

LAB 02 - LUỒNG NHẬP XUẤT

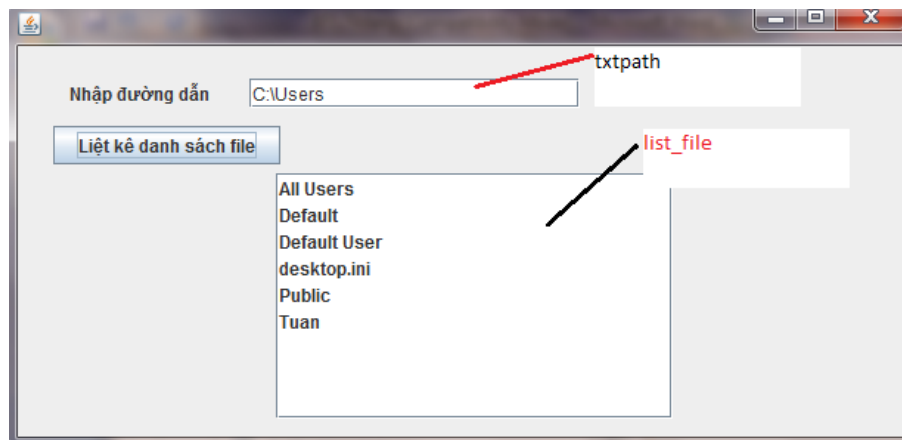


Bài 1. Viết chương trình nhập vào tên đường dẫn của một thư mục. Hiển thị tên tất các tập tin trong thư mục lên màn hình (sử dụng phương thức `listFiles` của lớp đối tượng `File` và lớp `FileFilter`).

Hướng dẫn: Thao tác với `File` là một phần không thể thiếu trong các ứng dụng mạng, nhất là các ứng dụng truyền nhận dữ liệu.

1. Chương trình đơn giản, không có chức năng lọc danh sách.

Bước 1: Tạo Project mới → Tạo `JFrameForm` có giao diện như sau:



`TextField` để nhập đường dẫn.

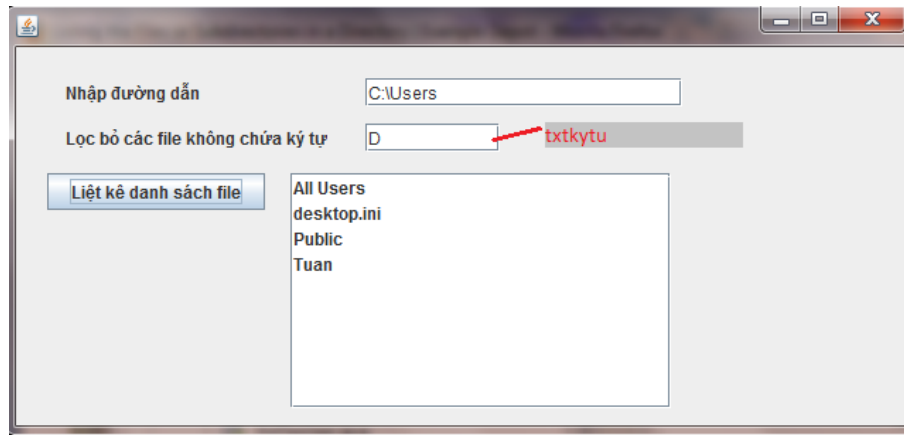
`Listbox` hiển thị tất cả các tập tin và thư mục có trong đường dẫn đến thư mục.

Bước 2: Thêm sự kiện cho button `Liệt kê danh sách file`

```
private void btnlietkeActionPerformed(java.awt.event.ActionEvent evt) {  
    File dir=new File(txtpath.getText());  
    File dsFile[]=dir.listFiles();  
    if(dsFile==null)  
    {  
        JOptionPane.showMessageDialog(null, "sai duong dan!");  
    }else{  
        try{  
            DefaultListModel dm=new DefaultListModel();  
            for(int i=0;i<dsFile.length;i++){  
                String filename=dsFile[i].getName();  
                dm.addElement(filename);  
            }  
            list_file.setModel(dm);  
        }catch(Exception e)  
        {JOptionPane.showMessageDialog(null,e.toString());}  
    }  
}
```

2. Thêm chức năng lọc tên file cho chương trình:

Bước 1: Thêm TextField để nhận ký tự lọc vào giao diện:

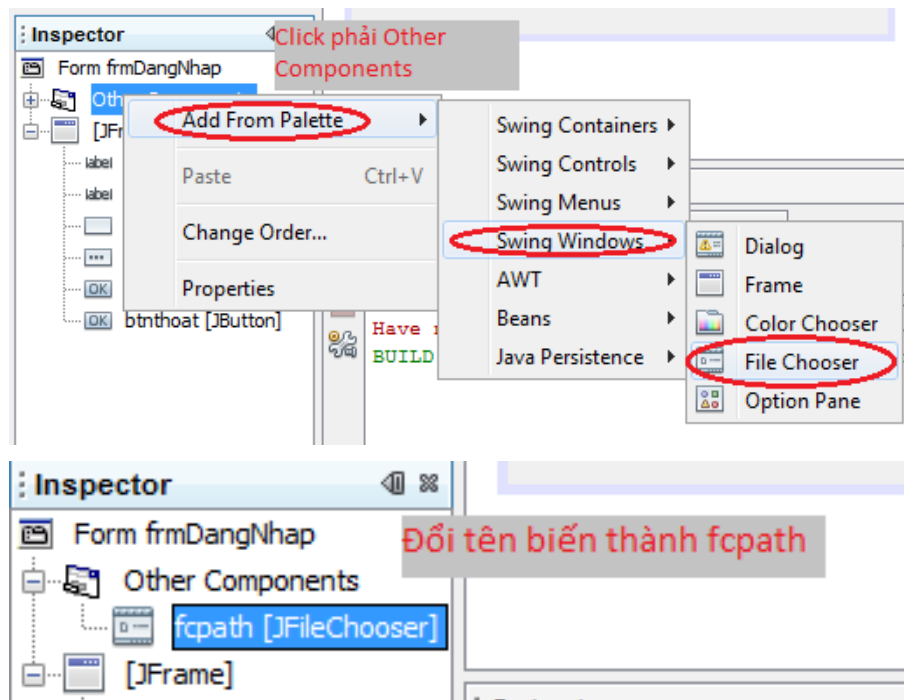


Bước 2: Sửa code cho button Liệt kê danh sách file:

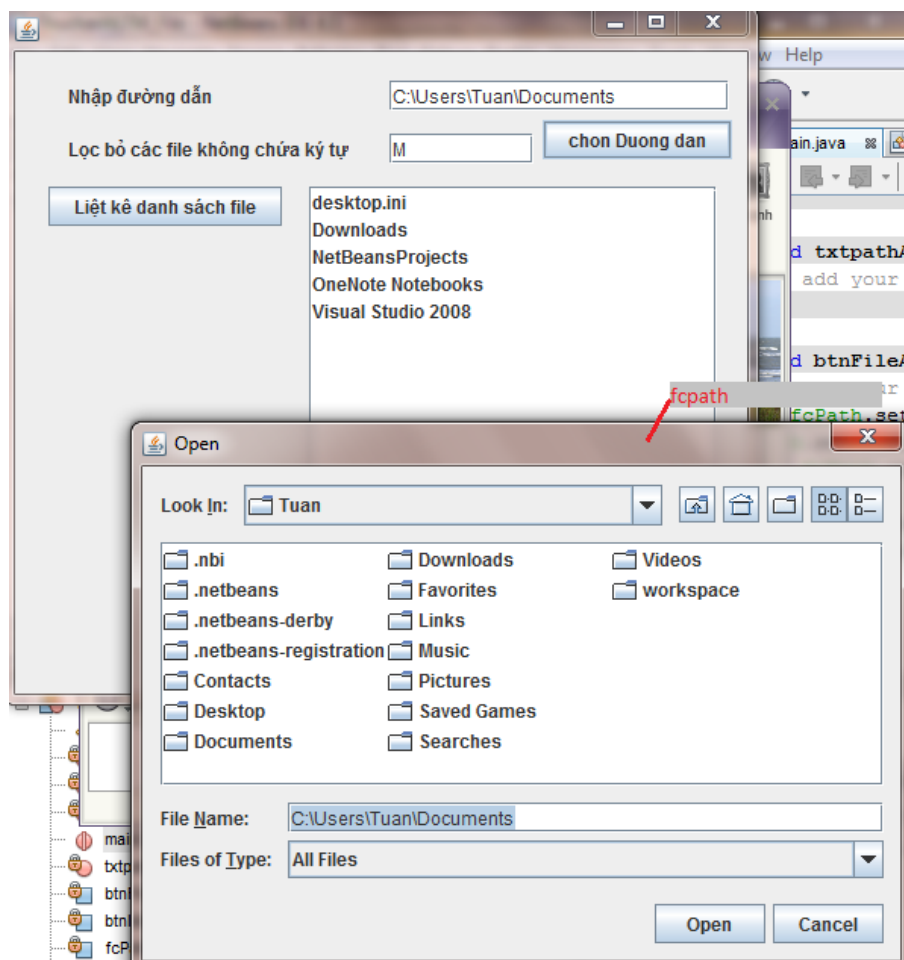
```
File dir = new File(txtpath.getText());
File dsFile[] = dir.listFiles();
if(dsFile == null){
    JOptionPane.showMessageDialog(null, "Sai duong dan!");
} else {
    try{
        DefaultListModel dm = new DefaultListModel();
        for (int i = 0; i < dsFile.length;i++) {
            String filename = dsFile[i].getName();
            if(filename.contains(txtKyTu.getText())){
                dm.addElement(filename);
            }
        }
        list_file.setModel(dm);
    }catch(Exception e){
        JOptionPane.showMessageDialog(null, e.toString());
    }
}
```

3. Thêm chức năng nhập đường dẫn tiện lợi hơn:

Bước 1: Thêm JFileChooser vào Form:



Bước 2: Chỉnh sửa lại giao diện như sau



Bước 3: Thêm sự kiện cho button Chọn đường dẫn:

```
private void btnFileActionPerformed(java.awt.event.ActionEvent evt) {  
    //thiết lập hiển thị hộp thoại chọn đường dẫn  
    this.fcPath.setVisible(true);  
    //thiết lập chế độ chọn tập tin hay thư mục  
    fcPath.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);  
    //cho hộp thoại hiển thị lên  
    if(this.fcPath.showOpenDialog(this)==JFileChooser.APPROVE_OPTION)  
    {  
        //nếu nút open được chọn  
        try{  
            txtpath.setText(fcPath.getSelectedFile().getCanonicalPath());  
        }catch(IOException e)  
        { JOptionPane.showMessageDialog(this,e); }  
    }  
}
```

Bài 2. Viết chương trình nhập vào tên một tập tin.

2.1. Viết một dãy số ngẫu nhiên n phần tử vào tập tin này.

2.2. Sau đó, đọc nội dung tập tin này hiển thị lên màn hình. Yêu cầu: thực hiện ở hai chế độ (đọc/viết nhị phân và đọc viết văn bản) và có sử dụng các luồng đọc/viết có định kiểu và luồng đệm dữ liệu.

Hướng dẫn: Trong bài trên, chúng ta đã làm quen với thao tác trên tên file, trong phần này, chúng ta sẽ cùng đọc và ghi file.

Bước 1: Tạo Project mới → Tạo giao diện như hình sau.

The screenshot shows a Java Swing window with a title bar containing a maximize button, a close button, and a standard icon. The window has a light gray background. It contains three labels with corresponding text fields: 'Nhập tên tập tin:' with the value 'E:\Test.txt', 'Nhập chuỗi' with the value '1234567890', and 'Nội dung file' with the value '1234567890'. Below these fields are four buttons arranged in a 2x2 grid: 'Đọc File nhị phân', 'Ghi File nhị phân', 'Đọc File văn bản', and 'Ghi File văn bản'. All buttons have a blue gradient and white text.

Bước 2: Xử lý sự kiện cho button Ghi File nhị phân:

```
private void btVietFileNhiPhanActionPerformed(java.awt.event.ActionEvent evt) {  
    byte a[]=new byte[20];  
    File file;  
    int i;  
    char s[]=txtNhapChuoi.getText().toCharArray();  
    for(i=0;i<s.length;i++){  
        a[i]=(byte)s[i];  
    }  
  
    try {  
        file = new File(txtFileName.getText());  
        FileOutputStream fo=new FileOutputStream(file);  
        fo.write(a);  
        fo.close();  
    } catch (IOException e) {e.printStackTrace();}  
}
```

Bước 3: Xử lý sự kiện cho button Đọc File nhị phân:

```
private void btDocFileNhiPhanActionPerformed(java.awt.event.ActionEvent evt) {  
    byte a[]; //new byte[20];  
    File file;  
    try{  
        file=new File(txtFileName.getText());  
        FileInputStream fi=new FileInputStream(file);  
        a=new byte[fi.available()];  
        fi.read(a);  
        fi.close();  
        //xuất kết quả lên màn hình  
        txtNoiDungFile.setText(new String(a));  
    }catch(Exception e){}  
}
```

Bước 4: Xử lý sự kiện cho button Ghi File văn bản:

```
private void btVietFileVanBanActionPerformed(java.awt.event.ActionEvent evt) {  
    try{  
        FileWriter fw=new FileWriter(new File(txtFileName.getText()));  
        fw.write(txtNhapChuoi.getText());  
        fw.close();  
    }catch(Exception e){e.printStackTrace();}  
}
```

Bước 5: Xử lý sự kiện cho button Đọc File văn bản:

```
private void btDocFileVanBanActionPerformed(java.awt.event.ActionEvent evt) {  
    try{  
        FileReader fr=new FileReader(new File(txtFileName.getText()));  
        StringBuffer sb=new StringBuffer();  
        char ca[]=new char[5]; //đọc mỗi lần 5 ký tự  
        while(fr.ready()){  
            int len=fr.read(ca);  
            sb.append(ca,0,len);  
        }  
        fr.close();  
        txtNoiDungFile.setText(sb.toString());  
    }catch(Exception e){  
        e.printStackTrace();  
    }  
}
```

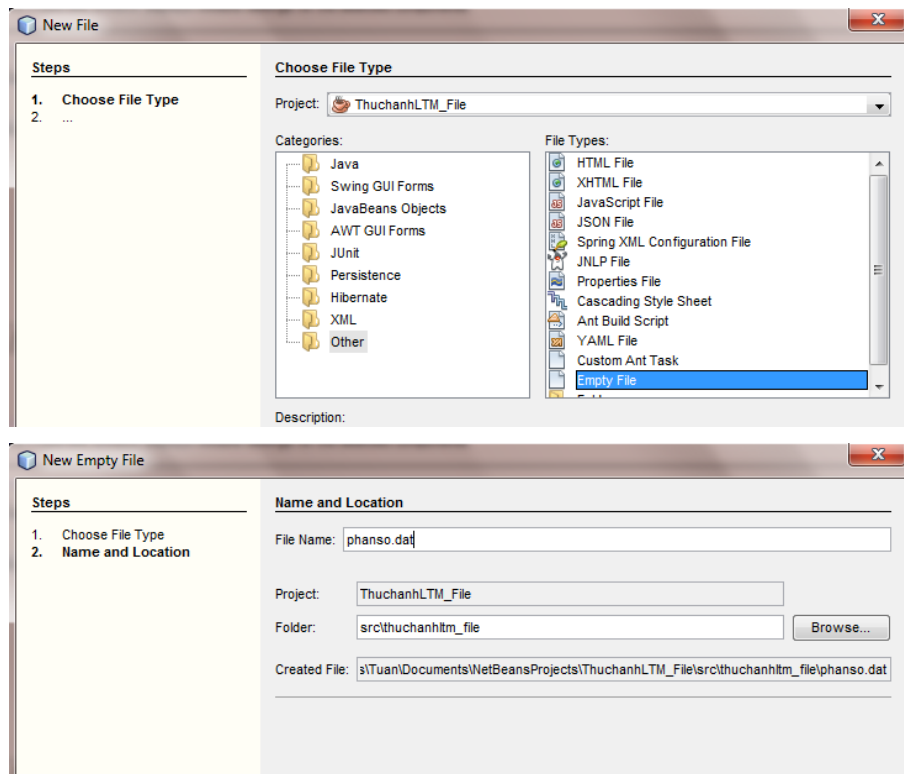
Bài 3. Viết chương trình tạo ra một số đối tượng phân số ngẫu nhiên. Lưu các đối tượng phân số này xuống tập tin “phanso.dat”. Sau đó, đọc các phân số có trong tập tin “phanso.dat” vào chương trình và hiển thị các phân số có mẫu số là số nguyên tố.

Bước 1: Tạo lớp đối tượng PhanSo.java: lớp này phải có phương thức để kiểm tra mẫu số là số nguyên tố, có hàm dựng cho phép tạo ra phân số ngẫu nhiên.

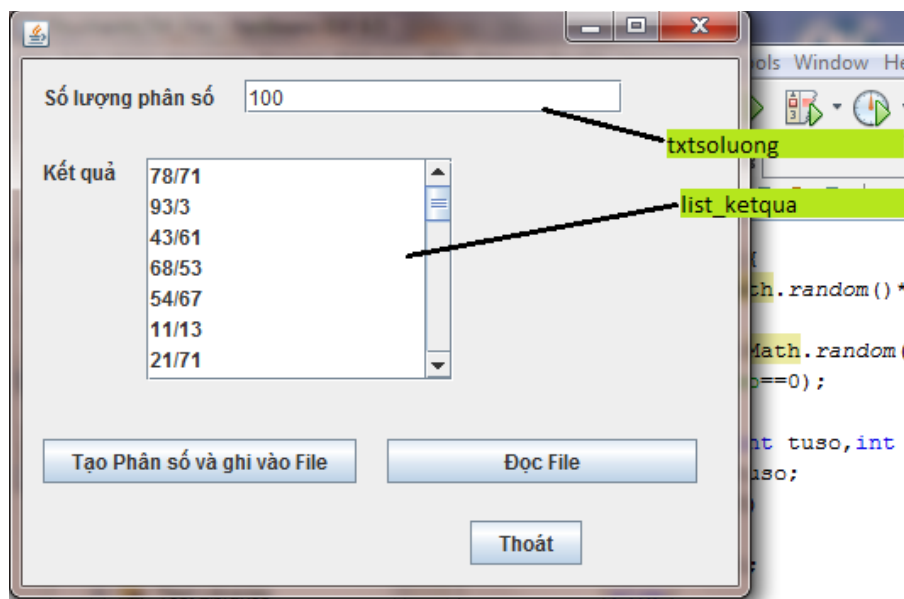
```
import java.io.Serializable;
public class PhanSo implements Serializable {
    private int tuso;
    private int mauso;
    //ham khoi tao phan so ngau nhien
    public PhanSo(boolean a){
        tuso=(int) (Math.random()*100);
        do{
            mauso=(int) (Math.random()*90+1);
        }while(mauso==0);
    }
    public PhanSo(){
        tuso=(int) Math.random()*100;
        do{
            mauso=(int) Math.random()*90+1;
        }while(mauso==0);
    }
    public PhanSo(int tuso,int mauso){
        this.tuso=tuso;
        if(mauso==0)
        {
            mauso=1;
        }else{
            this.mauso=mauso;
        }
    }
    public String toString(){
        return tuso+"/"+mauso;
    }
    public int getMauso(){
        return mauso;
    }
    public boolean ktNguyenTo(){
        if(mauso<1) return false;

        int t=(int) Math.sqrt(mauso);
        for(int i=2;i<=t;i++){
            if (mauso%i==0) return false;
        }
        return true;
    }
}
```

Bước 2: Tạo File phanso.dat để lưu các phân số.



Bước 3: Tạo Project mới, tạo Form với giao diện như hình sau:



Bước 4: Xử lý sự kiện cho button Tạo phân số và ghi vào File:

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    try{
        FileOutputStream fo=new FileOutputStream("phanso.dat");
        BufferedOutputStream bo=new BufferedOutputStream(fo);
        ObjectOutputStream oo=new ObjectOutputStream(bo);
        for(int i=0;i<Integer.parseInt(txtsoluong.getText());i++){
            PhanSo ps=new PhanSo(true);
            oo.writeObject(ps);
        }
        oo.close();
    }catch(Exception e)
    {

    }
}
```

Bước 5: Xử lý sự kiện cho button Đọc file (sinh viên viết xử lý cho thao tác này).

Bước 6: Biên dịch chương trình và kiểm tra kết quả.

Bài 4. Viết chương trình nhập tên một tập tin. Tách tập tin thành từng đoạn. Mỗi đoạn lưu vào mảng số nguyên có kích thước tối đa là 100 phần tử. Sau đó, đọc nội dung các mảng này ghi vào tập tin mới có tên là “Mang.txt”.

Bài 5: Cho bảng dữ liệu sau:

Ngày thực hiện Date	Trạng thái Status	Sản phẩm Item	Số lượng Unit	Đơn giá Price	Tổng tiền Total
2022-09-01	True	Pencil	95	1.99	189.05
2022-09-22		Binder	50	19.99	999.50
2022-10-09	True	Pencil	36	4.99	179.64
2022-11-20		Pen	27	19.99	539.73
2022-12-10	True	Pencil	56	2.99	167.44

Yêu cầu: Thực hiện thao tác ghi các dữ liệu ở bảng trên vào tập tin sau đó đọc dữ liệu từ tập tin và xuất ra màn hình.

Bài 6: Chương trình mã hóa chuỗi ký tự sử dụng thuật toán mã hóa Caesar. Trong đó:

- Lớp thứ nhất đọc dữ liệu thô từ tập tin.
- Lớp thứ hai mã hóa dữ liệu và ghi vào file kết quả.

