

## SAMPLE PROJECT

### BÀI 4: THỰC HIỆN DỰ ÁN 2

- ⊙ Kết thúc bài học này bạn có khả năng
  - ⊙ Xây dựng các thư viện tiện ích cho dự án
  - ⊙ Tổ chức mã điều khiển sự kiện cho trang chính
  - ⊙ Thực hiện chức năng đăng nhập

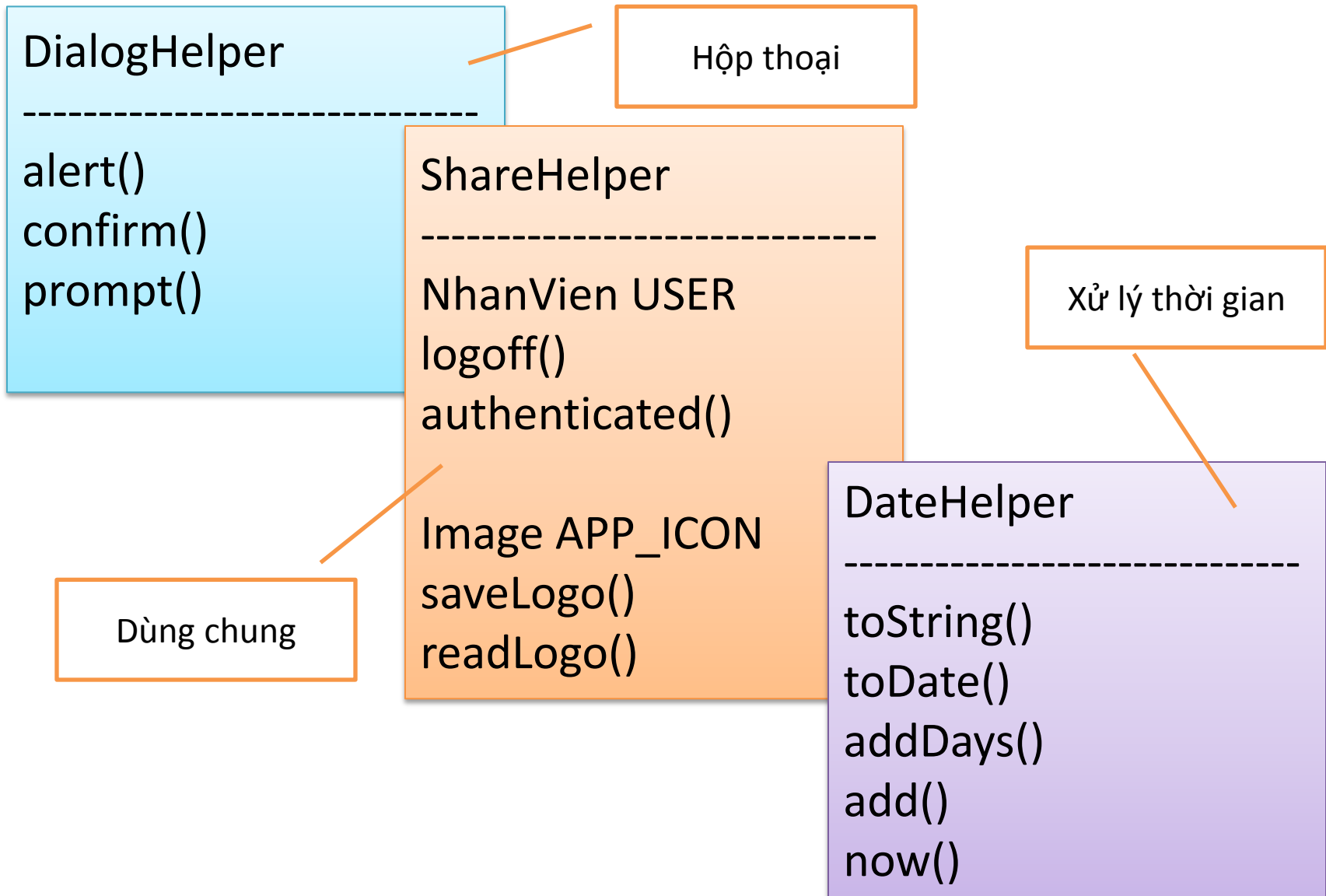


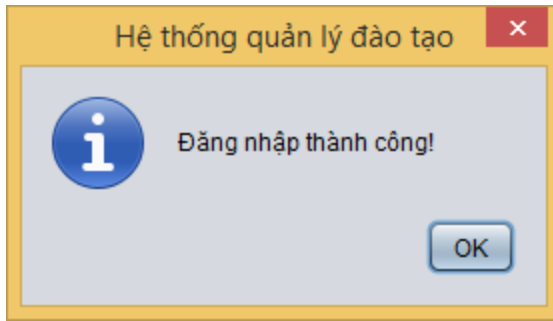
- 📖 Giới thiệu thư viện tiện ích
- 📖 Xây dựng thư viện tiện ích
- 📖 Phân tích các chức năng trang chính
- 📖 Tổ chức mã cho trang chính
- 📖 Tổ chức và viết mã cho các form đơn giản
- 📖 Mở trang hướng dẫn trên trình duyệt
- 📖 Tổ chức viết mã cho form đăng nhập



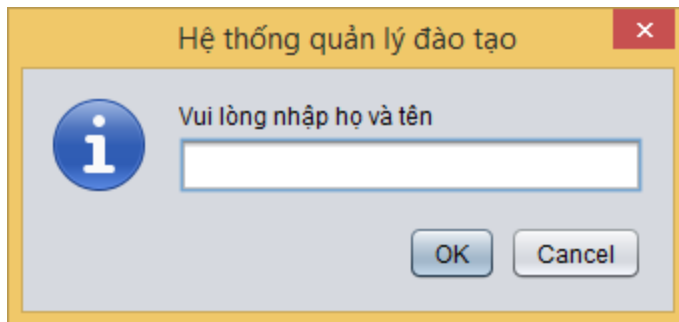
Hướng dẫn xây dựng  
các thư viện cần thiết  
để sử dụng trong dự án

- ❑ Thư viện tiện ích đóng vai trò rất quan trọng. Nó thể hiện tính chuyên nghiệp của người làm dự án. Cụ thể
  - ❖ Để rút ngắn mã nguồn
  - ❖ Dễ quản lý và nâng cấp
  - ❖ Dễ kiểm lỗi
- ❑ Trong dự án này bạn cần xây dựng 4 lớp chứa các hàm tiện ích sau
  - ❖ JdbcHelper: hỗ trợ làm việc với CSDL (đã giới thiệu)
  - ❖ DateHelper: hỗ trợ xử lý thời gian
  - ❖ DialogHelper: hỗ trợ hiển thị hộp thoại đơn giản
  - ❖ ShareHelper: hỗ trợ quản lý thông tin người dùng và tài nguyên

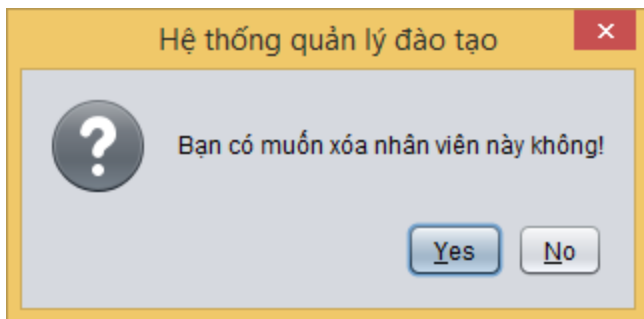




DialogHelper.**alert**(this,  
"Đăng nhập thành công!");



DialogHelper.**prompt**(this,  
"Vui lòng nhập họ và tên");



DialogHelper.**confirm**(this,  
"Bạn có muốn xóa nhân viên này không!");

## ❑ Chuyển đổi Date sang String

❖ String s = DateHelper.**toString**(Date, String)

➤ String: định dạng thời gian

## ❑ Chuyển String sang Date

❖ Date d= DateHelper.**toDate**(String, String)

➤ String: định dạng thời gian

## ❑ Lấy thời gian hiện tại

❖ Date d= DateHelper.**now**()

## ❑ Thêm số ngày vào thời gian hiện tại

❖ Date d= DateHelper.**add**(int)

➤ Int: số ngày

## ❑ Thêm số ngày vào thời gian

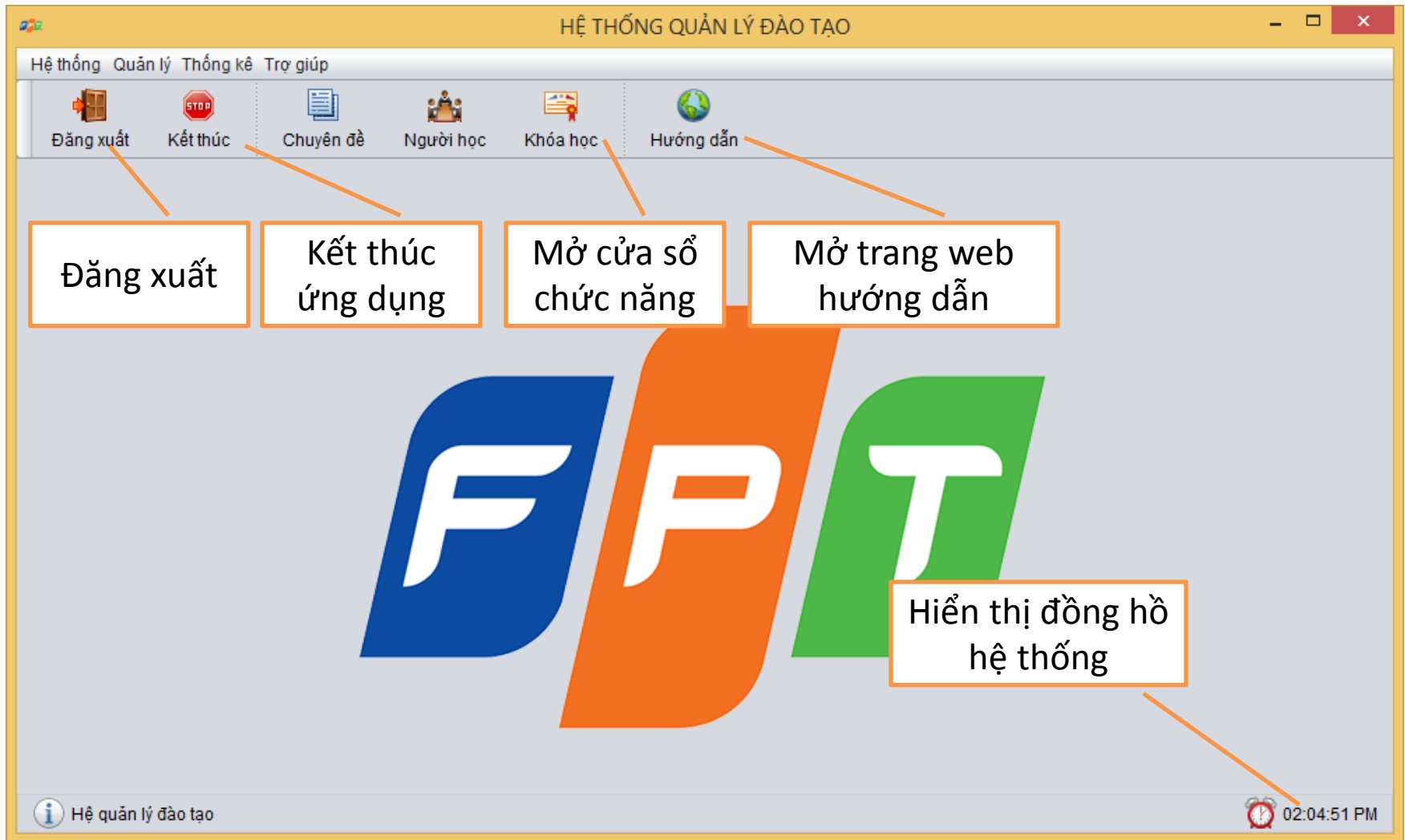
❖ Date d= DateHelper.**addDays**(Date, int)

➤ Int: số ngày



- ❑ Biểu tượng của ứng dụng
  - ❖ Image ShareHelper.**APP\_ICON**
- ❑ Lưu ảnh logo chuyên đề
  - ❖ ShareHelper.**saveLogo**(File file)
- ❑ Đọc ảnh logo chuyên đề
  - ❖ ImageIcon ShareHelper.**readLogo**(String)
- ❑ Thông tin nhân viên đăng nhập
  - ❖ NhanVien ShareHelper.**USER**
- ❑ Xóa thông tin đăng nhập
  - ❖ ShareHelper.**logoff**()
- ❑ Kiểm tra đăng nhập hay chưa
  - ❖ Boolean ShareHelper.**authenticated**()

# Hướng dẫn lập trình giao diện



```
void init() { ...16 lines }
```

```
void openLogin() { ...3 lines }
```

```
void openWelcome() { ...3 lines }
```

```
void logoff() { ...4 lines }
```

```
void exit() { ...5 lines }
```

```
void openThongKe(int index) { ...8 lines }
```

```
void openNhanVien() { ...8 lines }
```

```
void openKhoaHoc() { ...8 lines }
```

```
void openChuyenDe() { ...8 lines }
```

```
void openNguoiHoc() { ...8 lines }
```

```
void openAbout() { ...3 lines }
```

```
void openWebsite() { ...8 lines }
```

- ☐ Constructor
  - ❖ Init()
- ☐ [Đăng nhập]
  - ❖ openLogin()
- ☐ [Đăng xuất]
  - ❖ Logoff()
- ☐ [Kết thúc]
  - ❖ Exit()
- ☐ [Thống kê i]
  - ❖ openThongKe(0->3)
- ☐ [Nhân viên]
  - ❖ openNhanVien()
- ☐ [Khóa học]
  - ❖ openKhoaHoc()
- ☐ [Chuyên đề]
  - ❖ openChuyenDe()
- ☐ [Người học]
  - ❖ openNguoiHoc()
- ☐ [Hướng dẫn]
  - ❖ openWebsite()

```
void init() {
    setSize(1000, 600);
    setIconImage(ShareHelper.APP_ICON);
    setLocationRelativeTo(null);

    new Timer(1000, new ActionListener() {
        SimpleDateFormat format = new SimpleDateFormat("hh:mm:ss a");
        @Override
        public void actionPerformed(ActionEvent e) {
            lblDongHo.setText(format.format(new Date()));
        }
    }).start();

    this.openWelcome();
    this.openLogin();
}
```

```
void openLogin() {  
    new DangNhapJDialog(this, true).setVisible(true);  
}  
  
void openWelcome() {  
    new ChaoJDialog(this, true).setVisible(true);  
}  
  
void logoff() {  
    ShareHelper.logoff();  
    this.openLogin();  
}  
  
void exit() {  
    if(DialogHelper.confirm(this, "Bạn thực sự muốn kết thúc?")) {  
        System.exit(0);  
    }  
}
```

```
void openThongKe(int index){  
    if(ShareHelper.authenticated()){  
        new ThongKeJFrame(index).setVisible(true);  
    }  
    else{  
        DialogHelper.alert(this, "Vui lòng đăng nhập!");  
    }  
}
```

Tương tự

```
void openNhanVien() {  
    if(ShareHelper.authenticated()){  
        new NhanVienJFrame().setVisible(true);  
    }  
    else{  
        DialogHelper.alert(this, "Vui lòng đăng nhập!");  
    }  
}  
void openKhoaHoc() { ...8 lines }  
void openChuyenDe() { ...8 lines }  
void openNguoiHoc() { ...8 lines }
```

- ❑ Theo yêu cầu security thì form thống kê chỉ cho phép mở khi người đăng nhập là trưởng phòng.  
Vì vậy điều kiện để mở form này
  - ❖ Phải đăng nhập (`ShareHelper.authenticated()` phải có giá trị là `true`)
  - ❖ Vai trò phải là trưởng phòng  
(`ShareHelper.USER.getVaiTro()` phải có giá trị là `true`)
- ❑ Hãy chỉnh lại mã `openThongKe()` sao cho phù hợp

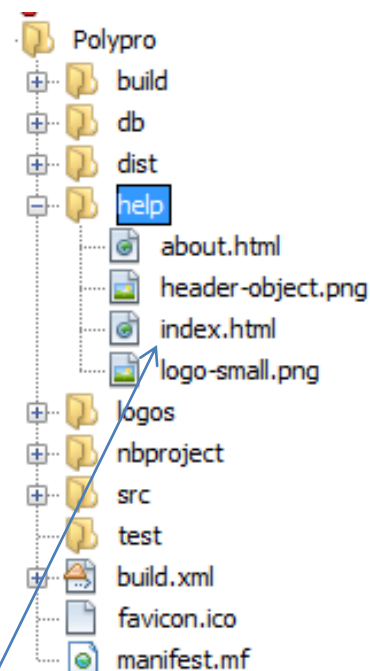


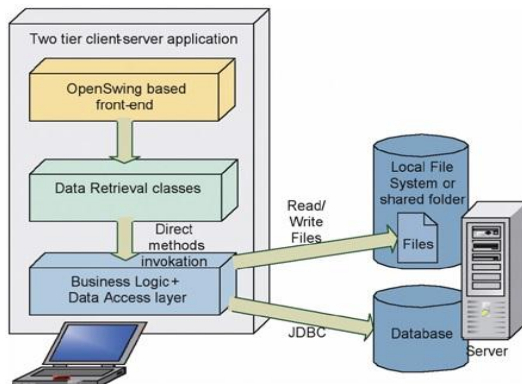
## Mở trang web

```

void openAbout() {
    new GioiThieuJDialog(this, true).setVisible(true);
}

void openWebsite() {
    try {
        Desktop.getDesktop().browse(new File("help/index.html").toURI());
    }
    catch (IOException ex) {
        DialogHelper.alert(this, "Không tìm thấy file hướng dẫn!");
    }
}
  
```





## SAMPLE PROJECT

### BÀI 4: THỰC HIỆN DỰ ÁN 2



Tăng đến max  
và đóng cửa sổ

```
void init() {  
    setLocationRelativeTo(null);  
    new Timer(20, new ActionListener() {  
        @Override  
        public void actionPerformed(ActionEvent e) {  
            int value = progressBar.getValue();  
            if(value < 100){  
                progressBar.setValue(value + 1);  
            }  
            else{  
                ChaoJDialog.this.dispose();  
            }  
        }  
    }).start();  
}
```

❑ Gọi init() trong constructor của của số



Click để đóng

Polypro là dự án mẫu. Mục tiêu chính là huấn luyện sinh viên qui trình thực hiện dự án.

Mục tiêu của dự án này là để rèn luyện kỹ năng IO (CDIO) tức không yêu cầu sinh viên phải thu thập phân tích mà chỉ thực hiện và vận hành một phần mềm chuẩn bị cho các dự án sau này. Các kỹ năng CD (trong CDIO) sẽ được huấn luyện ở dự án 1 và dự án 2.

Yêu cầu về môi trường:

1. Hệ điều hành bất kỳ
2. JDK 1.8 trở lên
3. SQL Server 2008 trở lên

## ❑ Constructor

- ❖ setLocationRelativeTo(null);

## ❑ Nhấp label chứa logo

- ❖ this.dispose();

HỆ THỐNG QUẢN LÝ ĐÀO TẠO

**ĐĂNG NHẬP**

Tên đăng nhập

TeoNV

Mật khẩu

\*\*\*\*\*

Đăng nhập Kết thúc

Thực hiện  
đăng nhập

Kết thúc ứng  
dụng

- ❑ Bổ sung khai báo sau vào cuối cửa sổ đăng nhập
- ❑ Gọi các phương thức trong các điều khiển sự kiện phù hợp
  - ❖ Constructor: `this.init()`
  - ❖ [Đăng nhập]: `this.login()`
  - ❖ [Kết thúc]: `this.exit()`;

```
NhanVienDAO dao = new NhanVienDAO();
```

```
void init() { ...3 lines }
```

```
void login() { ...26 lines }
```

```
void exit() { ...5 lines }
```



## ❑ Khái báo

```
NhanVienDAO dao = new NhanVienDAO()
```

➤ Dùng để làm việc với bảng *NhanVien*

## ❑ init()

```
setLocationRelativeTo(null);
```

➤ Đặt cửa sổ giữa màn hình

## ❑ exit()

```
if(DialogHelper.confirm(this,  
    "Bạn có muốn thoát khỏi ứng dụng không?")){  
    System.exit(0);  
}
```

➤ Kết thúc ứng dụng

## □ login()

```
String manv = txtMaNV.getText();
String matKhou = new String(txtMatKhou.getPassword());
try {
    NhanVien nhanVien = dao.findById(manv);
    if(nhanVien != null){
        String matKhou2 = nhanVien.getMatKhou();
        if(matKhou.equals(matKhou2)){
            ShareHelper.USER = nhanVien;
            DialogHelper.alert(this, "Đăng nhập thành công!");
            this.dispose();
        }
        else{
            DialogHelper.alert(this, "Sai mật khẩu!");
        }
    }
    else{
        DialogHelper.alert(this, "Sai tên đăng nhập!");
    }
}
catch (Exception e) {
    DialogHelper.alert(this, "Lỗi truy vấn dữ liệu!");
}
```

- ☑ Thư viện tiện ích JdbcHelper
- ☑ Mở form ChaoJDialog, DangNhapJDialog từ của sổ MainJFrame
- ☑ Mở một trang web
- ☑ Tổ chức viết mã cho form đăng nhập





**Cảm ơn**