/\*  
2. Ejercicio sobre literales: indica qué literales Java son válidos.  
Indica cuáles de los siguientes literales son válidos en Java. Si el literal es válido indica además de qué tipo es  
(int, double, long, etc) y el sistema de numeración en el que está escrito (decimal, binario, octal, hexadecimal).  
Si el literal no es válido explica porqué no lo es.  
  
1) 0.5 - Válido, tipo double, base decimal.  
2) .5 - Válido, tipo double, base decimal.  
3) 9.3e12 - Válido, tipo double, base decimal.  
4) 9.3e-12 - Válido, tipo double, base decimal.  
5) 12345678 - Válido, tipo int, base decimal.  
6) 12345678\_L - Válido, tipo long, base decimal.  
7) 0.8E+0.8 - No válido, el exponente tiene que ser entero.  
8) 0.8E 8 - No válido, no está permitido el espacio entre E y 8.  
9) 05\_15 - Válido, tipo int, base octal.  
10) 018CDF - No válido, si se tratara de un int de base hexadecimal deberia comenzar con 0x.  
11) 0XBC5DA - Válido, tipo int, base hexadecimal.  
12) 0x87e3a - Válido, tipo int, base hexadecimal.  
13) 234567L - Válido, tipo long, base decimal.  
14) 0\_B11 - No válido, si se tratara de un numero int de base binario no puede separarse el 0B con el símbolo '\_'.  
15) 010101 - Válido, tipo int, en binario.  
16) 0\_557 - Válido, tipo int, base octal.  
17) .00.8E2 - No válido, la base solo puede contener un punto que separe la parte entera de la base decimal.  
18) .3e3f - Válido, tipo float.  
19) 0b111 - Válido, tipo int, en binario.  
20) 12\_234L - Válido, tipo int, base decimal.  
21) 0Xabcd - Válido, tipo int, base hexadecimal.  
22) 0xabcEL - Válido, tipo long, base hexadecimal.  
23) \_234 - No válido, el símbolo '\_' no puede estar al principio del literal.  
24) 1010B - No válido, si se tratara de una int en binario deberia comenzar con 0B no finalizar.  
25) 0x1010B - Válido, tipo int, base hexadecimal.  
26) 1\_234.2E-2 - Válido, tipo double, base decimal.  
27) 1234.2EF - No válido, si se tratara de un numero int de base hexadecimal deberia comenzar con 0x.  
28) 1234.2E3F - Válido, tipo float, base decimal.  
29) 1\_1.2e\_2 - No válido, no puede aparecer el simbolo '\_' antes o despues de 'e'.  
30) 0bABCDL - No válido, un número de tipo int en binario no puede contener letras.  
31) 0X1A - Válido, tipo int, base hexadecimal.  
32) 0X12AL - Válido, tipo long, base hexadecimal.  
33) abcd - No válido, si se tratara de un int de base hexadecimal deberia comenzar con 0x.  
34) 0125 - Válido, tipo int, base octal.  
35) .01011 - Válido, tipo double, base decimal.  
36) 3e12 - Válido, tipo int, base decimal.  
37) 3\_e12 - No válido, no puede aparecer el simbolo '\_' antes o despues de la 'e'.  
38) -3E-1\_2 - Válido, tipo double, base decimal.  
39) 0.8E - No válido, ya que no hay exponente despues de la 'e'.  
40) 0B1212 - No válido, si se tratara de un numero int en binario deberia contener unicamente 0 y 1.  
41) 1\_2\_3 - Válido, tipo int, base decimal.  
42) 0xedad - Válido, tipo int, base hexadecimal.  
43) 0XBE2 - Válido, tipo int, base hexadecimal.  
44) 101e2 - Válida, tipo int, base decimal.  
45) B1101 - No válido, si se tratara un numero int en binario deberia comenzar por 0b.  
46) 1.34.5 - No válido, un numero no puede contener mas de un '.'.  
47) 12.3E4F - Válido, tipo float, base decimal.  
48) 0X12AG - No válido, si se tratara de un numero int de base hexadecimal deberia contener letras entre a y f.  
\*/  
/\*  
3. Ejercicio sobre literales de tipo char  
Indica cuáles de los siguientes literales de tipo char son válidos en Java. Si el literal no es válido explica el motivo.  
  
1) 'a' - Valido.  
2) '$' - Válido.  
3) '\n' - Válido.  
4) '/n' - No válido, un literal de tipo char debe contener unicamente un caracter a no ser que se trate de una secuencia de escape.  
5) '\\' - Válido.  
6) '\ñ' - No válido, un literal de tipo char debe contener unicamente un caracter a no ser que se trate de una secuencia de escape.  
7) "T" - No válido, ya que un literal de tipo char debe estar entre comillas simples.  
8) 'ñ' - Válido.  
9) 'xyz' - No válido, un literal de tipo char debe contener unicamente un caracter a no ser que se trate de una secuencia de escape.  
10) '\u0066' - Válido.  
11) "XYZ" - No válido, ya que contiene mas de un caracter y ademas esta entre comillas dobles.  
12) '4' - Válido.  
13) '\t' - Válido.  
14) '\b' - Válido.  
15) k - No válido, ya que no esta contenido entre comillas simples.  
16) + - No válido, ya que no esta contenido entre comillas simples.  
17) '+' - Válido.  
18) '?' - Válido.  
19) 'â' - Válido.  
20) ':' - Válido.  
  
/\*  
4. Ejercicio sobre literales de tipo String  
Indica cuáles de los siguientes literales de tipo String son válidos en Java. Si el literal no es válido explica el motivo.  
  
a) '8:15 P.M.' - No valido, los literales de tipo string deben estar contenidos entre comillas dobles.  
b) "Rojo, Blanco y Azul" - Valido.  
c) "Nombre: - No valido,  
d) "Capítulo \’3\’" - Válido.  
e) "1.3e-1-2" - Valido.  
f) "" - Valido  
g) " " - Valido.  
h) "A" - Valido.  
i) "FP' - No valido, el literal debe estar contenido entre comillas dobles.  
j) " programación " - Valido.  
k) "programación "Java"" - No valido, ya que para usar comillas dobles dentro del literal debemos usar la secuencia /".  
l) programación - No valido, ya que el literal debe estar contenido entre comillas dobles.  
m) 'W' - No valido, ya que el literal debe estar contenido entre comillas dobles.  
n) "\n" - Valido.  
o) "4 + 5 \* 2" - Valido.  
 \*/  
/\*  
5. Ejercicio sobre declaraciones de variables  
Ejercicio 1: Escribe la declaración para cada una de las variables y asígnales un valor inicial en la propia declaración de variable.  
  
a) Variables enteras: p, q  
int p = 0, q = 3;  
b) Variables float: x, y, z  
float x = 0.5f, y = 1.2F, z = -0.5f;  
c) Variables carácter: a, b, c  
char a = 'a', b = 'b', c = 'c';  
d) Variables double: raiz1, raiz2  
double raiz1 = 1.2e-5, raiz2 = 5.68794;  
e) Variable entera larga: contador  
long contador = 5e256;  
f) Variable entera corta: indicador  
short indicador = 5;  
g) Variable entera: indice  
int indice = 0;  
h) Variables double: precio, precioFinal  
double precio = 0.56, precioFinal = 1.26;  
i) Variables carácter: car1, car2  
char car1 = 'g', car2 = 'f';  
j) Variable tipo byte: valor  
byte valor = 0b0101;  
k) Variables lógicas: primero, ultimo  
boolean primero = true, ultimo = false;  
l) Variable de tipo cadena de caracteres: nombre  
String nombre = "Hola Mundo";  
  
Ejercicio 2: Escribe la declaración más apropiada para cada una de las siguientes variables. El nombre de cada una  
indica el tipo de dato que contendrá y servirá para determinar el tipo de dato más adecuado en la declaración  
de la variable.  
  
a) edad  
int edad;  
b) códigoPostal  
int códigoPostal;  
c) altura  
float altura;  
d) genero (valores: H: hombre, M: mujer)  
char genero;  
e) nombre  
String nombre;  
f) númeroDeHijos  
int númeroDeHijos;  
g) iva (valores: % a aplicar)  
double iva;  
h) tallaCamisa (valores numéricos)  
int tallaCamisa;  
i) peso  
double peso;  
j) precio  
double precio;  
k) alumnoRepetidor (valores: SI/NO)  
String alumnoRepetidor;  
l) mensaje  
String mensaje;  
m) letra  
char letra;  
n) mayorEdad  
String mayorEdad;  
o) minutos  
int minutos;  
p) días  
int días;  
q) matriculaCoche  
String matriculaCoche;  
r) contador  
int contador;  
s) mayorDeEdad (valores: verdadero/falso)  
boolean mayorDeEdad;  
t) tallaCamiseta (valores: S, L, XL...)  
String tallaCamisa;  
 \*/  
/\*  
6. Convertir expresiones algebráicas en expresiones algorítmicas  
Ejercicio: Convierte en expresiones algorítmicas las siguientes expresiones algebráicas. Coloca paréntesis solamente donde sean necesarios.  
  
a) 3/2 + 4/3  
b) 1/(x-5)-(3\*x\*y)/4  
c) 1/2 + 7  
d) 7 + 1/2  
e) a\*a/(b-c) + (d-e)/(f-(g\*h)/j)  
f) m/n + p  
g) m + n/(p-q)  
h) a\*a/b\*b + c\*c/d\*d  
i) (m+n/p)/(q-r/s)  
j) (3\*a + b)/(c-(d + 5\*e)/(f+g/2\*h))  
k) (a\*a + 2\*a\*b + b\*b)/(1/x\*x + 2)  
 \*/  
/\*  
8. Ejercicio sobre operadores aritméticos  
Ejercicio 1: Si a, b y c son variables enteras con valores a=8, b=3, c=-5, determina el valor de las siguientes expresiones aritméticas:  
  
a) a + b + c = 6  
b) 2 \* b + 3 \* (a – c) = 45  
c) a / b = 2.67  
d) a % b = 2  
e) a / c = -2.67  
f) a % c = 3  
g) a \* b / c = -4.8  
h) a \* (b / c) = -4.8  
i) (a \* c) % b = -1  
j) a \* (c % b) = -16  
k) (3 \* a – 2 \* b) % (2 \* a – c)   
 \*/