**Nhập xuất từ bàn phím**

Hẳn từ những ngày đầu biết sử dụng máy tính và phần mềm, các bạn không thể nào thiếu được cái Keyboard của mình 😂 Quả đúng là như vậy, bàn phím là thứ công cụ giúp chúng ta giao tiếp với máy tính, nên nếu Java không hỗ trợ cái này thì quả là một thiếu sót lớn.

Quay trở lại ví dụ của [Bài #3](https://loda.me/java-basic-3-ham-va-cau-lenh-dieu-kien-loda1553158182446). Chương trình kiểm tra tam giác của chúng ta còn thiếu sự linh động, do phải điều chỉnh các biến a,b,c trực tiếp trong code.

Bây giờ chúng ta sẽ upgrade bài toán một chút: "Nhận vào 3 số nguyên a,b,c **từ bàn phím** và kiểm tra nó là tam giác gì?"

Cùng xem cách mình thực hiện, và mình sẽ giải thích ở dưới:

public class Calculation {

public static void main(String[] args) {

// Chúng ta khai báo 3 biến a,b,c không có giá trị.

int a, b, c;

//Khai báo đối tượng Scanner, giúp chúng ta nhận thông tin từ keyboard

Scanner sc = new Scanner(System.in);

System.out.print("Nhập a: "); //print thay vì println, nó sẽ in ra, nhưng không xuống dòng

a = sc.nextInt(); // sc.nextInt() là cách để lấy giá trị từ bàn phím, nó sẽ chờ tới khi chúng ta nhập một số.

System.out.print("Nhập b: ");

b = sc.nextInt();

System.out.print("Nhập c: ");

c = sc.nextInt();

// In các giá trị ra màn hình

System.out.println("a = " + a + ", b = " + b + ", c = " + c);

// Đây là phép cộng String mình đã nói trong Bài #1.

}

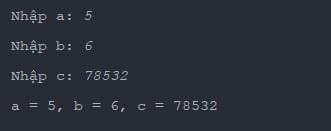
Khi chạy chương trình này, nó sẽ dừng lại sau khi in ra Nhập a như thế này:



Vì sao? 😕 Vì cái dòng lệnh này a = sc.nextInt(). Nó sẽ chờ cho tới khi bạn nhập 1 số nguyên và gõ Enter thì thôi. Giả sử mình nhập 5



Chương trình lại tiếp tục chạy cho tới khi gặp câu lệnh sc.nextInt() tiếp theo. Và cứ tiếp tục như vậy cho tới dòng lệnh cuối cùng.



Từ đây, các bạn có thể hiểu là đối tượng Scanner đã làm nhiệm vụ là nhận dữ liệu người dùng nhập từ bàn phím, và gán nó vào biến, bằng câu lệnh nextInt.

Bây giờ quay trở ngược lên trên 1 chút, ở câu lệnh:

Scanner sc = new Scanner(System.in);

các bạn sẽ thấy một khái niệm là new. cái này thì [Bài #5][link-bai5] mình sẽ nói chi tiết, còn ở đây thì bạn hiểu nó được sử dụng để tạo ra 1 đối tượng Scanner. Mà từ đó các bạn mới nhận dữ liệu từ bàn phím được.

**Các phương thức nhập xuất**

Tới đây, bạn đã có thể nhập xuất dữ liệu int từ bàn phím rồi, thế còn double hay String thì như thế lào :3

cái này bạn nào thực hành đoạn code trên rồi thì sẽ nhìn thấy ngay là Scanner có một loạt các hàm hỗ trợ như sau:

* next(): Nhận vào một String token (nhận vào 1 từ đầu tiên thay cả câu)
* nextInt(): Nhận vào một số int
* nextLong(): Nhận vào một số long
* nextFloat(): Nhận vào một số float
* nextDouble(): Nhận vào một số double
* sc.nextLine(): Nhận vào một chuỗi String (Cả 1 câu)
* nextByte(): Nhận vào một byte
* nextBoolean(): Nhận vào một boolean

Các hàm trên bạn hiểu nguyên lý là nó đều sẽ chờ cho tới khi bạn nhập kiểu dữ liệu nó muốn vào.

Có next() và nextLine() khá đặc biệt, mình sẽ ví dụ:

Scanner sc = new Scanner(System.in); //Tạo đối tượng Scanner

System.out.print("Nhập gì đó: ");

String a = sc.nextLine(); // nhận vào 1 string

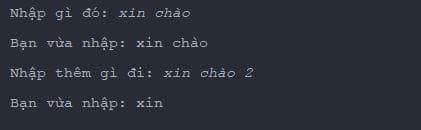
System.out.println("Bạn vừa nhập: "+a);

System.out.print("Nhập thêm gì đi: ");

String b = sc.next(); // cũng nhận vào 1 String

System.out.println("Bạn vừa nhập: "+b);

Chúng ta chạy chương trình để xem nó ra sao:



nextLine thì nhận vào cả 1 chuỗi dài String, cho tới khi bạn nhấn Enter. Còn next dù bạn có nhập dài như nào, nó cũng nhận 1 từ đầu tiên thôi.

Okie, giờ mình sử dụng các kiến thức này để nâng cấp bài tập kiểm tra tam giác ở [Bài #3](https://loda.me/java-basic-3-ham-va-cau-lenh-dieu-kien-loda1553158182446) nhé.

class Calculation {

public static void main(String[] args) {

int a, b, c; // Khai báo 3 biến int, không cần giá trị.

Scanner sc = new Scanner(System.in); // Tạo đối tượng Scanner

System.out.print("Nhập a: ");

a = sc.nextInt();

System.out.print("Nhập b: ");

b = sc.nextInt();

System.out.print("Nhập c: ");

c = sc.nextInt();

if (laTamgiac(a,b,c)){

System.out.println("Là tam giác!");

if(laTamgiacDeu(a,b,c)){

System.out.println("Và còn đều nữa!");

// Là tam giác đều thì không cần kiếm tra điều kiện còn lại nữa.

}else {

if (laTamgiacVuong(a, b, c)) {

System.out.println("Và còn vuông nữa!");

// Không thể xảy ra vuông cân. Vì chúng ta đầu vào chỉ là số nguyên.

// Còn muốn đầy đủ, bạn phải kiểm tra trường hợp vừa vuông vừa cân nữa.

}

if (laTamgiacCan(a, b, c)) {

System.out.println("Và còn cân nữa!");

}

}

}else{

System.out.println("Không phải là tam giác!");

}

}

public static boolean laTamgiac(int a, int b, int c) {

if ((a + b) > c && (a + c) > b && (b + c) > a) {

// Tam giacs: tổng 2 cạnh phải lớn hơn cạnh còn lại

return true;

} else {

return false;

}

}

public static boolean laTamgiacVuong(int a, int b, int c){

if ((a\*a + b\*b) == c\*c || (a\*a + c\*c) == b\*b || (b\*b + c\*c) == a\*a) {

// Là tam giác vuông nếu có 1 trong các đièu kiện thoả mãn pythagore.

return true;

} else {

return false;

}

}

public static boolean laTamgiacCan(int a, int b, int c){

if (a == b || b == c || c == a) {

return true;

}else{

return false;

}

}

public static boolean laTamgiacDeu(int a, int b, int c){

if (a == b && b == c) {

return true;

}else{

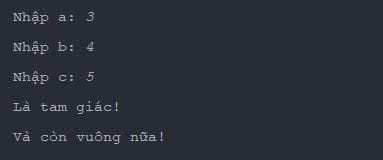
return false;

}

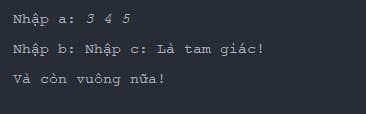
}

}

Kết quả:



1 Thứ hay ho, nếu bạn nhập liên tục các số như này:



Nó vẫn hiểu! và có vẻ như nó bỏ qua luôn các bước nhập b và nhập c

Vì sao? tới đây mình muốn nói kỹ hơn về bản chất của việc nhập vào bàn phím.

**Bản chất của next**

Bạn để ý là các hàm lấy giá trị từ bàn phím đều có chữ next. Bây giờ bạn chạy cho mình ví dụ này, bạn sẽ hiểu:

public static void main(String[] args) {

int a,b,c;

Scanner sc = new Scanner(System.in); // Tạo đối tượng Scanner

System.out.print("Nhập a: ");

a = sc.nextInt();

b = sc.nextInt();

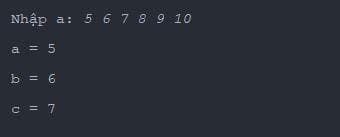
c = sc.nextInt();

System.out.println("a = "+a);

System.out.println("b = "+b);

System.out.println("c = "+c);

}



Bạn sẽ thấy là, nó đưa **tuần tự** các giá trị hiện có trên bàn phím vào các biến. bản chất của chữ next chính là **tuần tự**. Nó sẽ chờ bạn nhập nếu không có giá trị gì trên màn hình, nhưng nếu đã có sẵn giá trị rồi, nó sẽ ghi nhớ trong bộ đệm và khi gặp hàm nextInt() nó không chờ nữa, mà nó lấy luôn cái giá trị còn thừa ra, chưa sử dụng đến để gắn luôn vào biến 😂

Nhìn như như này cho dễ hiểu:

public static void main(String[] args) {

int a,b,c;

Scanner sc = new Scanner(System.in); // Tạo đối tượng Scanner

System.out.print("Nhập a: ");

a = sc.nextInt(); // Chờ bạn nhập.

// bạn nhập: 5 6 7 8 9 10

// bộ đệm = 5 6 7 8 9 10

// lấy 5 ra, gắn vào a

// bộ đệm còn: 6 7 8 9 10

b = sc.nextInt(); // gặp lệnh nextInt()

// thấy bộ đệm còn, lấy 6 ra, gắn vào b

// bộ đệm còn: 7 8 9 10

c = sc.nextInt(); // gặp lệnh nextInt()

// thấy bộ đệm còn thừa, lấy 7 ra, gắn vào b

// bộ đệm còn: 8 9 10

System.out.println("a = "+a); // in a

System.out.println("b = "+b); // in b

System.out.println("c = "+c); // in c

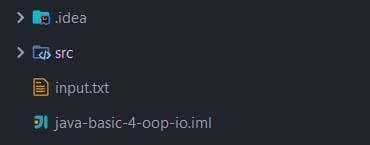
}

**Inpụt/ outpụt từ File**

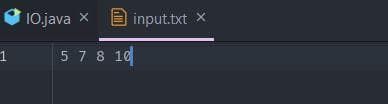
Để đề phòng thế giới bị phá hoại... ==! lại xàm r

Để cho thuận tiện trong việc đọc ghi, thì ngoài bàn phím, một trong những yêu cầu quan trọng khi lập trình đó là nhập xuất dữ liệu từ File, phần này sẽ không khác nhiều với từ bàn phím đâu các bạn, mình sẽ hướng dẫn.

Tại thư mục gốc của project, bạn click New > File. Tạo 1 tệp tên là input.txt. Như hình:



Bạn mở file ra và nhập dữ liệu vào dòng đầu tiên như lày:



Sau đó vào code ở trên, và sửa 1 dòng như thế này:

public static void main(String[] args) throws FileNotFoundException { // Thêm cái này vào đây

int a,b,c;

Scanner sc = new Scanner(new File("input.txt")); // Tạo đối tượng Scanner đọc tới cái file vừa tạo

System.out.print("Nhập a: ");

a = sc.nextInt();

b = sc.nextInt();

c = sc.nextInt();

System.out.println("a = "+a); // in a

System.out.println("b = "+b); // in b

System.out.println("c = "+c); // in c

}

// Kết quả chạy:

// Nhập a: a = 5

// b = 7

// c = 8