

JAVA	PYTHON
Importación de librerías para presentación y solicitud de información en Java	
import java.util.Scanner; //modo consola import javax.swing.JOptionPane; //modo gráfico //Este no requiere importar ni las librerías matemáticas ni las de sistema	#Este es un ejemplo para importar el módulo de formato JSON import json #otro ejemplo para importar módulos. Este es para crear aleatorios #(también se puede ...import randint para enteros) from random import random
Presentación de información en Java	Presentación de información en Python
System.out.println("mensaje"); //modo consola JOptionPane.showMessageDialog(null, "mensaje"); //modo gráfico	print("mensaje") #solo hay modo consola
Definición de variables en Java	Definición de variables en Python
// es OBLIGATORIO definir variables byte variablepequena; short variablemediana; int variablegrande; long variablemuygrande; float variabledecimalpequena; double variabledecimalgrande; String variabletexto; boolean variablelogica;	#no es obligatorio definir variables, pero es buena práctica variablegrande = int() variabledecimalgrande = float() variabletexto = str() variablelogica = bool()
Forzamiento de tipo durante cálculo (casting)	Forzamiento de tipo durante cálculo (casting)
variableentera = (int) (4 / 3) //fuerza un valor de un tipo a otro, en este caso genera un valor de 1 //En general se fuerzan tipos de datos de la misma naturaleza	variableentera = int(4 / 3) #Fuerza un valor de un tipo a otro. En general se fuerzan tipos de datos de la misma naturaleza #En Python, el casting se usa igual que el parsing
Conversión de datos en Java para captura (parsing o transformación)	Conversión de datos en Python para captura (parsing o transformación)
variablepequena = Byte.parseByte(variabletexto); variablemediana = Short.parseShort(variabletexto); variablegrande = Integer.parseInt(variabletexto); variablemuygrande = Long.parseLong(variabletexto); variabledecimalpequena = Float.parseFloat(variabletexto); variabledecimalgrande = Double.parseDouble(variabletexto); variablelogica = Boolean.parseBoolean(variabletexto); variabletexto = Integer.toString(variableentera);	#En Python, el casting se usa igual que el parsing variablegrande = int(variabletexto) variabledecimalgrande = float(variabletexto) variablelogica = bool(variabletexto) variabletexto = str(variablenumerica)

JAVA	PYTHON
Solicitud de datos en Java (se obtiene como texto)	Solicitud de datos en Python (se obtiene como texto)
<pre>//modo consola Scanner teclado = new Scanner(System.in); System.out.println("Ingresa un texto cualquiera"); variabletexto = teclado.nextLine(); //modo gráfico variabletexto = JOptionPane.showInputDialog(null, "Ingresa un texto cualquiera");</pre>	<pre>print("Ingresa texto") variabletexto = input() variabletexto = input("Ingresa un texto cualquiera: ")</pre>
Solicitud de datos en Java (con conversión desde texto)	Solicitud de datos en Python (con conversión desde texto)
<pre>//modo consola variablegrande = Integer.parseInt(teclado.nextLine()); //Se pueden usar funciones especializadas en captura de datos de un tipo particular como //teclado.nextInt() pero se debe siempre usar un nextLine() en blanco después de la captura numérica variablegrande = teclado.nextInt(); teclado.nextLine(); //modo gráfico, permite decorar con texto de pregunta variablegrande = Integer.parseInt(JOptionPane.showInputDialog(null, "Ingresa número")); variabledecimalgr = Double.parseDouble(JOptionPane.showInputDialog(null, "Ingresa número")); variablelogica = Boolean.parseBoolean(JOptionPane.showInputDialog(null, "true / false?"));</pre>	<pre>#modo consola print("Ingresa número entero") variablenumerica = int(input()) print("Ingresa número decimal") variabledecimal = float(input()) print("true / false?") varialelogica = bool(input()) #modo consola permite decorar con texto de pregunta variablenumerica = int(input("Ingresa número entero")) variabledecimal = float(input("Ingresa número decimal")) varialelogica = bool(input("True / False?"))</pre>
Concatenación en Java	Concatenación en Python
<pre>System.out.println("mensaje" + variable); //modo consola JOptionPane.showMessageDialog(null, "mensaje" + variable); //modo gráfico</pre>	<pre>print("mensaje" + str(variable)) # modo consola # si la variable no es str, toca convertir para concatenar print("mensaje" * 3) # mensajes multiplicados</pre>
Algunas funciones matemáticas en Java	Algunas funciones matemáticas en Python
<pre>variabledecimalgrande = Math.pow(base, exponente) variabledecimalgrande = Math.sqrt(numero) variabledecimalgrande = Math.random() //genera un número real aleatorio entre 0 y 0.99999999 variableentera = (int) (Math.random() * n) //genera un número entero aleatorio entre 0 y n - 1</pre>	<pre>variabledecimalgrande = base ** exponente variabledecimalgrande = math.sqrt(numero) variableentera = randint(0, n) #genera un número #entero aleatorio entre 0 y n (se importa randint de random) variabledecimalgrande = random() #genera un número real aleatorio entre 0 y 0.99999999 #(se importa random)</pre>

JAVA	PYTHON
Estructuras de control condicional en Java	Estructuras de control condicional en Python
<pre>if (condición) { //instrucciones; } else if (condición) { //estas otras opciones son opcionales //instrucciones; } else { //instrucciones; } //instrucciones fuera de la estructura de control //También se pueden usar estructuras condicionales anidadas o secuenciales</pre>	<pre>if (condición): #instrucciones. No se coloca entre corchetes, se maneja solo con indentación. #Si sale de la indentación se sale de la estructura elif (condición): #estas otras opciones son opcionales #instrucciones else: #instrucciones #instrucciones fuera de la estructura de control #También se pueden usar estructuras condicionales anidadas o secuenciales</pre>
Estructuras de control condicional por casos en Java	Estructuras de control condicional por casos en Python
<pre>switch (variable) { case 1: //instrucciones; break; case 2: //instrucciones; break; default: //instrucciones para caso por defecto; break; }</pre>	<pre>match variable : case 1: //instrucciones; case 2: //instrucciones; case _: //instrucciones para caso por defecto;</pre>
Estructuras de control iterativo indefinido en Java – condición previa	Estructuras de control iterativo indefinido en Python – condición previa
<pre>//Tener en cuenta los usos de variables para control de ciclos //(contadores, acumuladores, banderas, opciones) //Condición previa al ciclo while (condición) { //instrucciones; }</pre>	<pre>#Tener en cuenta los usos de variables para control de ciclos # (contadores, acumuladores, banderas, opciones) //Condición previa al ciclo while (condición) : #instrucciones. No se coloca entre corchetes, se maneja con indentación. # Si sale de la indentación se sale de la estructura #instrucciones fuera de la estructura de control</pre>

JAVA	PYTHON
Estructuras de control iterativo indefinido en Java – condición posterior	Estructuras de control iterativo indefinido en Python – condición posterior
<pre>//Condición posterior al ciclo do { //instrucciones; } while (condición); //interrupción forzada //dentro de cualquier ciclo, se puede forzar su cierre de inmediato dependiendo de otra condición //distinta a la inicial // dentro de cualquier ciclo colocar: if (otra_condición) break;</pre>	<pre>#Condición posterior al ciclo #No existe. Se usa el anterior, pero se hace un ciclo de condición previa infinito con True #y se obliga a salir con interrupción forzada al cumplirse la condición while (True) : #instrucciones if (condición) : break</pre>
Estructuras de control iterativo definido en Java	Estructuras de control iterativo definido en Python
<pre>//ciclo definido que repite desde 1 hasta que llegue a 7 (< 8), con paso de 1 //obligatorio colocar el paso. i debe estar declarado antes como entero for (int i = 1; i < 8; i++) { //instrucciones; }</pre>	<pre>#ciclo definido que repite desde 1 hasta que llegue a 7 (< 8), con paso de 1 #no es obligatorio colocar el paso for i in range(1, 8, 1): #instrucciones</pre>
Estructuras básicas de datos (arreglos) en Java	Estructuras básicas de datos (listas) en Python
<pre>int [] arregloenteros = new int[6]; //declara y aparta memoria para un arreglo de 6 enteros double [] arregloreales = new double[12]; //declara y aparta memoria para arreglo de 12 reales String [] arreglotextos = new String[5]; //declara y aparte memoria para un arreglo de 5 textos arregloreales[4] = 250.4258; //asigna un valor al elemento [4] del arreglo. Los índices inician en 0 arregloenteros = { 25, 48, -7, 8, 15, -65 }; //asigna los valores del arreglo de una vez. Se hace entre //corchetes int tamano = arregloreales.length; //obtiene el tamaño del arreglo</pre>	<pre>arregloenteros = [1, 8, 27, 65, 125] # asigna los valores de una #lista de una vez, no necesita apartar memoria antes arregloenteros[3] = 64 #asigna un valor al elemento [3] tamano = int(len(arregloenteros)) #obtiene el tamaño</pre>
Recorrido iterativo en estructuras de datos en Java	Recorrido iterativo en estructuras de datos en Python
<pre>//ciclo definido que recorre cada dato encontrado en la colección, dato se declara con el tipo de variable //de los elementos de la colección, y dato se usa dentro las instrucciones for (int dato : coleccion) { //instrucciones; }</pre>	<pre>#ciclo definido que recorre cada dato encontrado en la colección, y dato se usa dentro las instrucciones for dato in coleccion: #instrucciones</pre>