PALABRAS RESERVADAS EN JAVA

Java reserva ciertas palabras clave como parte del lenguaje. No hay muchas, de cualquier
modo. A continuación se muestran

abstract: Especifica la clase o método que se va a implementar más tarde en una subclase.
boolean: Tipo de dato que sólo puede tomar los valores verdadero o falso.
break: Sentencia de control para salirse de los bucles.
byte: Tipo de dato que soporta valores en 8 bits.
byvalue: Reservada para uso futuro.
case: Se utiliza en las sentencias *switch* para indicar bloques de texto.
cast: Reservada para uso futuro.
catch: Captura las excepciones generadas por las sentencias *try.*
char: Tipo de dato que puede soportar caracteres *Unicode* sin signo en 16 bits.
class: Declara una clase nueva.
const: Reservada para uso futuro.
continue: Devuelve el control a la salida de un bucle.
default: Indica el bloque de código por defecto en una sentencia *switch.*
do: Inicia un bucle *do-while.*
double: Tipo de dato que soporta números en coma flotante, 64 bits.
else: Indica la opción alternativa en una sentencia *if.*
extends: Indica que una clase es derivada de otra o de una interfaz.
final: Indica que una variable soporta un valor constante o que un método no se sobrescribirá.
finally: Indica un bloque de código en una estructura *try - catch* que siempre se ejecutará.
flota: Tipo de dato que soporta un número en coma flotante en 32 bits.
for: Utilizado para iniciar un bucle *for.*

future: Reservada para uso futuro.
generic: Reservada para uso futuro.
goto: Reservada para uso futuro.
if: Evalúa si una expresión es verdadera o falsa y la dirige adecuadamente.
implements: Especifica que una clase implementa una interfaz.
import: Referencia a otras clases.
inner: Reservada para uso futuro.
instanceof: Indica si un objeto es una instancia de una clase específica o implementa una interfaz específica.
*int:*Tipo de dato que puede soportar un entero con signo de 32 bits.
interface: Declara una interfaz.
long: Tipo de dato que soporta un entero de 64 bits.
*native:*Especifica que un método está implementado con código nativo (específico de la plataforma).
new: Crea objetos nuevos.
null: Indica que una referencia no se refiere a nada.
operator: Reservado para uso futuro. .
outer: Reservado para uso futuro.
package: Declara un paquete Java.
private: Especificador de acceso que indica que un método o variable sólo puede ser accesible desde la clase en la que está declarado.
*protected:*Especificador de acceso que indica que un método o variable sólo puede ser accesible desde la clase en la que está declarado (o una subclase de la clase en la que está declarada u otras clases del mismo paquete).
public: Especificador de acceso utilizado para clases, interfaces, métodos y variables que indican que un tema es accesible desde la aplicación (o desde donde la clase defina que es accesible).
rest: Reservada para uso futuro.
return: Envía control *y* posiblemente devuelve un valor desde el método que fue invocado.
short: Tipo de dato que puede soportar un entero de 16 bits.
*static:*Indica que una variable o método es un método de una clase (más que estar limitado a un objeto particular).

super: Se refiere a una clase base de la clase (utilizado en un método o constructor de clase).
switch: Sentencia que ejecuta código basándose en un valor.
synchronized: Especifica secciones o métodos críticos de código multihilo.
this: Se refiere al objeto actual en un método o constructor
throw: Crea una excepción.
throws: Indica qué excepciones puede proporcionar un método,
transient: Especifica que una variable no es parte del estado persistente de un objeto.
try: Inicia un bloque de código que es comprobado para las excepciones.
var: Reservado para uso futuro.
void: Especifica que un método no devuelve ningún valor.
volatile: Indica que una variable puede cambiar de forma asíncrona.
while: Inicia un bucle while