

BÀI THỰC HÀNH SỐ 1

Lập trình Java cần biết

1. Mục đích

Hiểu về phần cứng các thiết bị đầu cuối, các phép toán logic, lập trình tổng quát trong lập trình Java.

2. Nội dung

Hiểu về các thiết bị đầu cuối và các phép toán logic.

Hiểu về các phép toán logic.

3. Các bước thực hiện

Bước 1: Tạo project mới trong NetBeans

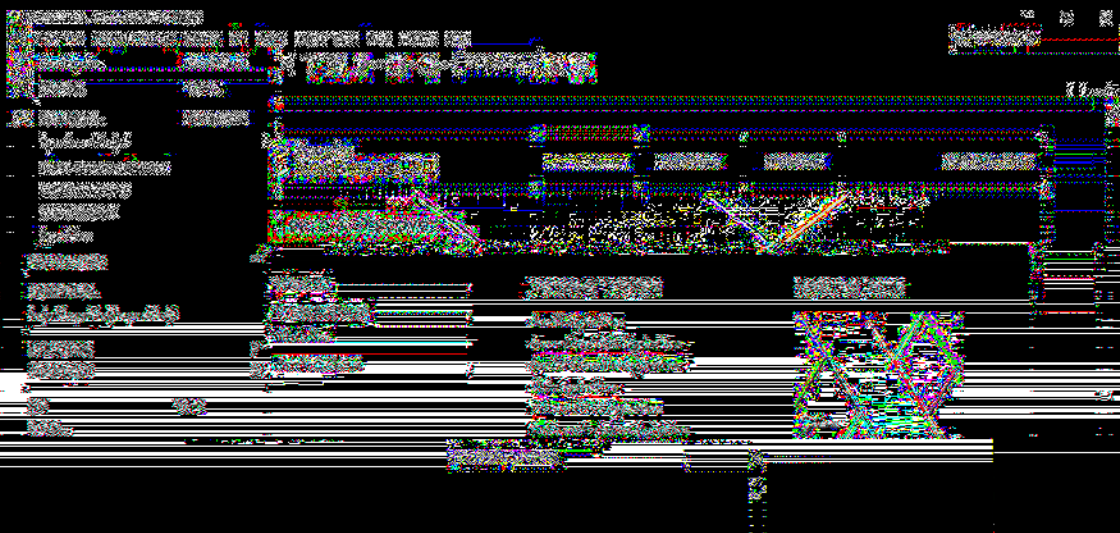
Bước 2: Cài đặt NetBeans

- Bước 3: Tải xuống NetBeans phiên bản Java SE hoặc All tại

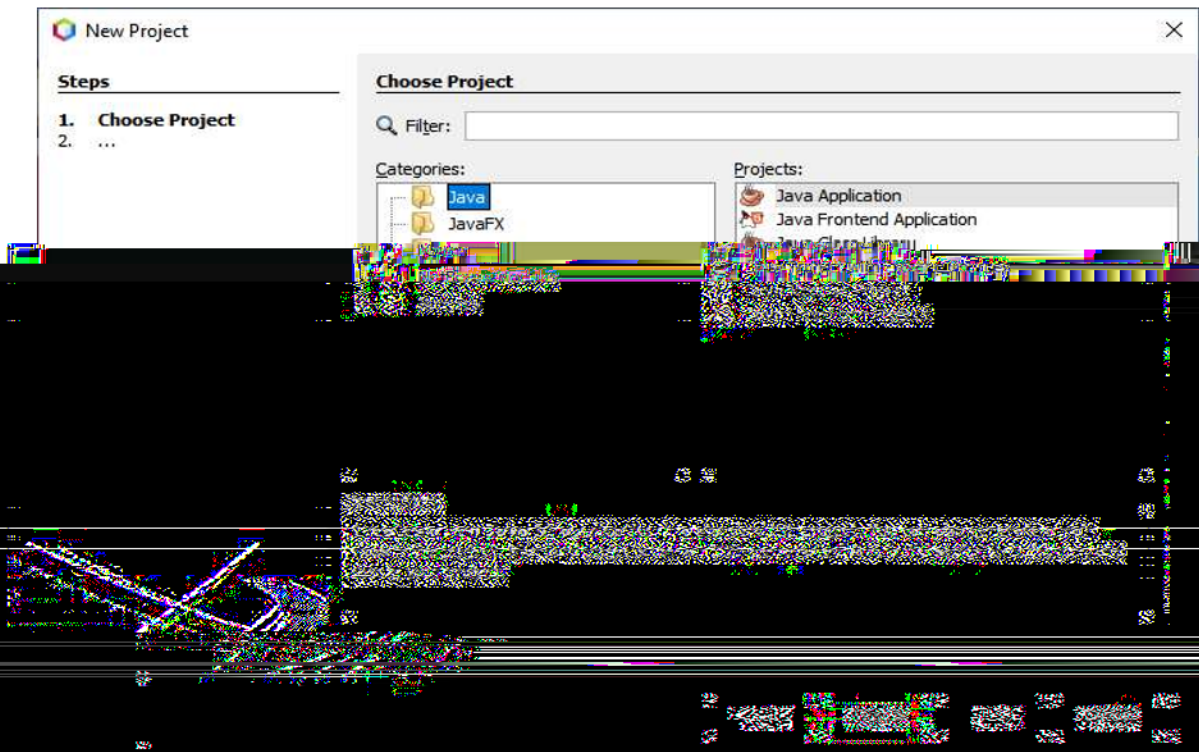
<https://netbeans.org/downloads/>

- Bước 4: Cài đặt NetBeans

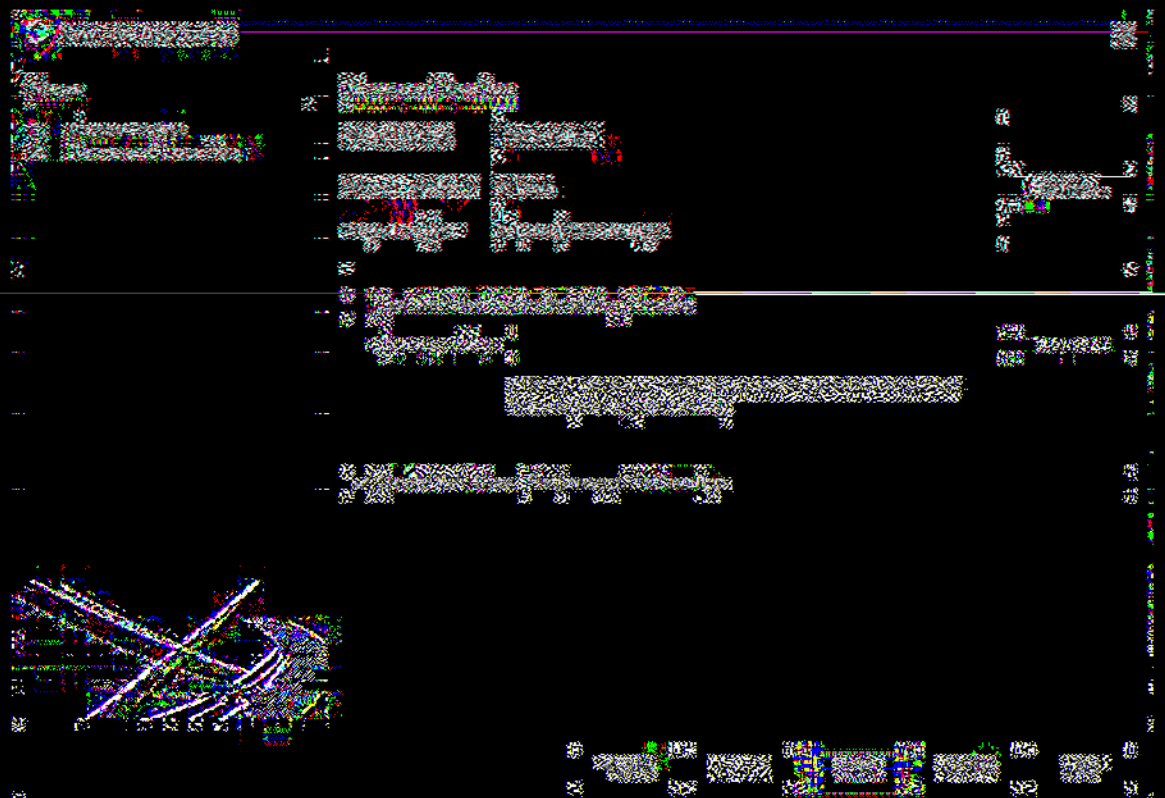
- Bước 5: Mở NetBeans, chọn File > New Project...



- **Bước 4:** Xuất hiện cửa sổ **New Project**, chọn **Projects: Java Application-> Next**



- **Bước 5:** Đặt tên **Project Name** sẽ được dùng để lưu ứng dụng đang viết và là **Java Application**. Chọn check ở **Create Main Class** để tạo class **Main** cho ứng dụng. Sau đó chọn **Finish**. Lưu ý đặt định tên **Main Class** sẽ bằng tên của **Project Name**.



Chúng ta cần một cách để đọc dữ liệu từ bàn phím và đưa nó vào chương trình của mình để xử lý.

Chúng ta



Câu 2: Nhập ký tự, số từ bàn phím.

Chúng ta sẽ sử dụng các lớp sau đây:

- Cách 1: sử dụng lớp `BufferedReader`.
- `BufferedReader` được dùng để đọc dữ liệu từ bàn phím hay từ file. Có thể đọc một chuỗi, một mảng hoặc một ký tự. Dùng để đọc dữ liệu từ bàn phím để tạo đối tượng `BufferedReader` là `InputStreamReader`, hoặc dùng `FileReader` để đọc dữ liệu từ file.

VĐ: `BufferedReader nhap = new BufferedReader(new InputStreamReader(System.in));`

- Nhập số phương thức để đọc dữ liệu của lớp `BufferedReader`:

- `read()` để đọc một ký tự
- `readLine()` để đọc một dòng text

Ví dụ sau đây:

```
// Tạo đối tượng BufferedReader
BufferedReader nhap = new BufferedReader(new InputStreamReader(System.in));
String ten = "";
System.out.print("Nhập tên: ");
```

```
//Nhập từ bàn phím dùng BufferedReader
try {
    ten = nhap.readLine();
} catch (IOException ex) {
    System.out.println("Lỗi nhập dữ liệu!");
}
//Hiển thị tên
System.out.println("Hello " + ten + "!");
```

- Lưu ý: các phương thức để nhập dữ liệu của Bufferreader phải được đặt trong khối lệnh try... catch với lớp xử lý ngoại lệ **IOException**.
- **Cách 2**: Sử dụng lớp **Scanner** (hay dùng)
 - Scanner phân loại được dữ liệu mà người dùng nhập vào theo kiểu int hay float, double...
 - Một số phương thức để đọc dữ liệu của lớp Scanner

- **next() //dùng cho String**

- nextInt()
- nextFloat()
- nextBoolean()
- nextDouble()
- nextLine()

- Ví dụ minh họa:

```
// Tạo một đối tượng Scanner
Scanner scanner = new Scanner(System.in);
String ten = ""; // Tên
int tuoi; // Tuổi
System.out.println("Nhập tên và tuổi: ");
ten = scanner.nextLine();
tuoi = scanner.nextInt();
System.out.println("Tên " + ten + "!");
System.out.println("Tuổi " + tuoi + "!");
```

- Class sử dụng `JOptionPane`
- `JOptionPane` là một lớp kế thừa từ lớp `Component`
- Khi chạy sẽ xuất hiện một dialog box cho phép nhập dữ liệu.
- Một số phương thức của `JOptionPane`
 - `showInputDialog()`: hiển thị một cửa hỏi lựa chọn giống như

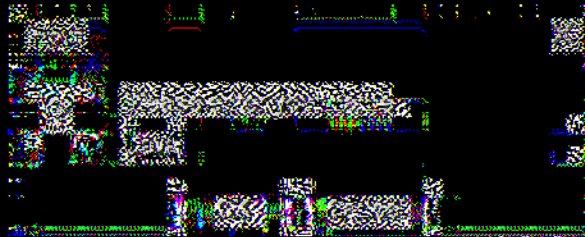
yes/no/cancel.

- **showInputDialog():** hiển thị box nhập.

- **showMessageDialog():** báo cho người dùng một sự kiện vừa xảy ra.

Ví dụ nhập tên:

```
String ten = "";  
ten = JOptionPane.showInputDialog("Please enter your name"); String msg = "Hello " + ten +  
" !"; JOptionPane.showMessageDialog(null, msg); System.out.println("Xin chào : " + msg);
```



Window of dialog

- **Printout** là một lớp trong gói javax.swing.text hỗ trợ việc in ra màn hình.

2.

Để nhập dữ liệu từ bàn phím và xuất ra màn hình thì

ta dùng phương thức **read()** hoặc **System.out.println()**.

read(); **String s =**

readLine(); **String s =**

readLine(); **String s =**

readLine(); **String s =**

phần này để hướng dẫn ở đây

- **Printout** là một lớp trong gói javax.swing.text hỗ trợ việc in ra màn hình.

màn hình bằng cách dùng **System.out.println()**.

Scanner scan = new Scanner(System.in);

String s =

scan.nextLine(); **String s =**

scan.nextLine(); **String s =**

scan.nextLine(); **String s =**

IV. Thực hành tại lớp

Bài 1: Viết hàm nhập vào bán kính của đường tròn, tính chu vi đường tròn đó.

Hướng dẫn: sử dụng hằng số Math.PI để tính chu vi.

Bài 2: Viết hàm Nhập vào 2 số a, b. Xuất ra màn hình kết quả a/b (lấy số 3 lẻ)

Hướng dẫn: sử dụng hàm printf để in ra. `System.out.printf("%.3f", r);`

Bài 3: Viết hàm nhập 2 phân số. Tính tổng, hiệu, tích, thương của 2 phân số với nhau.

Bài 4: Viết hàm nhập vào 2 chuỗi x và y (Hướng dẫn: sử dụng hàm `nextLine()` của lớp `Scanner`). Trả về kết quả theo các yêu cầu sau:

- Cho biết tổng chiều dài của chuỗi x. (Hướng dẫn: sử dụng hàm `length()`)
- Cho biết 3 ký tự đầu tiên của chuỗi x. (Hướng dẫn: sử dụng hàm `substring()`)
- Cho biết 3 ký tự cuối của chuỗi x.
- Cho biết ký tự thứ 6 của chuỗi x. (Hướng dẫn: sử dụng hàm `charAt()`)
- Tạo chuỗi mới gồm 3 ký tự đầu tiên của chuỗi x và 3 ký tự cuối của chuỗi y.
- Kiểm tra 2 chuỗi x, y có bằng nhau hay không (phân biệt chữ hoa, chữ thường)? (Hướng dẫn: sử dụng hàm `equals()`)
- Kiểm tra 2 chuỗi x, y có bằng nhau hay không (không phân biệt chữ hoa, chữ thường)? (Hướng dẫn: sử dụng hàm `equalsIgnoreCase()`)
- Cho biết y có xuất hiện trong x hay không? Nếu có, tại vị trí nào? (Hướng dẫn: sử dụng hàm `indexOf()`)
- Cho biết tất cả các vị trí xuất hiện của y trong x.

Bài 5: Viết hàm nhập vào số điện sử dụng của chúng ta tính tiền điện dựa vào phương pháp lấy tiền.

- Nếu số điện ≤ 50 kw thì giá mỗi số điện là 2000 đ.
- Nếu số điện > 50 kw và ≤ 100 kw thì giá mỗi số điện vượt hạn mức 50 kw là 2500 đ.
- Nếu số điện > 100 kw thì thì giá mỗi số điện vượt hạn mức 50 kw là 2500 đ.

Bài 6: Viết hàm để tổ chức 1 menu để gọi món ăn từ bài 1 đến bài 5 như sau

#####

1. Bài 1
2. Bài 2
3. Bài 3
4. Bài 4
5. Bài 5
6. Thoát

#####

Chọn chức năng: _

Chọn chức năng từ 1 đến 6 để gọi các hàm đã viết ở bài 1 đến 5

Hướng dẫn:

- Dùng hàm **System.out.println()** để xuất ra màn hình menu như mẫu.
- Dùng hàm **nextInt()** để nhận ký tự số từ 1 đến 6 nhập từ bàn phím tương đương với 6 chức năng trong menu.
- Sử dụng lệnh **Switch** để gọi đến các hàm tương ứng đã viết ở bài 1 đến 5.

Bài 7: Viết chương trình nhập vào một số nguyên dương từ bàn phím và kiểm tra xem số đó có phải là số nguyên tố hay không? (Số nguyên tố là số chỉ chia hết cho 1 và chính nó).

Hướng dẫn: sử dụng vòng lặp **for**

ok = true;

for (int i = 2; i < n-1; i++){ //n là số nguyên dương nhập từ bàn phím

if (n % i == 0){ //n chia hết cho i

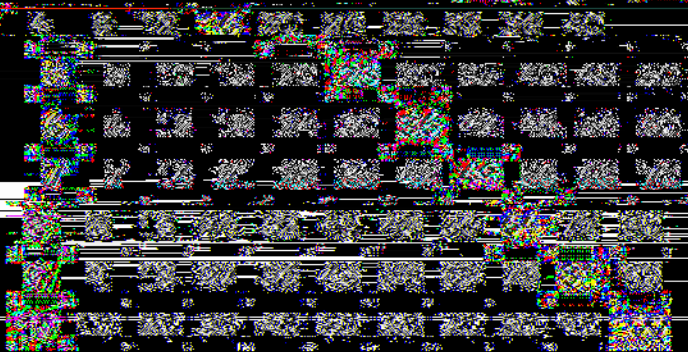
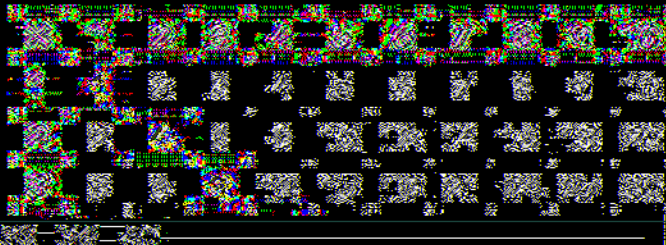
ok = false; break;

}

}

Kiểm tra, nếu sau khi hết vòng lặp mà ok = true thì n là số nguyên tố

Bài 8: Viết chương trình xuất ra màn hình "Bảng tính Pythagoras"



Bài 9: Viết chương trình nhập vào một số nguyên dương n và in ra màn hình các số nguyên tố từ 1 đến n.

Hướng dẫn: Sử dụng vòng lặp for để kiểm tra xem số đó có phải là số nguyên tố hay không.

Bài 10: Viết chương trình nhập vào một số nguyên dương n và in ra màn hình các số nguyên tố từ 1 đến n.

Hướng dẫn: Sử dụng vòng lặp for để kiểm tra xem số đó có phải là số nguyên tố hay không.

Bài 11: Viết chương trình nhập vào một số nguyên dương n và in ra màn hình các số nguyên tố từ 1 đến n.

Hướng dẫn: Sử dụng vòng lặp for để kiểm tra xem số đó có phải là số nguyên tố hay không.

Bài 12: Viết chương trình nhập vào một số nguyên dương n và in ra màn hình các số nguyên tố từ 1 đến n.

Hướng dẫn: Sử dụng vòng lặp for để kiểm tra xem số đó có phải là số nguyên tố hay không.

Bài 13: Viết chương trình nhập vào một số nguyên dương n và in ra màn hình các số nguyên tố từ 1 đến n.

Hướng dẫn: Sử dụng vòng lặp for để kiểm tra xem số đó có phải là số nguyên tố hay không.

- Tìm phần tử lớn nhất và nhỏ nhất trong mảng. (Hướng dẫn: Dùng hàm **Math.max** và **Math.min**)

- Nhận vào số x , tìm ra x có nằm trong mảng hay không? (Hướng dẫn: dùng hàm **Array.indexOf()**)

• Viết chương trình đếm số phần tử trong mảng có giá trị lớn hơn 10

• Viết chương trình đếm số phần tử trong mảng có giá trị nằm trong khoảng từ 10 đến 20

V. Bài tập về số

Bài 1: Cho một số nguyên lớn nhất và một số nguyên nhỏ nhất của 2 số nguyên và tính tổng của chúng.

Bài 2: Nhập vào số nguyên dương n , tính tổng các số từ 1 đến n .

Bài 3: Nhập vào hai số nguyên và tính tổng của chúng. Nếu tổng của hai số là số chẵn thì in ra tổng của chúng.

Bài 4: Viết chương trình tính tổng của các số nguyên dương từ 1 đến n bằng cách sử dụng vòng lặp. Ví dụ: nếu $n=6$ thì tổng các số (từ 1 đến 6) bằng 21. Ví dụ: nếu $n=6$ thì tổng các số $1+2+3+4+5+6=21$.

Bài 5: Nhập vào mảng số nguyên A (kích thước n)

- Tạo mảng số nguyên B (kích thước n) với các giá trị ngẫu nhiên. (Hướng dẫn: sử dụng hàm **Math.random()**)
- Xóa phần tử của mảng B ra khỏi mảng.
- Tạo mảng C từ mảng A (copy A qua C) (Hướng dẫn: sử dụng hàm **Array()**)
- Thay đổi phần tử thứ 1 > 2 của mảng C bằng 3 phần tử cuối của mảng B (Hướng dẫn: sử dụng hàm **System.arraycopy()**)
- Sắp xếp mảng C tăng dần và xóa ra khỏi mảng.

Bài 6: Viết chương trình tính tổng của mảng hai chiều A có n dòng, m cột, các phần tử là số nguyên lớn hơn 0 và nhỏ hơn 100. Tính tổng của các phần tử trong mảng.

• Tính phần tử lớn nhất của mảng A bằng một số của nó.

- Tính giá trị của các phần tử là số nguyên tố của mảng A (các phần tử là những số nguyên tố khác 0).

- Sắp xếp các giá trị của mảng A theo thứ tự tăng dần và in ra kết quả ra ngoài màn hình.

Bài 7: Viết chương trình tính tổng của các số nguyên từ 1 đến n bằng cách sử dụng vòng lặp.

Viết output và input của chương trình sau:

```

/
70.0 69.0 69.0 68.0 67.0 67.5 67.3 68.0 69.2 72.1 74.3 78.3
72.4 69.4 69.4 68.4 67.4 67.2 67.7 68.4 69.2 72.1 74.7 78.7
70.0 69.0 69.0 68.0 67.0 67.2 67.7 68.0 69.2 73.1 74.7 78.3
71.1 69.1 69.1 69.1 67.1 67.2 67.7 69.1 69.2 73.1 74.7 78.7
70.6 69.6 69.6 68.6 65.6 65.2 65.0 68.6 69.2 72.1 74.0 78.0
70.3 68.3 68.3 68.3 67.3 67.5 67.7 68.3 68.5 73.1 74.7 77.7

```


69.4 70.4 70.4 68.4 67.4 67.2 67.7 68.4 70.2 72.1 74.7 78.7

File input lưu dữ liệu về nhiệt độ trong ngày. Số đầu tiên là số dòng, biểu thị số ngày được lấy dữ liệu. Mỗi ngày sẽ có 12 số double, biểu thị cho nhiệt độ mỗi 2 giờ đồng hồ (có ngày có thể từ 2 đến 1000, nhưng mỗi ngày luôn có 12 lần đo nhiệt độ). Tìm nhiệt độ trung bình, nhiệt độ cao nhất, nhiệt độ thấp nhất của mỗi ngày và trung bình cả các ngày.

Bài 8: Viết chương trình đọc hai file đầu vào input (ví dụ: "input8.1.txt", "input8.2.txt", in ra năm bình số thực tự động sau số tự động (chẵn nhau trong cả hai file, thêm "/" nếu dòng trong cùng ở file "input8.1.txt" và "1") nếu dòng trong cùng với "input8.2.txt".

Ví dụ: 10// a**o**

10\\ c**a**

Bài 9: Viết chương trình kiểm tra chuỗi nhập vào có phải là palindrome hay không. Ví dụ input có thể có nhiều dòng.

Ví dụ:

```
123456789101112
```

"Madam, I'm Adam."

"A man a plan a canal Panama."