dùng (GUI).

#### Lưu ý:

- Nằm trong gói java.awt
- Các thành phần của AWT là phụ thuộc
  - → Cùng một thành phần như: TextField, Label, Button, nhưng ở mỗi Hệ điều hành sẽ có cách thể hiện khác nhau.
- Môi trường lập trình: Eclipse, Netbeans
- Cung cấp nhiều thành phần hỗ trợ lập trình giao diện như:
  - Containers

Graphics and drawing capabilities

Components

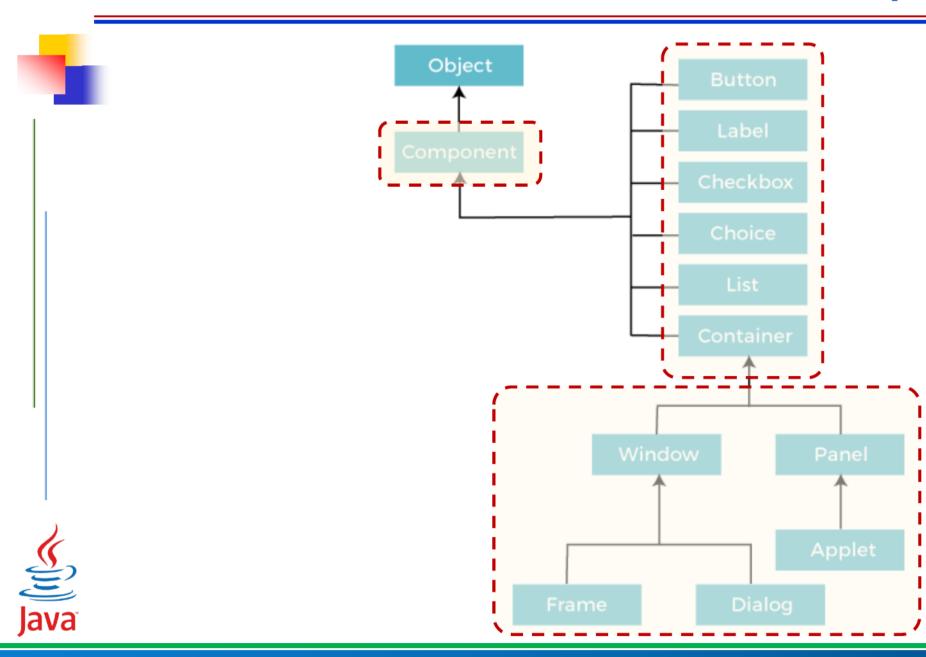
Fonts

Layout managers

Events



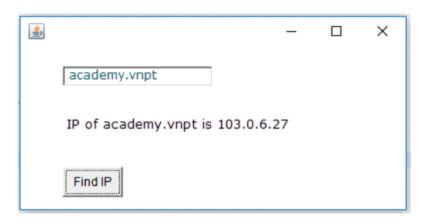
# Biểu đồ các lớp kế thừa của AWT

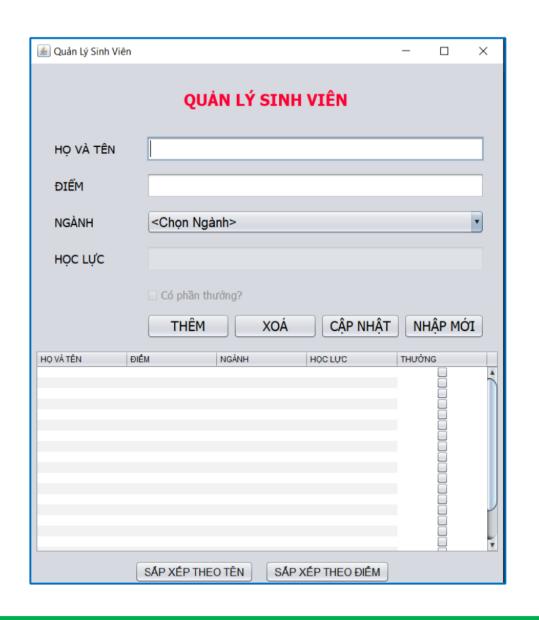


## 2. Các thành phần cơ bản của AWT



- Button
- Lable
- TextField
- Checkbox, Choice, List, Scrollbar, Panel, Diaglog







### **Button**



- Button: tạo ra một đối tượng nút (có thể) gán nhãn và sẽ đưa ra một sự kiện khi được nhấn vào.
- Hàm dựng:

TT	Constructor	Description
1.	Button()	Tạo ra một nút với nhãn trống
2.	Button (String text)	Tạo ra một nút với nhãn cho trước

Ví dụ:





#### Lable

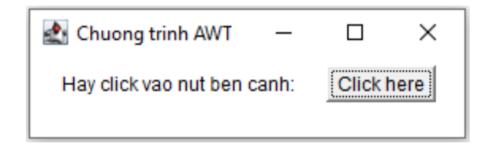


- Lable: tạo ra một đối tượng nhãn và đặt vào trong một đối tượng Containner
- Lưu ý:
  - Nhãn chỉ đọc khi thể hiện
  - Không tạo ra sự kiện khi bị tác động

#### Hàm dựng:

TT	Constructor	Description
1.	Label()	Tạo ra một nhãn trống
2.	Label(String text)	Tạo ra một nhãn với tên cho trước
3.	Label(String text, int alignement)	Tạo ra một nhãn với tên và vị trí căn lề cho trước

#### o Ví dụ:





## **TextField**



 TextField: tạo ra một đối tượng cho phép người dùng nhập vào một dòng và có thể chỉnh sửa.

#### Lưu ý:

- Nhãn chỉ đọc khi thể hiện
- Không tạo ra sự kiện khi bị tác động

#### Hàm dựng:

TT	Constructor	Description
1.	TextField ()	Tạo ô nhập trống
2.	TextField (String text)	Tạo ô nhập với nội dung cho trước
3.	TextField (int columns)	Tạo ô nhập trống với độ rộng hiển thị cho trước
4.	TextField (String text, int columns)	Tạo ô nhập với nội dung và độ rộng hiển thị cho trước



## Checkbox, Choice, List, Scrollbar, Panel, Diaglog



#### Một số thành phần cơ bản khác

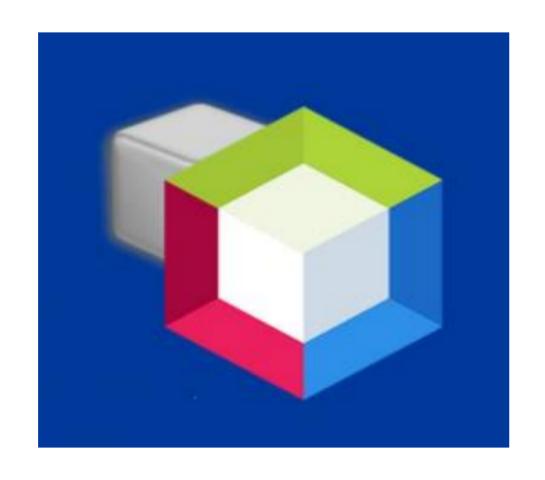
TT	Class	Nội dung
1.	Checkbox	Tạo ra đối tượng Checkbox
2.	Choice	Tạo ra một ô lựa chọn với danh sách nội dung có sẵn
3.	List	Tạo ra một list danh sách lựa chọn cho người dùng
4.	Scrollbar	Tạo ra một thanh kéo (trượt ngang hoặc dọc)
5.	Panel	Cung cấp một không gian để chứa các thành phần khác (Không có đường biên và title)
6.	Diaglog	Tạo ra cửa sổ có đường biên và tiêu đề



## 3. Quản lý giao diện



- Khái niệm: Quy định cách thức sắp xếp vị trí các thành phần trên giao diện.
- Gồm các class:
  - java.awt.BorderLayout
  - java.awt.FlowLayout
  - java.awt.GridLayout
  - java.awt.CardLayout





## **Border Layout**



- Ý nghĩa: Sắp xếp các thành phần theo 5 vùng: Bắc (North), Nam (South),
  Đông (East) và Tây (West)
- Lưu ý:
  - Là quản lý giao diện mặc định
  - Mỗi vùng chỉ chứa một đối tượng
- Hàm dựng:

TT	Constructor	Description
1.	BorderLayout ()	Tạo đối tượng BorderLayout không có khoảng cách giữa các vùng.
2.	BorderLayout (int hgap, int vgap)	Tạo đối tượng BorderLayout có khoảng cách giữa các vùng theo hàng và cột.



Ví dụ sử dụng Border Layout:

## **Grid Layout**



- Ý nghĩa: Sắp xếp các thành phần theo các hàng và cột cho trước
- Lưu ý:
  - · Chèn lần lượt từ trái qua phải, trên xuống dưới
  - Mỗi vùng chỉ chứa một đối tượng
- Hàm dựng:

тт	Constructor	Description
1.	GridLayout()	Tạo đối tượng GridLayout gồm 1 hàng, 1 cột
2.	GridLayout(int rows, int columns)	Tạo đối tượng GridLayout có số hàng và cột cho trước
3.	GridLayout(int rows, int columns, int hgap, int vgap)	Tạo đối tượng GridLayout có số hàng, số cột, khoảng cách giữa các hàng và cột cho trước.



## **Flow Layout**



- Ý nghĩa: Sắp xếp các thành phần theo dòng từ trái sang phải
- Lưu ý:
  - Là Layout mặc định cho Panel
- Hàm dựng:

TT	Constructor	Mô tả
1.	FlowLayout()	Tạo đối tượng FlowLayout với các đối tượng sắp xếp căn giữa và khoảng cách theo hàng và cột là 5.
2.	FlowLayout(int align)	Tạo đối tượng FlowLayout với các đối tượng sắp xếp căn vị trí cho trước và khoảng cách theo hàng và cột là 5.
3.	FlowLayout(int align, int hgap, int vgap)	Tạo đối tượng FlowLayout với các đối tượng sắp xếp căn vị trí và khoảng cách theo hàng, cột cho trước.



## **Card Layout**



- Ý nghĩa: Sắp xếp các thành phần chồng lên nhau, mỗi lần một thành phần được hiển thị
- Lưu ý:
  - Thành phần nào được chèn trước thì sẽ hiển thị trước
  - Gọi thành phần tiếp theo bằng phương thức next(Container parent);

#### Hàm dựng:

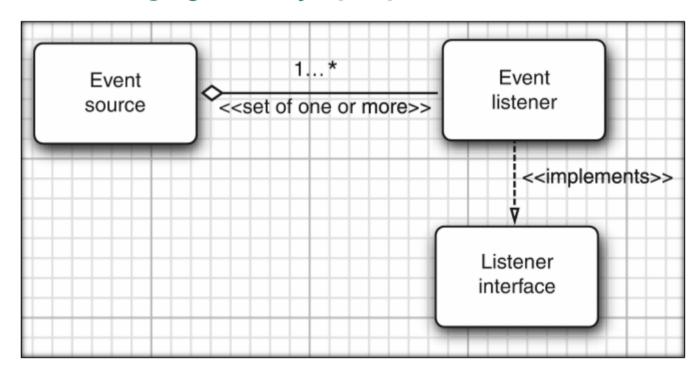
TT	Constructor	Mô tả
1.	CardLayout ()	Tạo đối tượng CardLayout với khoảng cách hàng, cột bằng 0
2.	CardLayout (int hgap, int vgap)	Tạo đối tượng CardLayout với khoảng cách hàng, cột cho trước



## 4. Xử lý sự kiện



- Sự kiện: Thay đổi trạng thái của một đối tượng được xem như là một sự kiện (Click chuột, rê chuột...lên một nút).
- Cách thức xử lý sự kiện:
  - Đối tượng sự kiện: java.awt.event
  - Lắng nghe, xử lý sự kiện: Listener Interfaces



 Ví dụ 3: Viết ứng dụng tính: tổng, hiệu, tích, thương của hai số.



#### 5. SWING



- SWING: là một phần của Java Foundation Classess (JFC) được sử dụng để lập trình các ứng dụng window.
- Lưu ý:
  - Phát triển bổ sung (KHÔNG thay thế) AWT
  - Các thành phần của SWING là độc lập
  - Nằm trong gói javax.swing gồm các: JTextField, JLabel, Jbutton...
  - Chứa nhiều thành phần hơn AWT
- Ví dụ: Viết ứng dụng gồm 1 Jbutton và 1 JTextField, JTextField có giá trị ban đầu bằng 0, khi kích chuột vào Jbutton thì giá trị tai JTextField tăng lên 1.



## Bài tập thực hành cuối chương 10



- Đề bài: Viết chương trình tạo một máy tính điện tử đơn giản gồm 4 phép toán cộng, trừ, nhân, chia.
- Gợi ý:
  - Sử dụng các button đại diện cho các nút từ 1 đến 9;
  - Sử dụng các button đại diện cho 4 phép toán
  - Sử dụng một TextField để hiển thị kết quả và không cho cho phép chỉnh sửa
  - Gắn ống nghe cho tất cả các nút



# THANK YOU!