**LAB 2: THỰC HÀNH VỚI CHUỖI, ĐIỀU KIỆN VÀ VÒNG LẶP**

**I/ Yêu cầu thực hành**

1. Học viên đã nắm được kiến thức về xử lý chuỗi, điều kiện và vòng lặp trong java
2. Thực hành đọc hiểu đề bài, phân tích và tìm giải pháp cho đề bài
3. Thực hành đưa giải pháp vào viết code, thực thi và sửa lỗi code nếu có
4. Lưu lại code.

**II/ Ví dụ mẫu**

**1. (If**): Yêu cầu nhập vào từ bàn phím một số nguyên N, kiểm tra số vừa nhập là số chẵn hay số lẻ.

*Phân tích:*  
 Số chẵn là số chia hết cho 2 (%2 = 0). Số lẻ thì ngược lại.

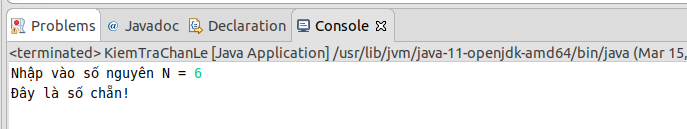
*Giải pháp:*

* Sử dụng **Scanner** để nhập dữ liệu từ bàn phím
* Sử dụng If-else để kiểm tra số vừa nhập là số chẵn hay số lẻ.
* In ra màn hình thông báo “Đây là số chẵn” hoặc “Đây là số lẻ”

*#Code*

| **package** test;  **import** java.util.Scanner;  **public** **class** KiemTraChanLe {  **public** **static** **void** main(String[] args) {  System.***out***.print("Nhập vào số nguyên N = ");  Scanner scanner = **new** Scanner(System.***in***);  **int** N = scanner.nextInt();  **if** (N % 2 == 0) {  System.***out***.println("Đây là số chẵn!");  } **else** {  System.***out***.println("Đây là số lẻ!");  }  }  } |
| --- |

*#Kết quả:*



**2. (While) :** Nhập vào từ bàn phím một số nguyên dương N > 0. In ra màn hình dãy số theo thứ tự giảm dần từ N → 1.  
Ví dụ: nhập N = 10 thì in ra dãy số 10 9 8 7 6 5 4 3 2 1

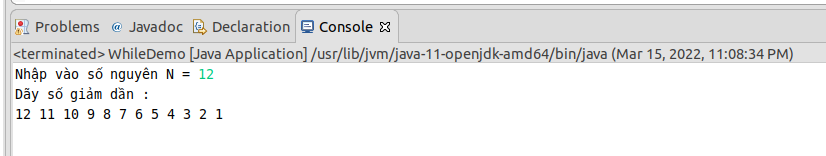
*Phân tích:*  
 Giảm dần số vừa nhập về 1, mỗi lần giảm 1 đơn vị và in ra màn hình  
*Giải pháp:*

* Sử dụng **Scanner** để nhập dữ liệu từ bàn phím
* Sử dụng vòng lặp while để lặp, mỗi vòng lặp giảm số vừa nhập đi 1 đơn vị, vòng lặp sẽ lặp cho đến khi giá trị giảm về 0 thì dừng lại.
* Mỗi vòng lặp, in ra màn hình giá trị hiện tại

*#Code*

| **package** test;  **import** java.util.Scanner;  **public** **class** WhileDemo {  **public** **static** **void** main(String[] args) {  System.***out***.print("Nhập vào số nguyên N = ");  Scanner scanner = **new** Scanner(System.***in***);  **int** N = scanner.nextInt();  System.***out***.println("Dãy số giảm dần :");  **while** (N > 0) {  System.***out***.print(N + " ");  N--;  }  }  } |
| --- |

*#Kết quả:*



**3.(Do-while)** Yêu cầu nhập vào một số nguyên dương N > 0, Nếu N < 0 hoặc N = 0 thì yêu cầu nhập lại.

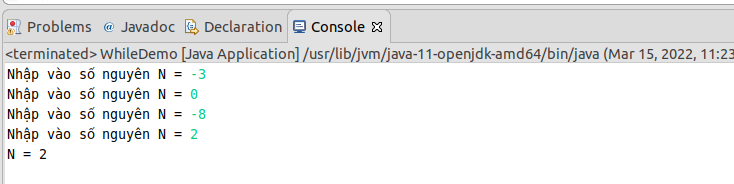
*Giải pháp:*

* Sử dụng **Scanner** để nhập dữ liệu từ bàn phím
* Sử dụng do- while để kiểm tra số vừa nhập vào, nếu nhập sai, yêu cầu nhập lại.

*#Code*

| **package** test;  **import** java.util.Scanner;  **public** **class** WhileDemo {  **public** **static** **void** main(String[] args) {  Scanner scanner = **new** Scanner(System.***in***);  **int** N;  **do** {  System.***out***.print("Nhập vào số nguyên N = ");  N = scanner.nextInt();  } **while** (N <= 0);  System.***out***.println("N = " + N);  }  } |
| --- |

*#Kết quả:*



**4. (For) :** Yêu cầu nhập vào một số nguyên dương N > 0. Tính tổng các số chẵn và các số lẻ từ 1→ N. In kết quả ra màn hình.

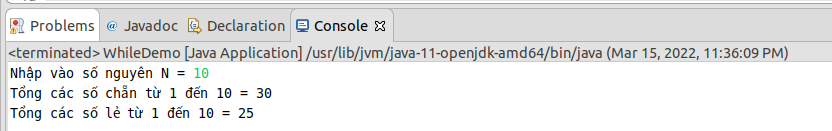
*Gợi ý:*

* Sử dụng **Scanner** để nhập dữ liệu từ bàn phím
* Sử dụng vòng lặp for để duyệt từ 1 đến N
* Sử dụng điều kiện if -else kiểm tra chẵn/lẻ. Số chẵn chia hết cho 2 và ngược lại.
* Tạo 1 biến int tongChan: để lưu lại tổng các số chẵn. Biến int tongLe: để lưu lại tổng số lẻ.
* Với mỗi vòng lặp, thực hiện tính tổng số chẵn và số lẻ bằng cách lấy tổng hiện tại + chẵn/lẻ

*#Code*

| **package** test;  **import** java.util.Scanner;  **public** **class** DoWhileDemo {  **public** **static** **void** main(String[] args) {  **int** tongChan = 0;  **int** tongLe = 0;  Scanner scanner = **new** Scanner(System.***in***);  System.***out***.print("Nhập vào số nguyên N = ");  **int** N = scanner.nextInt();  **for** (**int** i = 1; i <= N; i++) {  **if** (i % 2 == 0) {  // day la so chan  tongChan += i; // tongChan = tongChan + i  } **else** {  //day là so le  tongLe += i; // tongLe = tongLe + i  }  }    System.***out***.println("Tổng các số chẵn 1 đến " + N + " = " + tongChan);  System.***out***.println("Tổng các số lẻ 1 đến " + N + " = " + tongLe);  }  } |
| --- |

*#Kết quả:*



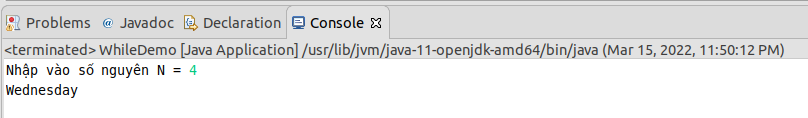
**5. (Switch – case):** Nhập vào 1 số bất kì từ bàn phím. Nếu số đó có giá trị bằng 2 → in ra màn hình “Monday. Nếu = 3 thì in ra “Tuesday … 8→ “Sunday”  
Nếu nằm ngoài giá trị từ 2→ 8, in ra “Giá trị không hợp lệ”

*Giải pháp:*

* Sử dụng **Scanner** để nhập dữ liệu từ bàn phím
* Sử dụng switch- case để kiểm tra giá trị nhập vào và in ra dòng thông báo tương ứng

| **package** test;  **import** java.util.Scanner;  **public** **class** SwitchCaseDemo {  **public** **static** **void** main(String[] args) {  Scanner scanner = **new** Scanner(System.***in***);  System.***out***.print("Nhập vào số nguyên N = ");  **int** input = scanner.nextInt();  String message = ""; // Tạo một biến message để lưu thông báo  **switch** (input) {  **case** 2:  message = "Monday";  **break**;  **case** 3:  message = "Tuesday";  **break**;  **case** 4:  message = "Wednesday";  **break**;  **case** 5:  message = "Thursday";  **break**;  **case** 6:  message = "Friday";  **break**;  **case** 7:  message = "Saturday";  **break**;  **case** 8:  message = "Sunday";  **break**;  **default**:  message = "Giá trị không hợp lệ";  **break**;  }  System.***out***.println(message); //in message ra  }  } |
| --- |

*#Kết quả:*



**6.(String)**:Cho chuỗi sau : "Xi nc hao c a cba n";

* Đếm số dấu cách (khoảng trắng) trong chuỗi.
* Xóa bỏ toàn bộ khoảng trắng
* Chuyển tất cả chữ cái về in hoa.
* In kết quả ra màn hình.

*Giải pháp:*

* Lấy ra chiều dài của chuỗi, sử dụng vòng lặp FOR để duyệt chuỗi, đếm số ký tự (khoảng trắng) . Lấy ra ký tự tại vị trí thứ i trong chuỗi và kiểm tra ký tự đó có bằng ký tự trắng hay không:  
    
   char kyTu = chuoi.charAt(i);

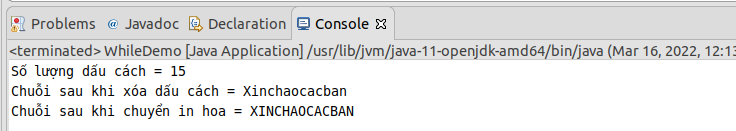
if( kyTu == ' ') ..

* Sử dụng phương thức replaceAll() để xóa khoảng trắng.
* Sử dụng Up

*#Code*

| **package** test;  **import** java.util.Scanner;  **public** **class** StringDemo {  **public** **static** **void** main(String[] args) {  String input = "Xi nc hao c a cba n";  **int** totalSpace = 0; // Biến đếm dấu cách  **int** len = input.length(); // lấy ra độ dài chuỗi  **for** (**int** i = 0; i < len; i++) {  **char** kyTu = input.charAt(i);  **if** (kyTu == ' ') { // so sánh với dấu cách  totalSpace++; // totalSpace = totalSpace + 1  }  }  System.***out***.println("Số lượng dấu cách = " + totalSpace);  // xóa khoảng trắng  input = input.replaceAll(" ", "");  System.***out***.println("Chuỗi sau khi xóa dấu cách = " + input);  // chuyển sang in hoa  input = input.toUpperCase();  System.***out***.println("Chuỗi sau khi chuyển in hoa = " + input);  }  } |
| --- |

*#Kết quả:*



**III. Thực hành**

**Bài 1**. **Thực hành với điều kiện**

**1.1.** Nhập vào từ bàn phím số nguyên N là số năm tương ứng. Kiểm tra năm đó có phải là năm nhuận hay không?.

In kết quả ra màn hình.  
  
*Gợi ý:*

Năm nhuận là:

* Năm chia hết cho 400.
* Năm chia hết cho 4 nhưng không chia hết cho 100.

Ví dụ: 1800 không phải năm nhuận, 2000 là năm nhuận.  
  
**B1**: Nhập vào N > 0  
  
**B2**: Sử dụng toán tử % để lấy dư. Ví dụ: 4%2 =0 → phép chia hết, 5%2=1 → không chia hết.  
 - DK1: Kiểm tra có chia hết cho 400 không?

| boolean dk1 = (năm % 400 == 0);  (lấy dư của phép chia 400 và so sánh == 0. Kết quả sẽ là false hoặc true) |
| --- |

- DK2: Kiểm tra có chia hết cho 4 không ?

- DK3: Kiểm tra có không chia hết cho 100 không ?

**B3**. Sử dụng if-else để kiểm tra tổng hợp các điều kiện

**1.2.** Viết chương trình giải phương trình bậc 2 có dạng: ax2+ bx + c = 0

Với tham số a,b,c nhập từ bàn phím.  
  
*Gợi ý:*

Ôn lại kiến thức giải phương trình bậc 2.

**B1**. Nhập tham số a,b,c và Tính Delta = b\*b -4ac   
 (chú ý, tất cả các tham số và biến đều là kiểu số thực: float hoặc double.Các bạn có thể sử dụng Math.pow(x, y) = x mũ y để tính b bình phương.)  
**B2**. Xét điều kiện:

* Nếu Delta < 0 thì phương trình vô nghiệm.
* Nếu Delta = 0 thì phương trình có 1 nghiệm được tính bằng cách: x1 = x2 = -b / 2 \* a.
* Nếu Delta > 0 thì phương trình có hai nghiệm phân biệt là

x1 = (-b + Math.sqrt(delta)) / (2 \* a).

x2 = (-b – Math.sqrt(delta)) / (2 \* a).

(Note: **Math.sqrt (x)**: hàm lấy căn bậc 2 của x)

Gợi ý khác: có thể sử dụng if - else if - else để xét điều kiện cho delta  
**B3**: In kết quả ra màn hình

**Bài 2**. **Thực hành với vòng lặp while và do-while**

Viết chương trình nhập vào số nguyên N từ bàn phím, việc này sẽ tiếp tục cho đến khi tổng các giá trị nhập vào >= 100.  
Thực hiện in ra giá trị N sau cùng thỏa mãn điều kiện:  
- Nếu N là một số lẻ :  
 In ra màn hình “**Dãy số lẻ giảm dần :** ” và kèm theo dãy số lẻ giảm dần từ N về 1  
- Ngược lại, nếu N là một số chẵn thì In ra màn hình “**Dãy số chẵn giảm dần :** ” và kèm theo dãy số chẵn giảm dần từ N về 2

*Gợi ý:*

**B1**. Tạo một biến kiểu **int** để lưu giá trị tổng và gán giá trị khởi tạo ban đầu bằng 0.   
 Sử dụng vòng lặp do-while để nhập N. Với mỗi lần lặp thì cộng dồn giá trị N với tổng trước đó. (tong = tong + N)  
Vòng lặp sẽ chỉ dừng lại khi điều kiện tổng >= 100  
**B2**. Sử dụng if-else để kiểm tra tính chẵn lẻ của N  
**B3**: Sử dụng vòng lặp while để lặp từ N về 0 , mỗi vòng lặp giảm N đi 2 đơn vị

**Bài 3**. **Thực hành với vòng lặp for**

Viết chương trình nhập vào một số nguyên dương N từ bàn phím.  
Đếm và In ra màn hình tất cả các số nguyên tố từ 1 đến N.

*Gợi ý:*

**B1**. Xác định tính chất số nguyên tố:  
 Một số nguyên tố là số chỉ chia hết cho 1 và chính nó.  
**B2**. Viết 1 hàm kiểm tra 1 số nguyên bất kì có phải là số nguyên tố hay không

| **public static boolean** kiemTraSoNguyenTo(**int** k) {  **boolean** isNguyenTo = **true**;  *//// code here* **for** ...  **if** (...) {  isNguyenTo = **false**;  **break**;  }    **return** isNguyenTo; } |
| --- |

Trong hàm này:  
Nếu K > 0 và K <= 2 thì trả luôn ra là số nguyên tố.  
Ngược lại:  
Các bạn có thể dùng vòng lặp for để kiểm tra các số từ (2→ K)  
 Nếu phát hiện K chia hết cho 1 số bất kì nằm trong khoảng này thì ngay lập tức  
 xác định K không phải là số nguyên tố và dùng từ khóa **break** để thoát khỏi vòng lặp.  
   
**B3**: Nhập vào số N, Sử dụng vòng lặp FOR để duyệt từ 1 đến N.

Trong mỗi vòng for, kiểm tra giá trị hiện tại có phải là số nguyên tố hay không thông qua hàm **kiemTraSoNguyenTo**,

Nếu là số nguyên tố thì biến đếm sẽ tăng lên 1 đơn vị và đồng thời in số đó ra màn hình.

B4: Sau cùng, In biến đếm số số nguyên tố từ 1 đến N.

**Bài 4**. **Thực hành với switch - case**  
Viết chương trình cho phép người dùng chọn đáp án chính xác từ câu hỏi sau:  
 Gói nào trong java chứa lớp Scanner dùng để nhập dữ liệu từ bàn phím ?

A. java.net  
B. java.io  
C. java.util  
D. java.awt  
  
Câu trả lời của bạn : C   
  
=> Đáp án in hoa hay in thường đều được chấp nhận.  
Sau khi người dùng chọn đáp án , Chương trình sẽ thông báo đáp án là đúng hoặc sai.  
  
*Gợi ý:*

**B1**. In toàn bộ câu hỏi và đáp án ra màn hình  
**B2**. Chú ý :  
Cần sử dụng lệnh **trim()** để cắt bỏ khoảng trắng 2 đầu cuổi đáp án người dùng.

Ví dụ:

String input = scanner.nextLine();

input = input.trim();  
  
Cần sử dụng **toUpperCase**() hoặc **toLowerCase**() để chuyển đáp án người dùng sang chữ Hoa hoặc chữ thường.

input = input.toUpperCase();

Sử dụng **switch**- **case** để kiểm tra đáp án người dùng nhập vào.   
  
Trường hợp người dùng nhập đáp án không có trong 4 đáp án, in ra thông báo là “Đáp án không hợp lệ!”  
  
**B3**: In ra thông báo kết quả tương ứng với các đáp án

**Bài 5**. **Thực hành với Chuỗi (String)**  
Cho chuỗi sau : "xinchao"  
Viết chương trình để viết hoa chữ cái đầu tiên của chuỗi trên.  
Kết quả mong muốn: “Xinchao”  
  
*Gợi ý:*

**B1**. Sử dụng phương thức cắt chuỗi **substring()** để tách chuỗi trên thành 2 chuỗi con. (thứ tự các ký tự của chuỗi sẽ bắt đầu từ 0 … length -1)  
Chuỗi con thứ nhất chỉ chứa ký tự đầu tiên của chuỗi   
Chuỗi con còn lại chứa các ký tự còn lại.

substring(**int** beginIndex, **int** endIndex)  
 ví dụ s = “abcdefgh”  
 s1 = s.substring(0,2) //~(0<= index <2)  
 => s1 = “ab”

**B2**. Sử dụng **toUpperCase()** đểchuyển chuỗi con đầu tiên thành In hoa và gán lại cho chuỗi 1.  
  
**B3**: Cộng 2 chuỗi lại với nhau ta sẽ thu được chuỗi mong muốn.